

gRPC LabVIEW installation and usage on Linux RT HowTo (V.0.2.3):

Tested on environment:

Server tested on:

NI-PXIe-8840Quad-Core-030EEB7A

NI Linux Real-Time x64 4.14.146-rt67-cg-8.6.0f0-x64-48 (20.6)

Client code using Python 3.7

Setup a server

Overview worksteps:

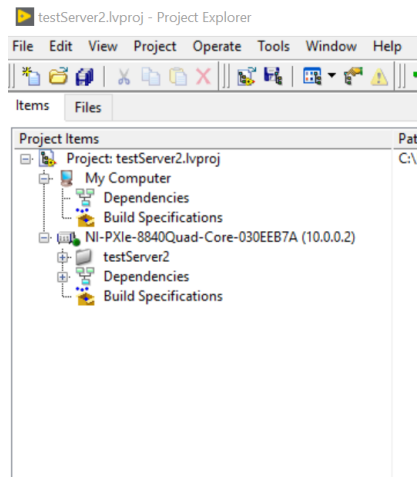
- 1) Download gRPC LabVIEW template from <https://github.com/ni/grpc-labview>
- 2) Download or create Proto file.
- 3) Use gRPC template to generate server and VI:s from Proto-file
- 4) Generate Python client using the compiled python output from Proto-file

Detailed worksteps:

- 1) Download gRPC LabVIEW
 - a. <https://github.com/ni/grpc-labview/releases/tag/v0.2.3> grpc-labview.zip
 - b. Unzip grpc-labview.zip to an arbitrary directory
- 2) Download or create Proto fil
 - a. Option1. In the downloaded repo, there is an example server containing a proto file and a client file
 - b. Option2. A protofile and a prebuilt client can be found <https://grpc.io/docs/languages/python/quickstart/>
 - c. Build you own proto file <https://developers.google.com/protocol-buffers/docs/overview>
- 3) Generate server and VI:s from Proto-file
 - a. Create a LabVIEW project that you can generate the server to
 - b. Open LabVIEW gRPC.lvproj from LabVIEW gRPC directory
 - c. Follow the guide <https://github.com/ni/grpc-labview/blob/master/docs/ServerCreation.md> and choose the desired Proto file

---FOR LINUX RT---

- d. Move the generated server to your real time target

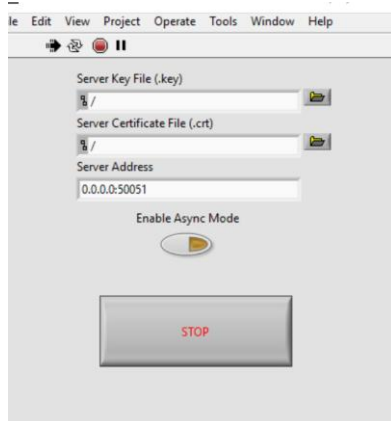


- e. Put your *.SO file on the RT target and check that Get Server DLL Path.vi points to it

- f. Write the server address and port

---For Sync / Sequential execution ---

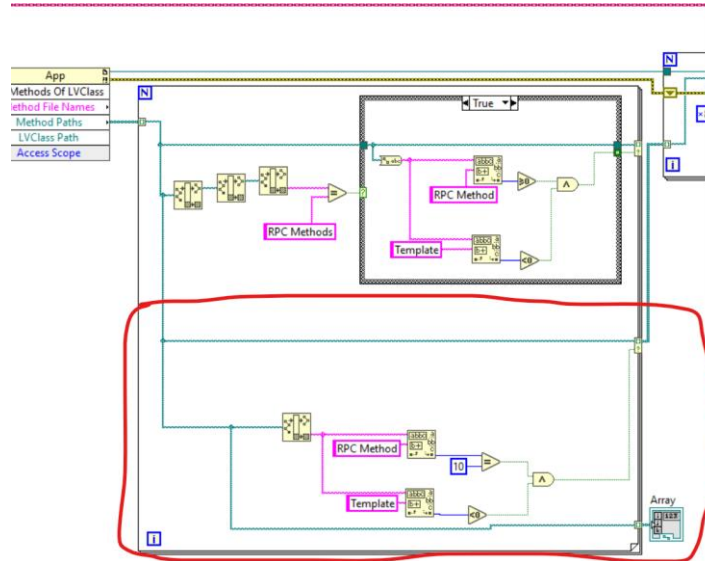
- g. Press “play” to use non Async (still async but sequential)



---For Async execution ---

- h. To use Async with interactive mode

- i. Make a change to Start Async. The change to is marked with red in the image below.



- ii. Select your server (testServer2 from the above image) and deploy all (this enables dynamic references to work)
 - iii. Press “play”
- 4) Generate or use client t python
 - a. Generate needed code
 - i. `python -m grpc_tools.protoc -I PathToProtosDir --python_out=. --grpc_python_out=. Path/myProtoFile.proto`
 - ii. For more information: `python -m grpc.tools.protoc -h`
 - b. Write client, see example <https://grpc.io/docs/languages/python/quickstart/> or https://github.com/ni/grpc-labview/blob/master/examples/query_server/Clients/python/queryserver.py