

Power Grid Communication Planning and Modeling Tool (PGCPMT) Code Generator for ns-3

Jeffrey M. Young

Background:

The PGCPMT Code Generator provides a tool that can:

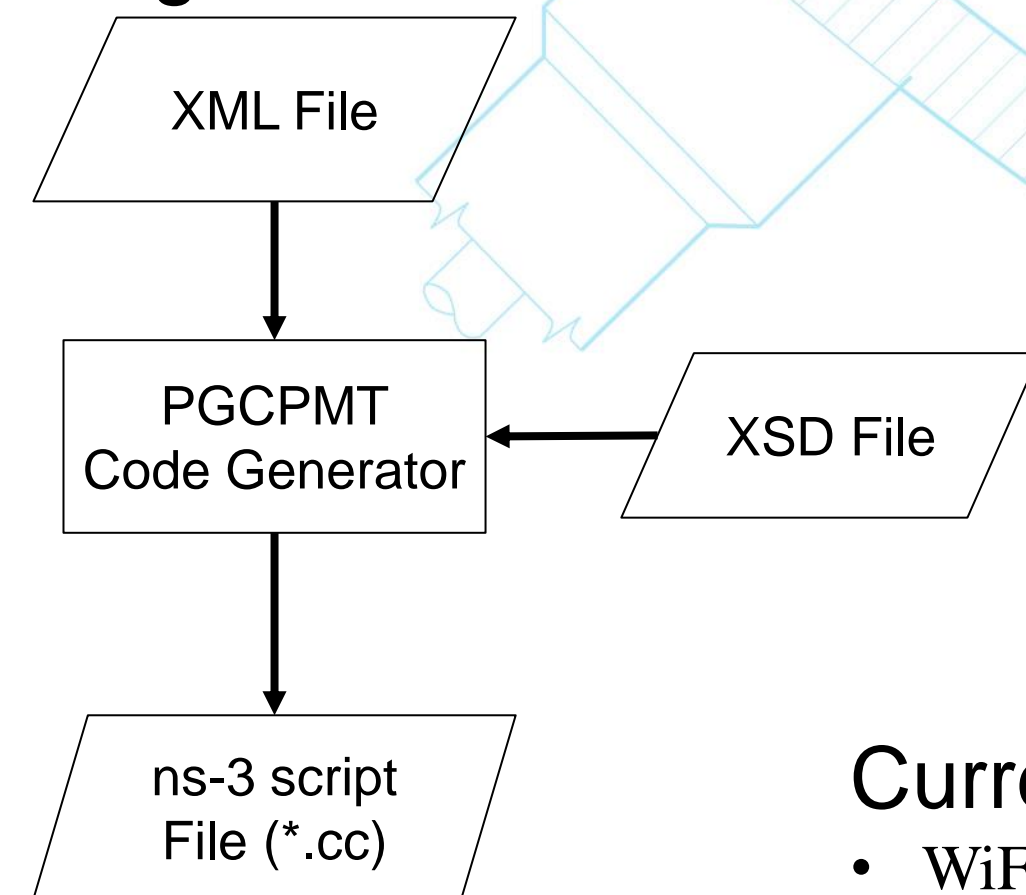
- Interpret a higher level description of a network topology
- Generate ns-3 statements that can be executed with the latest version of ns-3
- Free the user from having to know the details of the syntax needed to build and execute a simulation

Based on the ns-3 Topology Generator - Weiss and Vincent¹

Features:

- Generate ns-3 syntax from XML topology description
- Non-GUI, Command line interface
- XSD validation
- XML data interface
- Electric Smart Grid applications

Program execution:



Currently supported topologies:

- WiFi
- Point to Point
- CSMA

Network Hardware XSD definition:

```

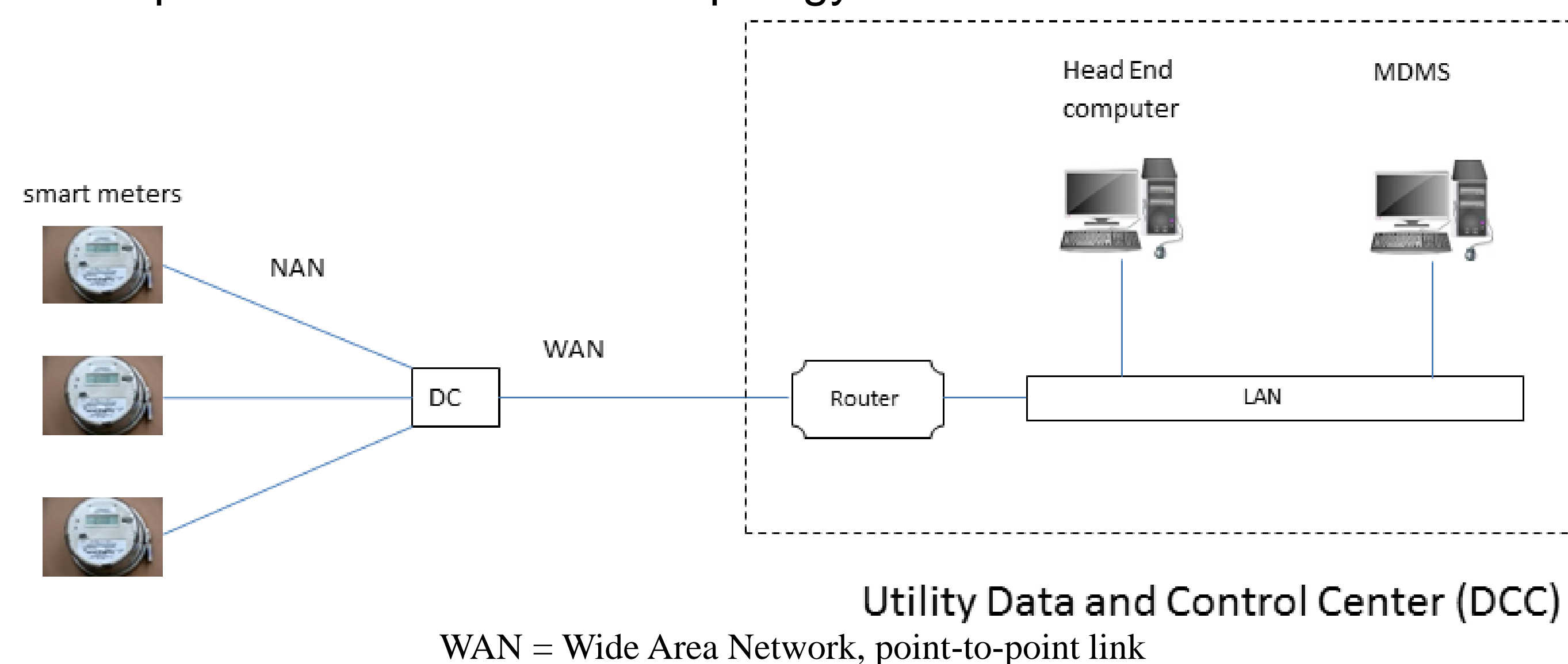
<xsd:element name="networkHardware" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="type" type="hardwareType"/>
    </xsd:sequence>
    <xsd:element name="name" type="xsd:string"/>
  </xsd:complexType>
  <xsd:element name="dataRate" type="xsd:string"/>
  <xsd:element name="linkDelay" type="xsd:string"/>
  <xsd:element name="enableTrace" type="xsd:string"/>
  <xsd:element name="mobility" type="xsd:string" minOccurs="0"/>
  <xsd:element name="connectedNodes">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="name" type="xsd:string" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
  
```

XML description for point to point link:

```

<networkHardware>
  <type>PointToPoint</type>
  <name>p2p_0</name>
  <dataRate>62200000</dataRate>
  <linkDelay>1</linkDelay>
  <enableTrace>true</enableTrace>
  <connectedNodes>
    <name>ap_0</name>
    <name>term_0</name>
  </connectedNodes>
</networkHardware>
  
```

Example Smart Grid network topology:



WAN = Wide Area Network, point-to-point link

ns-3 simulation code:

```

...
PointToPointHelper p2p_p2p_0;
p2p_p2p_0.SetDeviceAttribute("DataRate", DataRateValue(62200000));
p2p_p2p_0.SetChannelAttribute("Delay", TimeValue(MilliSeconds(1)));
...
NodeContainer all_p2p_0;
all_p2p_0.Add(ap_0);
all_p2p_0.Add(term_0);
NetDeviceContainer ndc_p2p_0 = p2p_p2p_0.Install(all_p2p_0);
...
p2p_p2p_0.EnablePcapAll("p2p_0");
...
  
```

NetworkHardware
m_type : String
m_indice : Integer
m_networkHardwareName : String
m_dataRate : String
m_networkHardwareDelay : String
m_ndcName : String
m_allNodeContainer : String
m_enableTrace : Boolean
m_nodes : String
GenerateHeader()
GenerateNodeCpp()
GenerateNetDeviceCpp()
GenerateVarsCpp()
GenerateCmdLineCpp()
GenerateTraceCpp()
GetIndice()
SetIndice()
GetNetworkHardwareName()
GetDataRate()
GetNetworkHardwareDelay()
GetNdcName()
GetInstalledNodes()
GetInstalledNode()
removeInstalledNode()
SetNetworkHardwareName()
SetDataRate()
SetNetworkHardwareDelay()
SetNdcName()
Install()
GroupAsNodeContainerCpp()
GetAllNodeContainer()
SetAllNodeContainer()
SetTrace()
GetTrace()
GetLinkType()

Node
m_indice : Integer
m_nodeName : String
m_interfaceName : String
m_type : String
GenerateHeader()
GenerateNodeCpp()
GenerateInterfaceCpp()
SetNodeName()
SetInterfaceName()
GetNodeName()
GetInterfaceName()
GetIndice()
SetIndice()
GetNodeType()

Application
m_indice : Integer
m_appName : String
m_senderNode : String
m_receiverNode : String
m_startTime : Integer
m_endTime : Integer
GenerateApplicationCpp()
GetIndice()
SetIndice()
GetAppName()
SetAppName()
GetSenderNode()
SetSenderNode()
GetReceiverNode()
SetReceiverNode()
GetStartTime()
SetStartTime()
GetEndTime()
SetEndTime()
GetEndTimeNumber()
SetEndTime()
GetApplicationType()

INL Contribution:

- New main() with
 - XSD validation
 - XML parsing
- 3.24.1 compliance
 - New syntax
 - Header file realignment
- Flowmonitor output
- Applications
 - UDP Client/Server
 - UDPCosm Client/Server
 - DataConcentrator
 - MeterDataManagement
 - DemandResponse

Acknowledgement:

This research is supported by the United States Department of Energy, Office of Electricity Delivery and Energy Reliability (DOE-OE), Under DOE Idaho Operations Office Contract DEAC0705ID14517.

For more information please contact:

Jeffrey M. Young
 Idaho National Laboratory
 2525 N. Fremont Ave.
 Idaho Falls, ID 83415
 208-526-0226, jeffrey.young@inl.gov

1. University of Strasbourg, https://www.nsnam.org/wiki/Topology_Generator