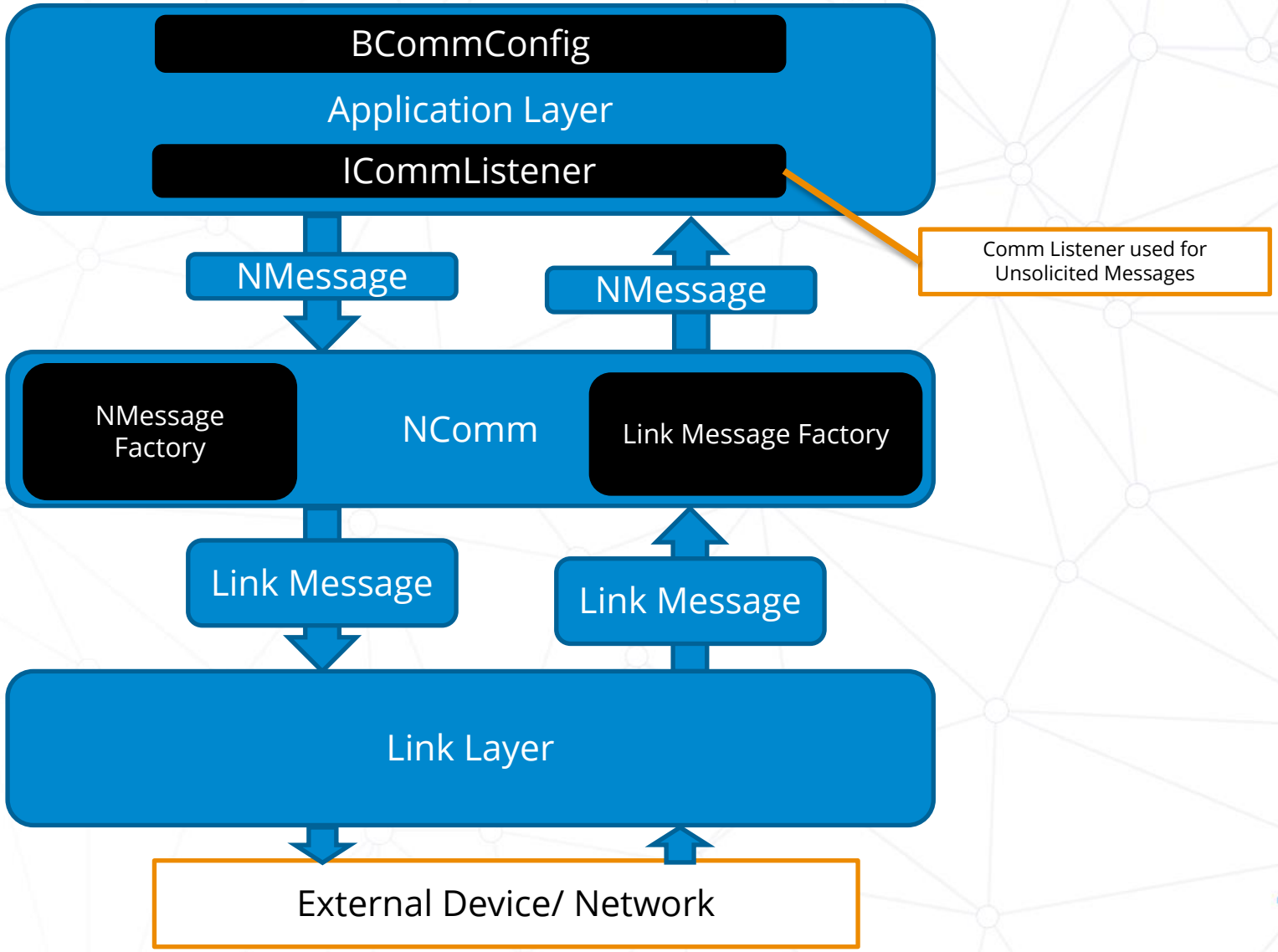
CONNECTING
THE WORLD

NDriver

Ajay Mathew
Software Engineer @ Tridium

TRIDIUM

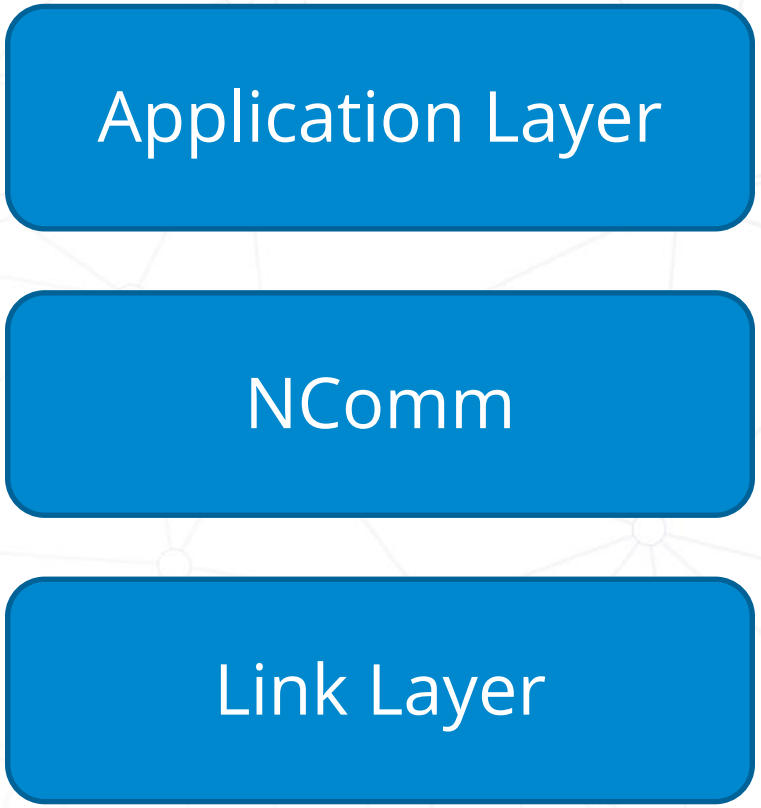
The NDriver


BCommConfig
Application Layer
ICommListener
Comm Listener used for Unsolicited Messages
NMessage
NMessage
NMessage Factory
NComm
Link Message Factory
Link Message
Link Message
Link Layer
External Device/ Network

Why Use NDriver

- Simplified and intuitive model with a clean API
- NDriver wizard, get started faster
- Less boiler plate code to implement
- Do not need to write your own:
  - TCP / UDP / Managing Sockets
  - Session handling
  - Retry & timeout logic
  - Fragment buffering
  - Manager views

3 Layer Architecture



Application Layer

- Adds minimal Application Layer functionality
  - Support is provided for adding auto manager views
  - Adds an Async worker so that operations requiring comm access can run on different thread (postAsync())

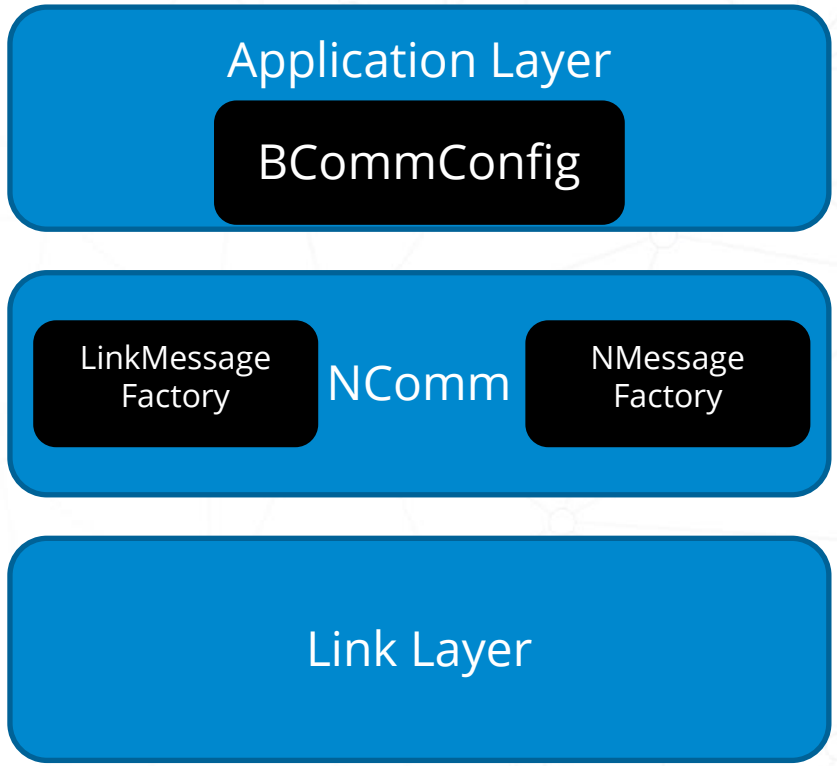
NComm Layer

- Application Layer access to NComm Services
- Transport Layer Functions:
  - Transaction Manager
  - Message Timeouts & Retries
  - Message Fragmentation
- Calls Application Layer listeners for unsolicited messages.

Link Layer

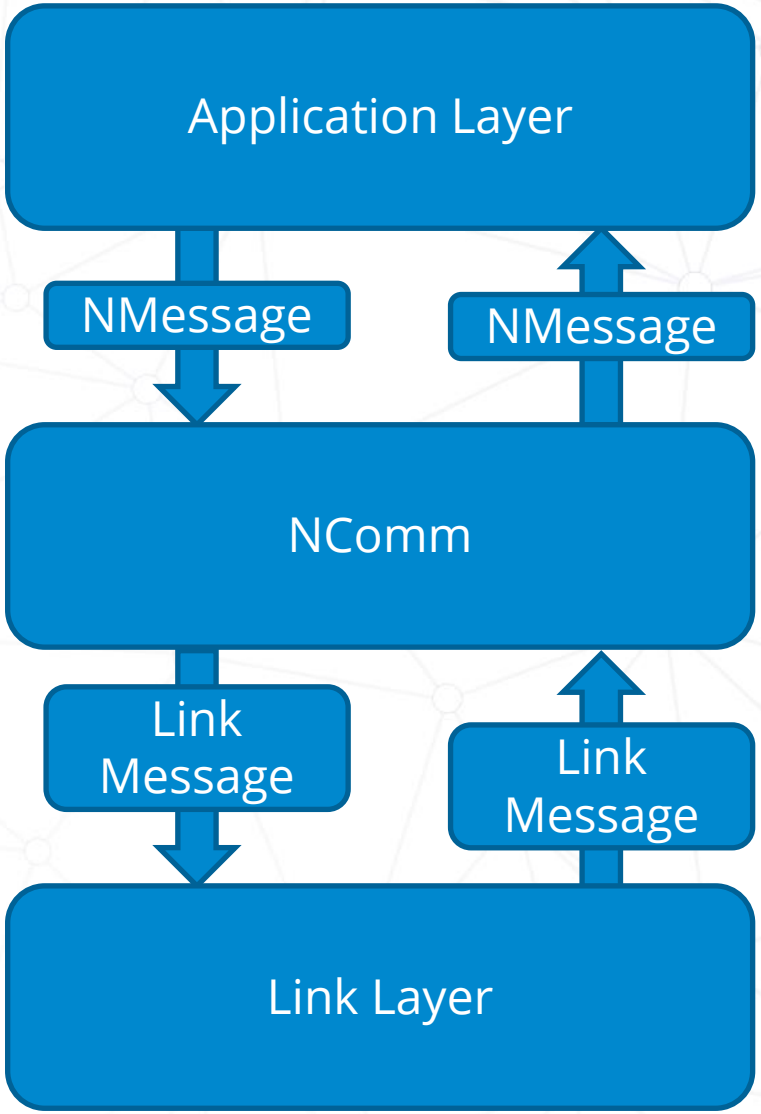
- Pluggable – Implement ILinkLayer
- Access Native Link Layer Objects
- Customization

The Lifecycle



- Link Layer Parameters
- Houses Message Factories
- Manages Comm stack lifecycle
  - started() & stopped()
  - changed()
- Any changes to CommConfig results in a call to verifySettings() allowing linkLayer resources to be updated or restarted

Message Centric

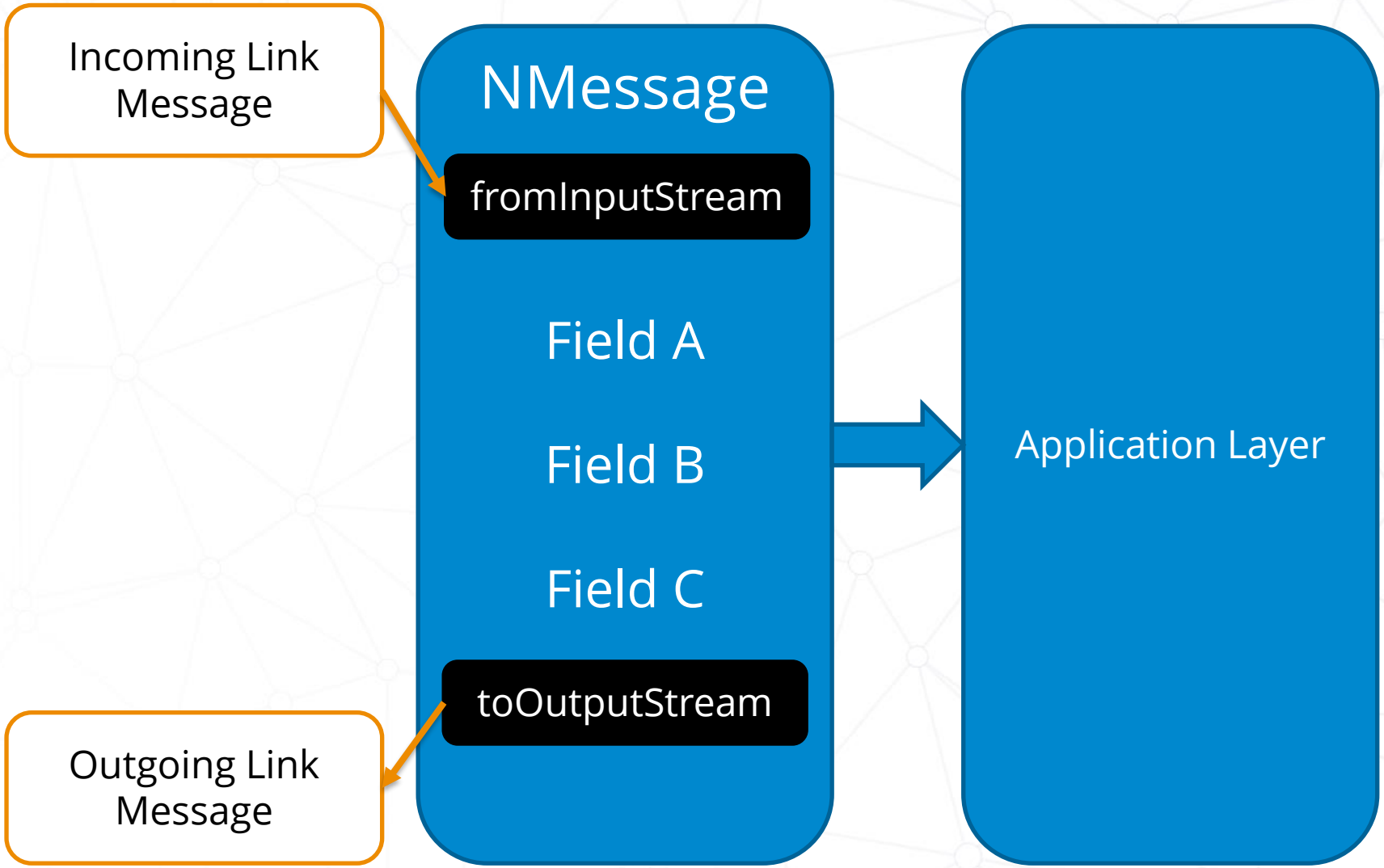


NMessage vs Link Message

- NMessage
  - Variables – message state.
  - ReadValue/ WriteValue/ Address/ Priority
  - Ping/ Connect/ Disconnect/ ReadValue/ WriteValue/ Responses.

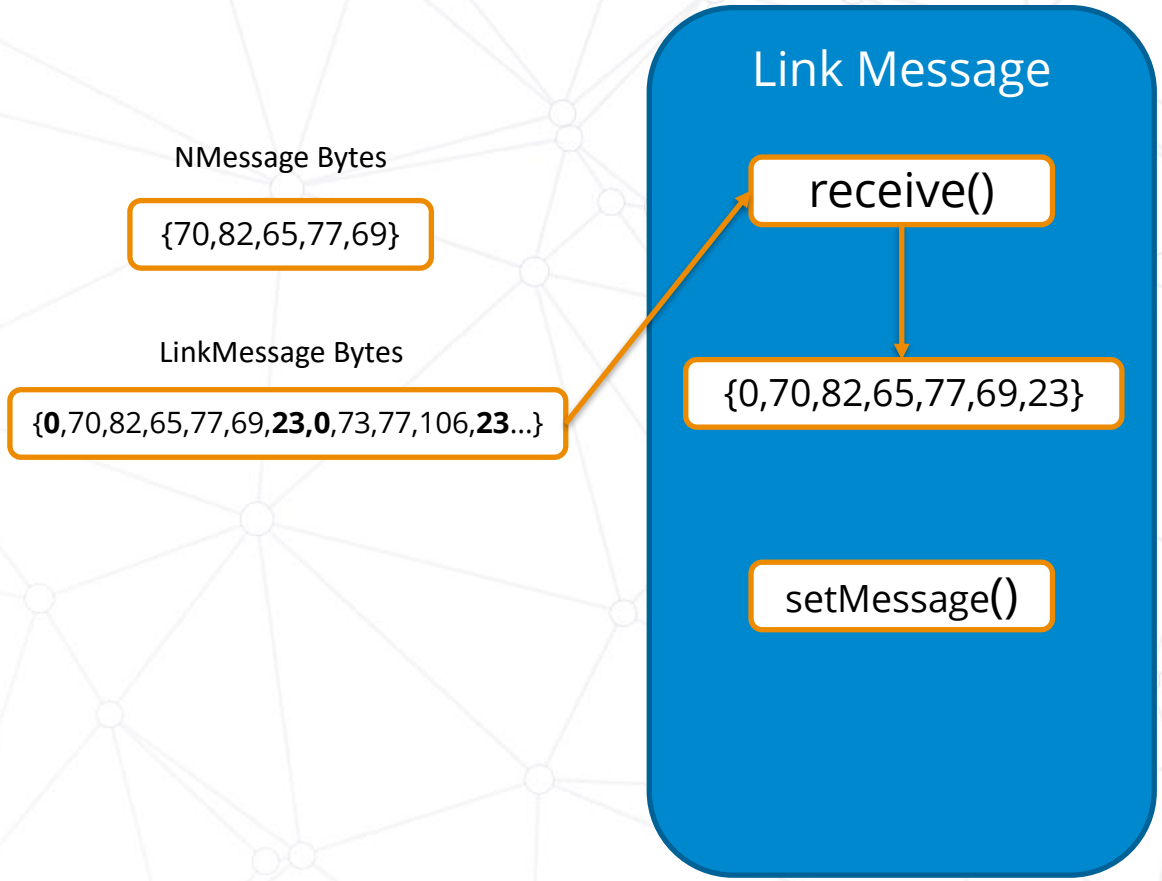
- LinkMessage
  - Byte Array
  - Typically single class
  - setMessage/ recieve

NMessage


Incoming Link Message
NMessage
fromInputStream
Field A
Field B
Field C
toOutputStream
Application Layer
Outgoing Link Message

Link Message

- Stream data to/from byte array representation.
- Link Message instances are reusable – avoid state



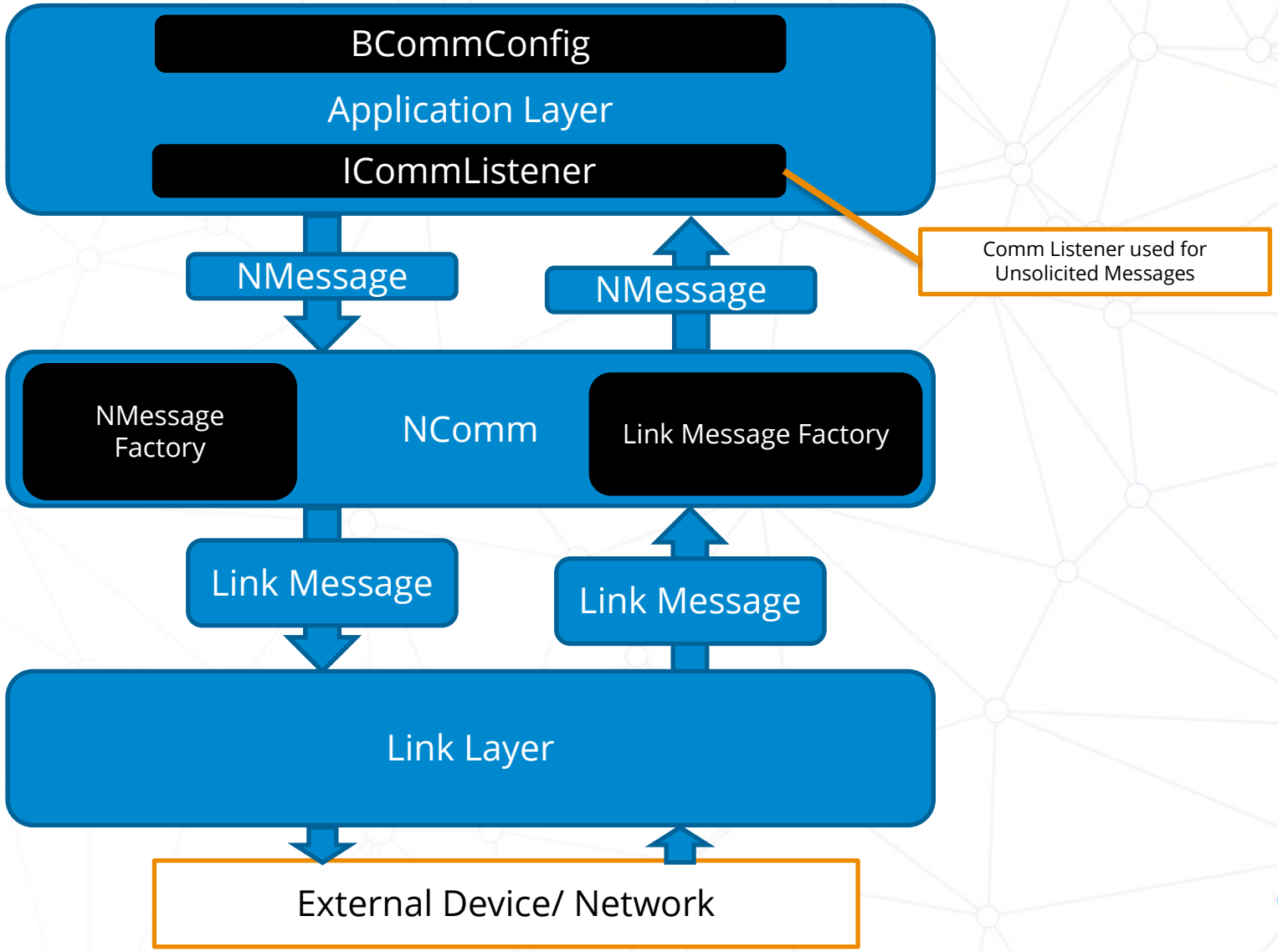
Customising NMessage

```
public class NFooMessage  extends NMessage
{
  public void fromInputStream(InputStream in) throws Exception
  {
    TypedInputStream typedInStream =  (TypedInputStream)in;
    int unsignedBytes =  typedInStream.readUnsigned8();
    int signedBytes =  typedInStream.readSigned16();
    float messageFloat =  typedInStream.readFloat();
    int bit = typedInStream.readBit(7,3,2);
  }

  public boolean toOutputStream(OutputStream out) throws Exception
  {
    TypedOutputStream typedOutStream = new TypedOutputStream();
    typedOutStream.writeUnsigned8(unsignedBytes);
    typedOutStream.writeSigned16(signedBytes);
    typedOutStream.writeBit(bit,7,3,2);
    typedOutStream.toOutputStream( out );
    return false; //no frags
  }
}
```

Link Message Outbound

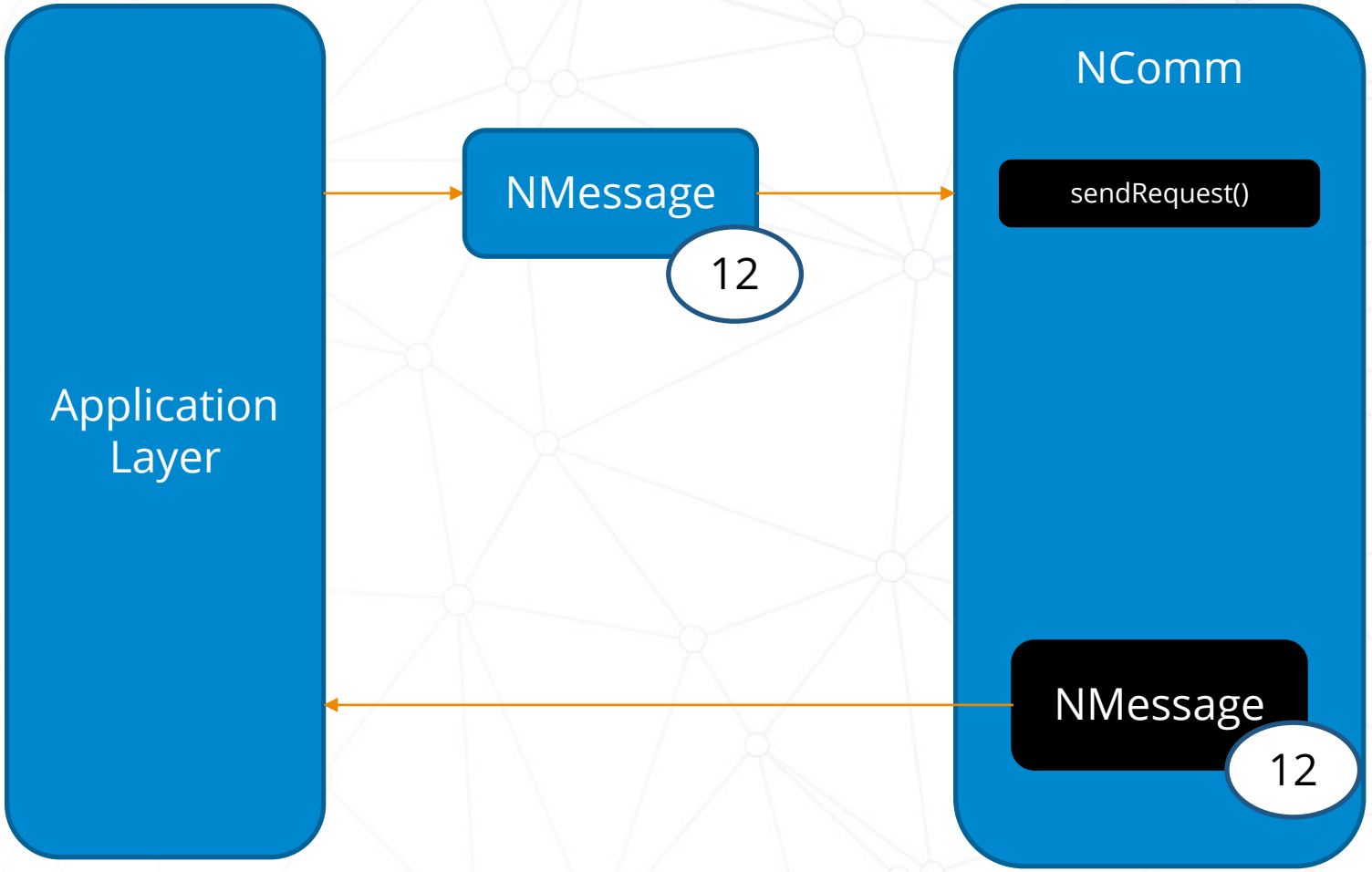
```
public class XLinkMessage extends LinkMessage
{
  ...
  public boolean setMessage(NMessage msg) throws Exception
  {
    byte[] linkMessageBytes = new byte[MAX_LINK_MSG_LENGTH];

    TypedOutputStream typedOutputStream = new TypedOutputStream(linkMessageBytes);

    msg.toOutputStream(typedOutputStream);

    OutputStream outputStream = getOutputStream();

    int msgLen = typedOutputStream.size();

    outputStream.write(BEGIN_CHAR);

    for (int i = 0; i < msgLen; ++i)
    {
      outputStream.write(linkMessageBytes[i]);
    }

    outputStream.write(END_CHAR);
    // no more fragments
    return false;
  }
}
```

The NDriver


BCommConfig
Application Layer
ICommListener
Comm Listener used for Unsolicited Messages
NMessage
NMessage
NMessage Factory
NComm
Link Message Factory
Link Message
Link Message
Link Layer
External Device/ Network

Messaging Patterns

Message Tags



Sending Unacknowledged Messages

```
NFooMessage unacked = new NFooMessage();
unacked.setAddress(new BIpAddress("10.10.11.100",23));
network.tcomm().sendMessage(unacked);
```

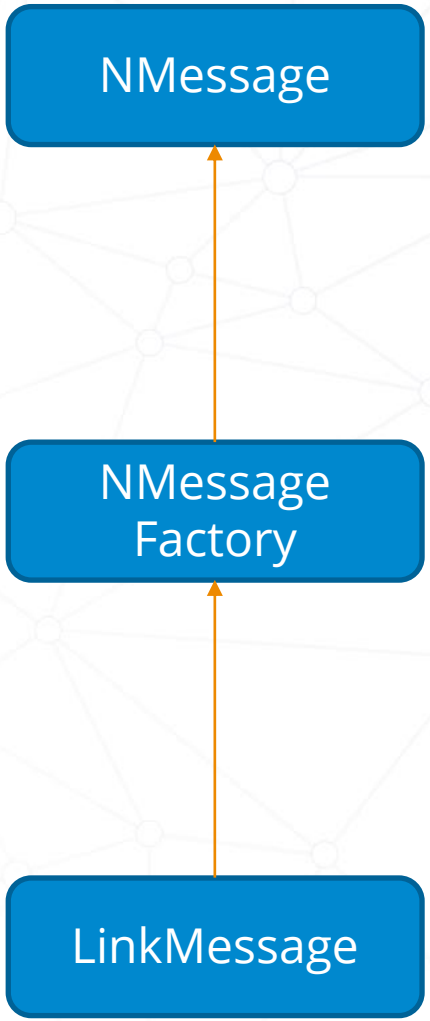
- Unacknowledged Messages can be sent with the sendMessage() method
- Non-blocking call

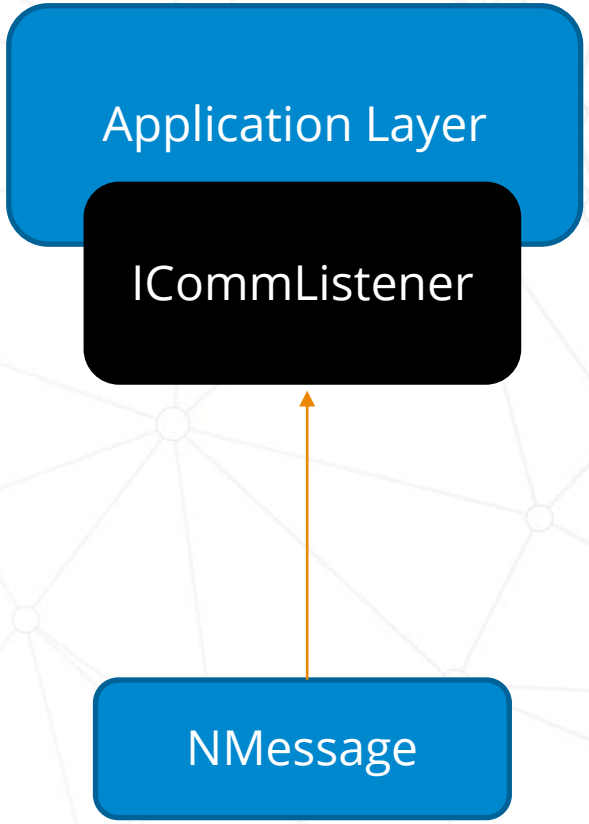
Sending Requests

```
NFooRequest request = new NFooRequest();
unacked.setAddress(new BIpAddress("10.10.11.100",23));
NMessage reply = network.tcomm().sendRequest(request);
```

- Requests block until a response message is received
- Responses are matched to Request Messages through the getTag() method
- If protocol does not support tags, the NULL tag matches first incoming response to the outgoing request
- Response Messages are generated from LinkMessage in Message Factory and must override isResponse() method

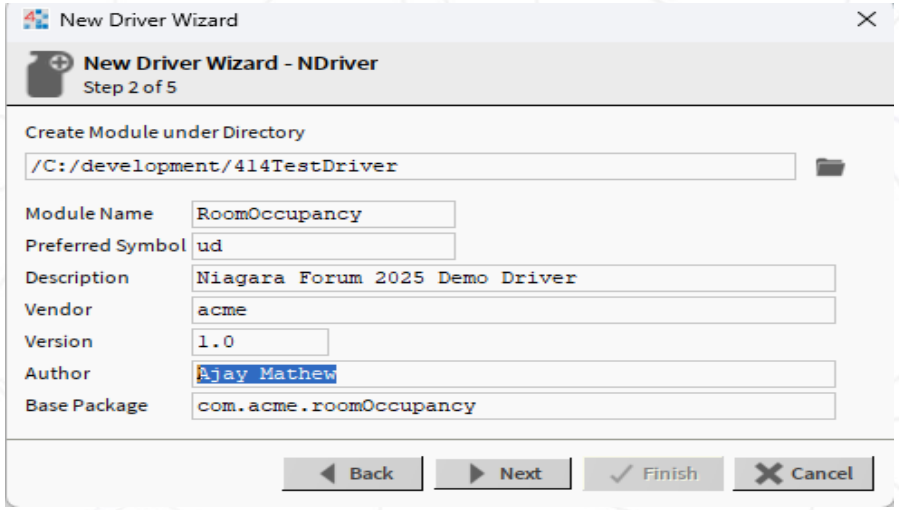
Unsolicited Messages





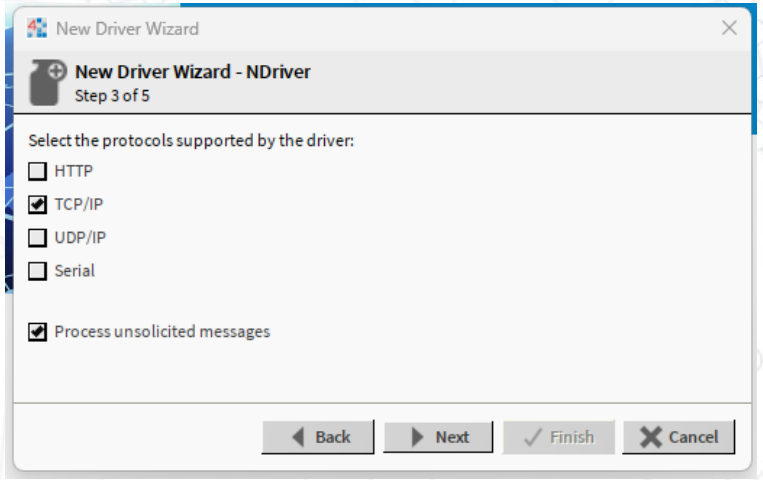
Implementing NDriver

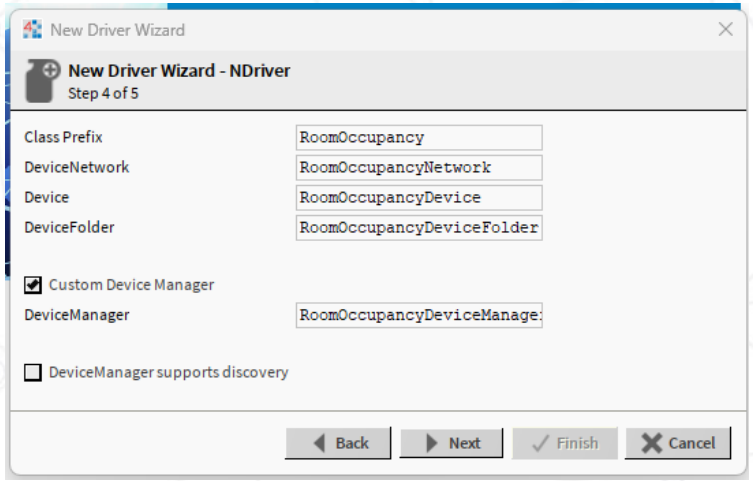
- Tools -> New Driver
- Complete Module information
  - Base Package used for creating module packages
  - Author used for auto-generated comments



- Select Protocol Transport Type(s)
- Select **Process Unsolicited Messages** to create unsolicited message handler

- Specify class names and default prefix
- Create Custom Device Manager or use Default
- Allow DeviceManager to support Discovery

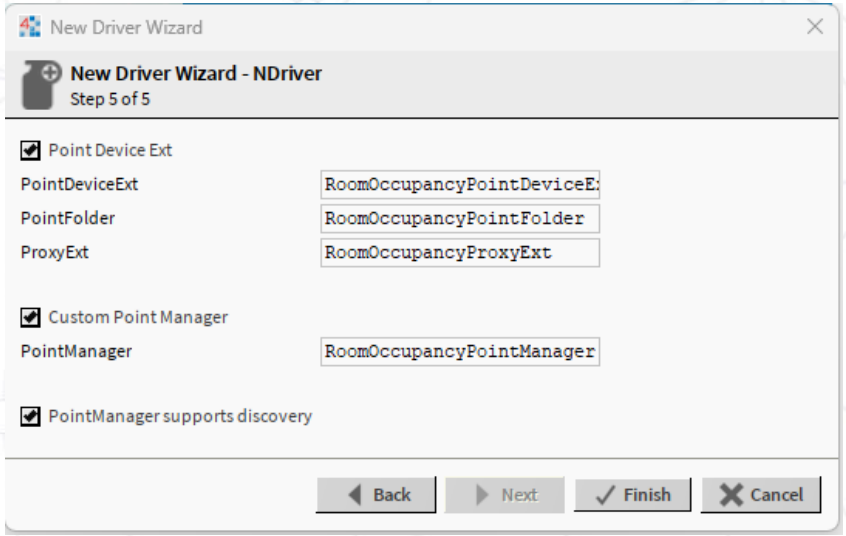




NF 25

TRIDIUM

- Select to add Point Device Extension
- Provide component names
- Use auto-generated Point Manager using AutoManager features, or check to create basic Point Manager



TRIDIUM

Logging

- Built in support for Spy and Log tracing
  - Log names created using BCommConfig **getResourcePrefix()** method
  - Log obtained from BNxxxNetwork.log() method
  - Custom logs should follow naming convention:

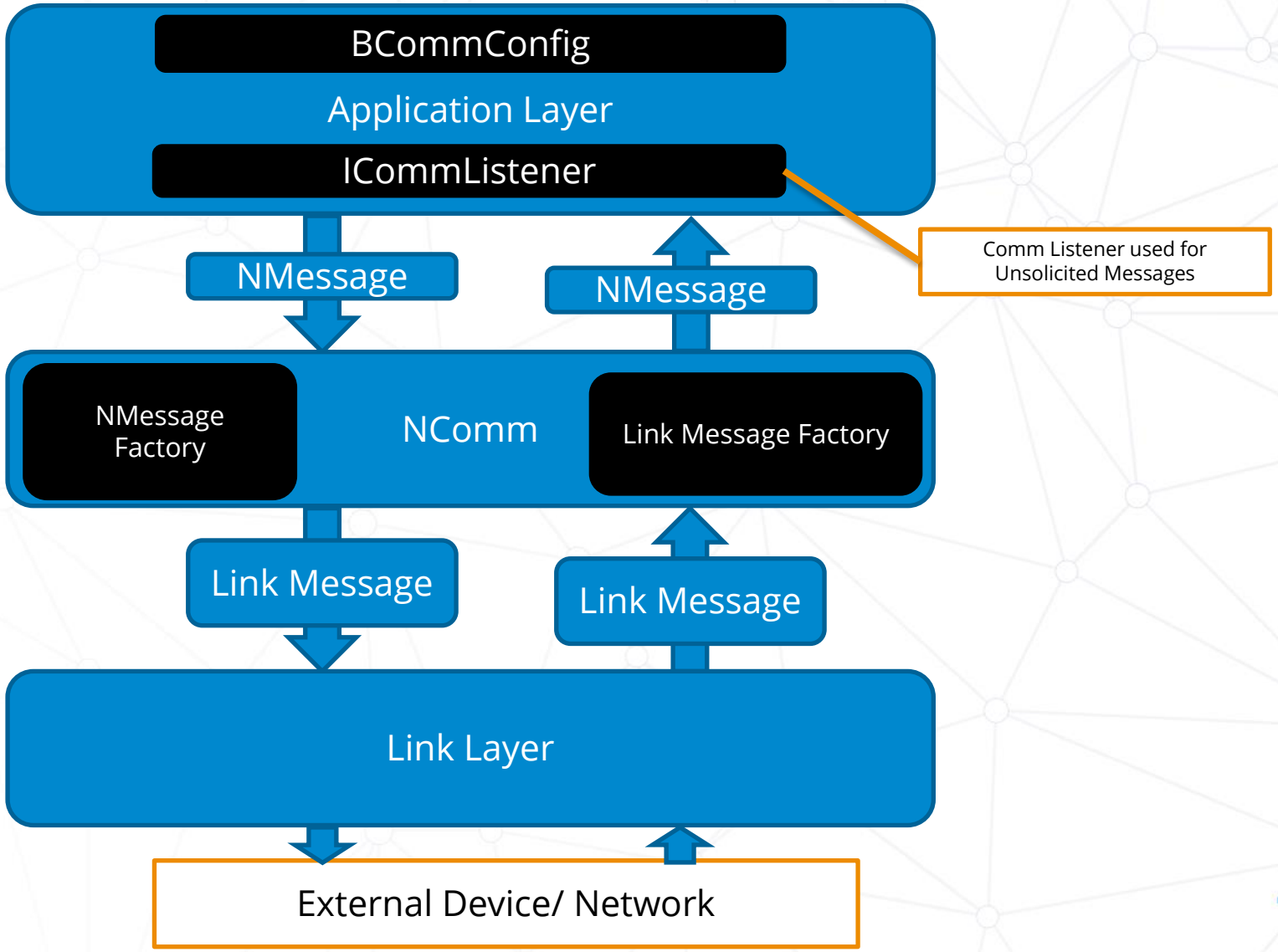
```
Logger log = Logger.getLogger(getNetworkName() + “extension”);
```

- Includes Comm tracing and Link Layer tracing
  - NMessages should override **toTraceString()** to return a trace level message for Comm trace
  - LinkLayer FINE level logging includes byte array of data read from input

Why Use NDriver

- Simplified and intuitive model with a clean API
- NDriver wizard, get started faster
- Less boiler plate code to implement
- Do not need to write your own:
  - TCP / UDP / Managing Sockets
  - Session handling
  - Retry & timeout logic
  - Fragment buffering
  - Manager views

The NDriver


BCommConfig
Application Layer
ICommListener
Comm Listener used for Unsolicited Messages
NMessage
NMessage
NMessage Factory
NComm
Link Message Factory
Link Message
Link Message
Link Layer
External Device/ Network

Developer Q&A @ 5:30pm

CONNECTING
THE WORLD