

# Basefiles-Version 2.10 and 2.11

## Basefile Explanation

Basefiles are complex, confusing, and extremely powerful, once understood.

Basefiles are designed with universal fields, however depending on the phone brand, and in some cases phone model, they generate the config files differently. The best way to use them is to make a change, then look at the config file to see if it is how you want it. It may take some playing, but they can do just about anything you want them to.

Below are 2 explanations. A simple brand, Grandstream. And the complex version, Polycom.

### Grandstream:

Grandstream and Aastra use a parameter=value system. Each basefile entry is 1 line in the config file.

Custom	Brand	Models	Device Type	Setting	Attribute	Value	Remove
--------	-------	--------	-------------	---------	-----------	-------	--------

Custom => This means that this setting is new or changed from the original

Brand => Always the brand you are in

Models => Defines which models this setting is used with. Must be EXACTLY as listed on the template page separated by commas. Blank means use it for all models.

Device Type => Only for some brands. Defines which type of device the setting applies to. Phone or FXS

Setting => The actual parameter name. In the case of Grandstream, a P followed by a number. In the case of Aastra, a parameter name.

Attribute => Used with a few brands. Grandstream and Aastra do not use it.

Value => Value for the parameter

Remove => Check this box to remove the custom change and re-instate the default setting (if available).

### Polycom:

Polycom uses a multiple XML config style. Because of this we need to define which file the setting is for, as well as what "Parent" it belongs to.

When editing Polycom basefiles, you need to select which file you want to work with. Once a file is selected, all new/edits will be for that file only.

### NOTE: The following does not work. It was an example created to show structure!

Lets add a new section to Ext.cfg (as it is small). We will call it NETWORK. Under that we will create a sub-category of LLDP. Under that we will set all of our LLDP settings. Final output should be similar to:

```
<NETWORK>
  <NETWORK.LLDP NETWORK.LLDP.EnableLLDP="1" NETWORK.LLDP.PacketInterval="120">
</NETWORK.LLDP>
</NETWORK>
```

To accomplish this, we first need to define the NETWORK section.

Because NETWORK is destined to be at the top of the XML map, we need to assign it to the category of "TOP". We also assign it a value of "PARENT" to tell the module that it has settings under it.

polycom	Phone				
---------	-------	--	--	--	--

I left the Models field blank to force it to apply to all models.

Now we need to add the sub-category of LLDP. We assign it to the category of NETWORK because that is what we want it to be under. We still assign it a value of "PARENT" because it will have settings under it.

	polycom	Phone				
--	---------	-------	--	--	--	--

Now we can add the 2 settings with their appropriate values. We only need to set the category to NETWORK.LLDP.

	polycom	Phone				
	polycom	Phone				

Once complete. Submit the changes and Regenerate the template.

This is only a simple example of what can be done. Polycom is the most complex and confusing basefile. Even the program has to add entries, regenerate and look at the configs to get them to show correctly. Have patience, and take it step by step.