File Genx – JSON templates

The purpose of this document is to describe all JSON files templates.

JSON	Description

resources.json	This file is parsed by **Res** class. Every resource, such as GUI strings, images, icons, etc, are declared within this file. The template is:
	{
	“resource-type0”: { resources-declarations },
	“resource-type1”: { resources-declarations }
	}

resources.json → resources-declarations: Strings	This is how you should declare Strings so that they can be correctly binded to the GUI:
	“Strings”: {
	“gui-object-name”:[method-to-call, [calls]]
	}
	Where: **gui-object-name** is the name that identifies the object in the galde/XML file; **method-to-call** is a string containing the name of the method to be called (e.g. “set_text” for a Gtk.Label object); **calls** is a list where all the parameters for each call that must be made to **method-to-call** are. Note that each call is a list, even if there is only one call.
	Example:
	“Strings”: {
	“someComboBoxText”:[“append”,
	[[“0”, “item0”], [“1”, “item1”]]],
	“someLabel”:[“set_text”, [[“New label text”]]]
	}
	Two calls to the append method of a Gtk object called someComboBoxText are made: first append(“0”, “item0”), then append(“1”, “item1”). Then one call to set_text method of a Gtk object called someLabel is made as set_text(“New label text”).
	The need to put each call parameter(s) inside a list is because the splat operator is used in the binding process at the method MainWidget.__bindRes().