**1. Folder structure**

!!!The folder structure and the naming is fixed. Please make sure that it is respected by your project, otherwise the tool might not work as expected

**MOD\_CONFIG/Sirius** – parent folder

**GenFiles** – contains the generated files

**Project** - the xmlpj project file is located here (described in chapter 2)

**XML** - stores xml files

**Tool** – contains the executable for SiriusConfigurator

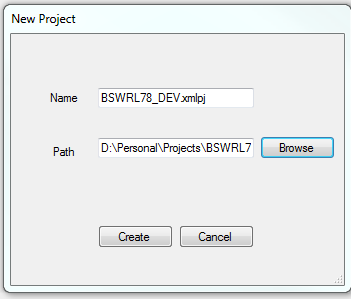
**Note**: the xsd file is located in the module‘s **Source** folder

**2. Create new project**

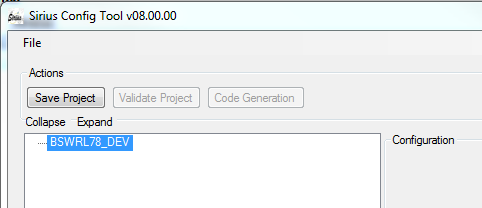
- File->New Project:

- **Name:** name of the project, ending with extension .**xmlpj**

- **Path:** path to the location where to save the project file (**..\MOD\_CONFIG\Sirius\Project**)



- **Create** button => the Project is created in Sirius

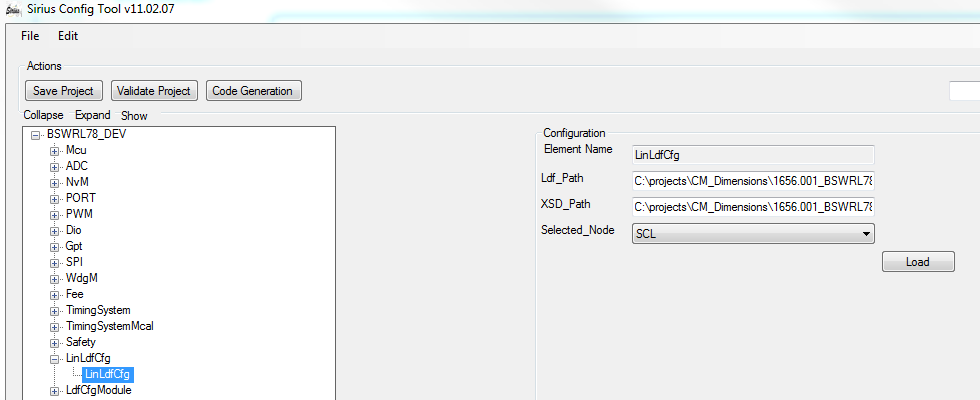


**3) Add a module in the project**

-> right click on the project: **Add Module**

- **Path xsd**: contains the path to the module’s xsd file.

For generating LIN configuration files based on LDF file, shall be added to the project LinLdfCfg.xsd from MOD\_MCAL\_LIN\_COMMON. Into the resulted attributes shall be added paths to project specific LDF and the coresponding xsd specific to the used LIN protocol into LDF from MOD\_MCAL\_LIN\_COMMON (SAEJ2602.xsd for LIN SAE J2602 protocol, v2.0.xsd for LIN 2.0 protocol, v2.1.xsd for LIN 2.1 protocol).



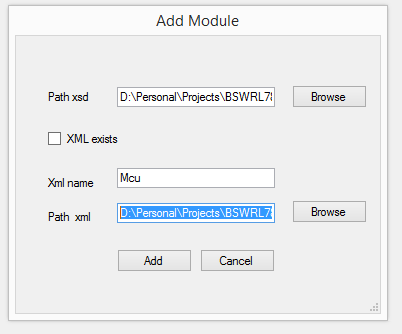
After the paths are added shall be pressed „Load“ button to add into Sirius the elements from the LDF. LdfCfgModule will be automatically created and will contain all these elements.

Into the „Selected\_Node“ atributte shall be selected the Slave for which the code will be generated.

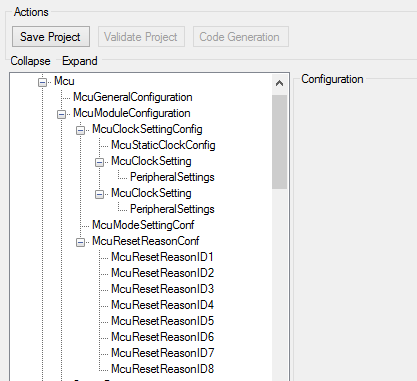
- **XML exists**: to be checked if the xml is already created

- **Xml name**: name of the xml which is derived from xsd file

**- Path xml**: the location where to save/load the xml file (..\**MOD\_CONFIG\Sirius\Project\xml\)**



* After the module is added in Sirius, the project can be saved (**Save Project**):

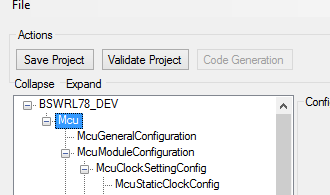


**4) Load the project**

- once a project (xmlpj file) is saved, it can be loaded : File -> Load project-> load the xmlpj file **(..\MOD\_CONFIG\Sirius\Project\BSWRL78\_DEV.xmlpj**)

**5) Delete a module from project**

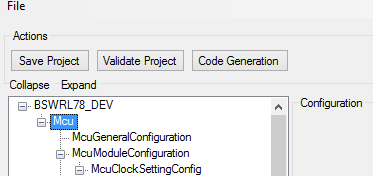
* select the module

****

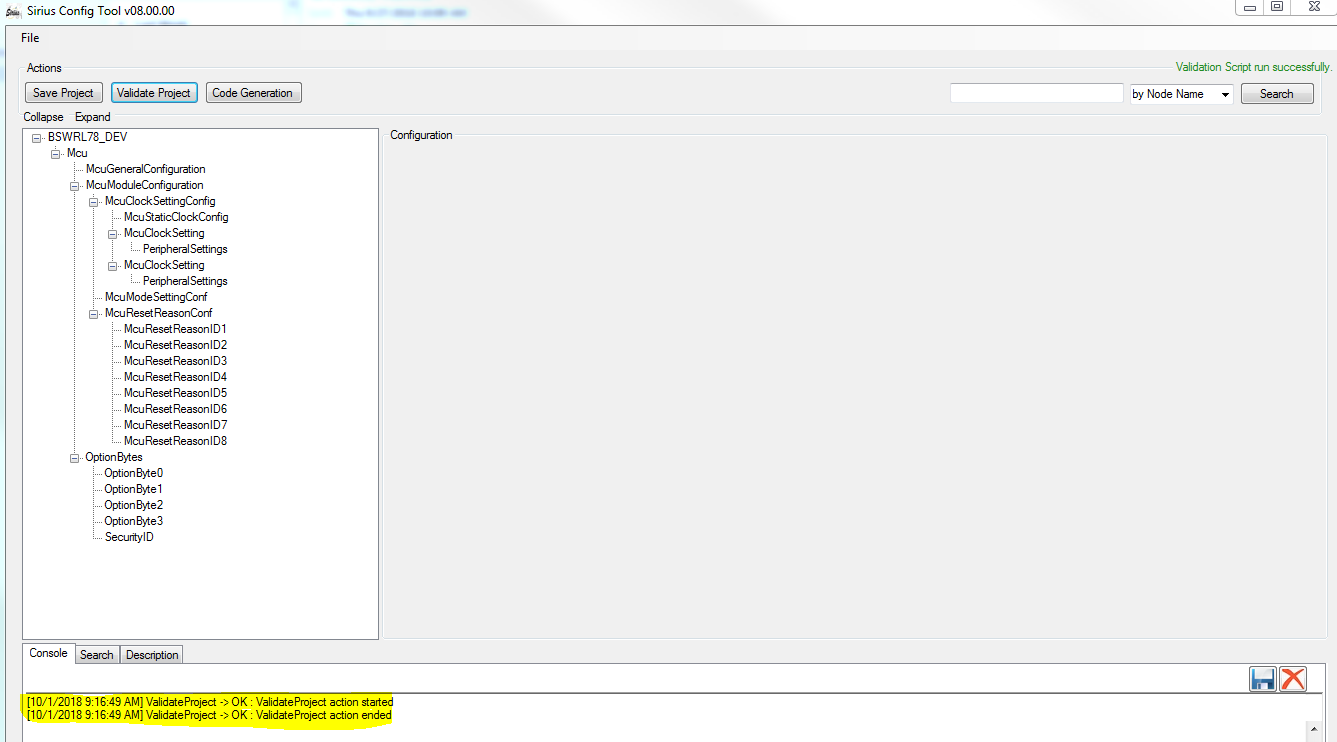
* use „**Delete**“ button on keyboard or right click and select **Delete** option

**6) Validate project**

- press **Validate Project**



-> if the project is valid, a message(**ValidateProject -> OK**) is shown in the console:



-> if the validation fails, the console shows a WARNING or ERROR message.

-> after project validation, the code can be generated

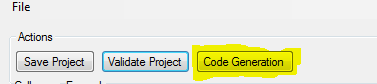
Note:

1) When the xml loaded inside Sirius has all the required data valid the “Validate Project” button will be unlocked and clickable.

2) If there is something not valid in the xml file the “Validate Project” button will remain locked.

**7) Code generation**

After the project was validated, “Code Generation button“ is unlocked. It allows the user to generate the necessary scripts.



Also, each module can be generated separately (**Right click on module -> Validate and Generate**)

Note: Files are generated at the following path: **..\MOD\_CONFIG\Sirius\GenFiles\Source\**