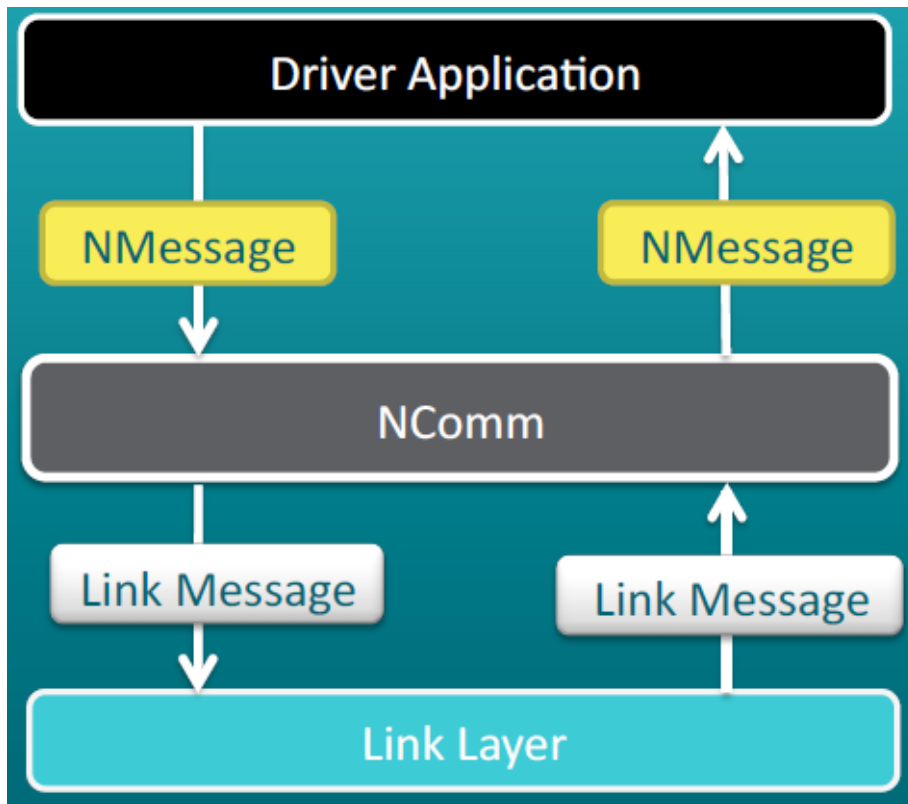


Drivers General Principles

Architecture



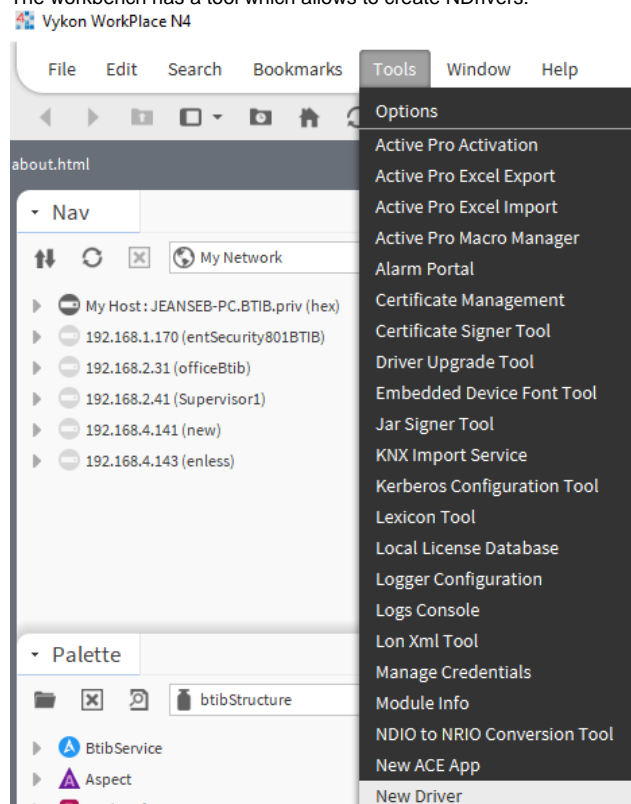
The basic architecture consists of a three layer abstraction:

- The Driver Application layer is provided by the specific driver implementation and makes use of the communication services.
- The NComm layer provides access points for the application, implements functionality associated with transport and network OSI layers, and passes messages from/to the link layer.
- The Link Layer provides bus specific functionality (Serial, Http, Udp, Tcp) and access to native driver APIs.

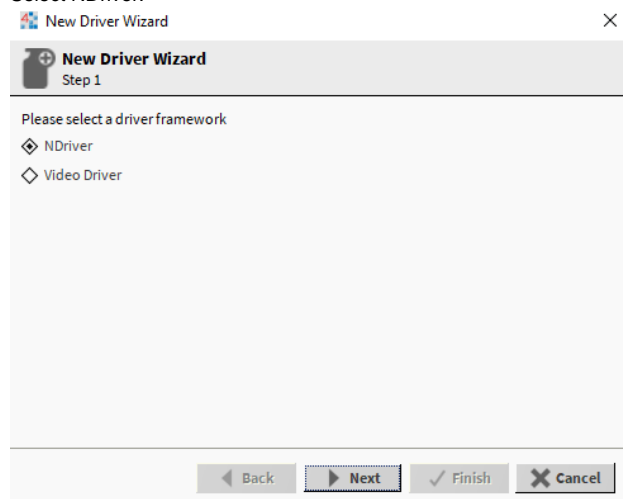
Driver specific customization is accomplished by providing extensions or implementations of commConfig classes, message classes, and message factory classes generated by the NDriver Wizard.

NDriver Wizard

The workbench has a tool which allows to create NDrivers:



Select NDriver:



This wizard will let you select which protocols your driver will use:

New Driver Wizard

New Driver Wizard - NDriver
Step 3 of 5

Select the protocols supported by the driver:

☐ HTTP

☐ TCP/IP

☐ UDP/IP

☒ Serial

☐ Process unsolicited messages

Back

Next

Finish

Cancel

And it will also propose many options to customize your plugin. Here is a table summarizing the options and their result:

Option	Class Created	Note
	BNfooNetwork	Add license check
	BNfooDevice	Add ping and poll support
	BNfooDeviceFolder	
	message/NfooMessage	
	message/NfooMessageFactory	
Serial	comm/BNfooSerialCommConfig	
	comm/NfooLinkMessage	Add begin & end of message checks to receive()
Tcp	comm/BNfooTcpCommConfig	
	comm/NfooLinkMessage	Add begin & end of message checks to receive()
Process Unsolicited Messages	comm/NfooListener	Add message processing in receiveMessage(NMessage nMsg)
Custom Device Manager	ui/BNfooDeviceManager	Customize as needed
Device Discovery (with custom manger)	learn/BNfooDeviceEntry	
	learn/BNfooLearnDeviceJob	
Device Discovery (with auto manger)	BNfooNetwork implements BINDiscoveryHost	Implement BINDiscoveryHost apis
	learn /BNfooDiscoveryPreferences	
	learn /BNfooDiscoveryLeaf	Add leaf elements
Point	point/BNfooPointDeviceExt	
	point/BNfooPointFolder	
	point/BNfooProxyExt	Add driver specified read / write
Custom Point Manager	ui/BNfooPointManager	Customize as needed
Auto Manager No Point Discovery	BNfooPointDeviceExt extends BPointDeviceExt	
Auto Manager With Point Discovery	BNfooPointDeviceExt extends BNPointDeviceExt	Implement getDiscoveryObjects

The wizard will create these classes with the principal methods to implement with some comments to help you understand their goal.