WSNet : Simulation configuration

Tutorial

Elyes Ben Hamida, ARES INRIA / CITI - INSA Lyon
Guillaume Chelius, ARES INRIA / CITI - INSA Lyon

WSNet : Simulation configuration: Tutorial

by Elyes Ben Hamida and Guillaume Chelius

Abstract

In this tutorial we will learn the basic steps necessary for setting up a configuration file.

Table of Contents

Chapter 1. Introduction

The WSNet simulator uses an xml file to configure a simulation. This file describes the simulation setup and specifies, for example, the number of nodes to simulate, the libraries used to model the radio medium and the nodes (e.g., for propagation, routing, ...). In this document, we analyse in details the architecture of a configuration file.

Chapter 2. Configuration file : architecture

The simulation configuration file is specified using the command line option **-c**. Example: **wsnet -c config.xml**. This configuration file is an xml file describing the simulation setup. It is composed of five main sections: **global parameters**, **entities**, **environment**, **bundles** and **node**.

Notes about parameter

Unit

- **time**: the time unit is the nano-second. However, a time parameter can be passed with a unit modifier as "s", "ms", "us" and "ns". Ex: **duration="200s"**, **birth="10s"**.
- **distance**: distances must be given in meters unless specified differently by the model.
- **size**: packet size must be given in B unless specified differently by the model.

Random values

Most of the parameters, i.e. distance/duration/time/destination/x/y/z, accept the "random" value. In most cases, a random value is drawn in a range chosen by the simulator as a function of the parameter type. For example, a random time will be taken between 0 and the simulation duration. A random destination will be chosen among all nodes except the node which destination is being specified.

Global parameters

Definition

The definition of a simulation is done using the following syntax:

Note

```
<simulation nodes="number-of-nodes" duration="simulation-duration" x="size" y="size" z="size" />
```

The parameters are the following:

- **nodes**: number of nodes during the simulation. This option is required and should appear only once.
- **duration**: specifies the duration of the simulation. The duration format is : **Xs**, **Xms**, **Xus**, **Xns** for second, millisecond, microsecond and nano-second, respectively. This option is required and should appear only once.
- **x**: width of the simulation area.
- **y**: length of the simulation area.
- **z**: height of the simulation area.

Example

The following example defines a simulation with 50 nodes in a 500x500x100 simulation area. The duration of the simulation is 10s.

Note

```
<simulation nodes="50" duration="10s" x="500" y="500" z="100" />
```

Entities

Definition

An entity is an instantiation of a dynamic library. The definition of an entity is done using the following syntax:

Note

```
<entity name="entity-name" library="dynamic-library-name" >
<init parameter-name="value" parameter-name="value" ... />
<default parameter-name="value" parameter-name="value" ... />
</entity >
```

The parameters are the following:

- **`name`**: the name of the entity. It is user-defined, required and should appear only once.
- **`library`**: the name of the dynamic library to instantiate. It is required and should appear only once.
- **`init`**: specifies the values for the global-parameters of the entity. It is optional and may appear only once.
- **`default`**: specifies default values for the node-parameters of the entity. It is optional and may appear only once.

A given dynamic library can be instanciated several times in different entities. Each of these entities may have different global-parameter values and node-parameter default values.

Examples

The following example defines the entity **`position`** which instanciates the library **`mobility_static`**. By default, nodes are randomly placed in x and y coordinates and with a height of 0

Note

```
<entity name="position" library="mobility_static" >
<default x="random" y="random" z="0" />
</entity >
```

The following example defines the entity **propagation** which instanciates the library **propagation_shadowing** with a pathloss of 3.

Note

```
<entity name="propagation" library="propagation_shadowing" >

<init pathloss="3" />

</entity >
```

Environment

Definition

The environment block describes the radio medium and the physical environment of the simulation. The definition of the environment is done using the following syntax **and order**.

Note

```
<environment >

<propagation entity="entity-name" range="range" />

<interferences entity="entity-name" />

<monitoring entity="entity-name" />

<modulation entity="entity-name" />

<...>

<with entity="entity-name" />

<...>

</environment >
```

The parameters are the following:

- **propagation**: the name of a propagation entity. Should appear only once.
- **interferences**: the name of an interferences entity. Should appear only once.
- **monitoring**: the name of a monitoring entity. May appear only once.
- **modulation**: the name of one modulation entity. Should appear at least once.
- **with**: the name of one environment entity. Environment entities are optional. May appear appear several times.

Example

The following example defines an environment with a shadowing propagation model, orthogonal interferences and bpsk modulation

Note

```
<environment >
<propagation entity="shadowing-propagation" range="200" />
<interferences entity="orthogonal-interferences" />
<monitor entity="nrj" />
<modulation entity="bpsk-modulation" />
<modulation entity="oqpsk-modulation" />
<with entity="fire" />
</environment >
```

Bundle

Definition

A bundle is a node architecture. The definition of a bundle is done using the following syntax:

Note

```
<bundle name="bundle-name" default="{true,false}" birth="time" /
>
<mobility entity="entity-name" />
<energy entity="entity-name" />
<antenna entity="entity-name" >
<up entity="entity-name" />
<...>
</antenna >
<with entity="entity-name" >
<default parameter-name="value" parameter-name="value" ... >
<up entity="entity-name" />
<...>
<down entity="entity-name" />
<...>
</with >
<with entity="entity-name" >
```

```
<up entity="entity-name" />

<...>

<down entity="entity-name" />

<...>

</with >

...

< /bundle >
```

The parameters are the following:

- **name**: name of the bundle. This option is required and should appear only once.
- **default**: specifies whether this bundle is the default one or not. This is an optional feature and should appear only once. Value is **true** or **false**. Default value is **false**.
- **birth**: default birth value for nodes. This is an optional feature, default value is 0.
- **mobility**: specifies which mobility entity is used for the bundle. Should appear only once.
- **energy**: specifies which battery entity is used for the bundle. May appear only once.
- **antenna**: specifies which antenna entity is used for the bundle. Should appear at least once.

The **with** option specifies the entities involved in the bundle (other than mobility, antenna and energy). This option may appear several times. **entity** represent the entity name, and:

- **up**: specifies an uplink in the bundle from the current entity to the specified one. May appear several times.
- **down**: specifies a downlink in the bundle from the current entity to the specified one.
- **default**: specifies default values for node-parameters of the entity, overriding the ones given in the entity definition. May appear only once.

Example

The following example defines a node architecture with one application, one routing protocol, one mac protocol, one radio and one antenna. It defines the uplinks and downlinks between these entities and specifies default values for some node-parameters of some entities. It also says that the node has physical sensors to measure the temperature and the light.

Note

```
<bundle name="sensor" default="true" />

<mobility entity="static" />

<antenna entity="omnidirectional" >

<up entity="radio" />
```

```
</antenna >
<with entity="radio" >
<up entity="mac" />
<down entity="antenna" />
</with >
<with entity="mac" >
<up entity="routing" />
<down entity="radio" />
</with >
<with entity="routing" >
<up entity="app" />
<down entity="mac" />
</with >
<with entity="app" >
<default period="1s" >
<down entity="routing" />
</with >
</bundle >
```

Node

Definition

A node instanciates a bundle. Definition of a node is done using the following syntax:

Note

```
<node id="node-id" as="bundle-name" birth="birth-time" >
<for entity="entity-name" parameter-name="value" ... />
<...>
</node>
```

The parameters are the following:

- **id**: node id. This option is required and should appear only once.
- **as**: bundle name which the node instantiates. This option is required and should appear only once.

- **`birth`**: birth date of node. This option is optional, default value is bundle birth value.
- **`for`**: is used to override the parameters of an entity defined in the bundle. This option is optional and may appear several times.

Not all nodes need to be defined in the config file. If not defined otherwise, a node instantiates the bundle which has the attribute **`default`** set to **`true`**. The values of the node-parameters are then the default ones as defined in the entity and bundle definitions.

Example

The following example specifies that the node whose id is 0 instantiates the bundle sensor:

Note

```
<node id="0" as="sensor" birth="10s" >

<for entity="app" period="500ms" />

</node>
```

Chapter 3. A complete example

The following example specifies a complete simulation configuration. We assume that 500 nodes are deployed randomly over an area of size 100x100x0. We use an omnidirectionnal antenna, a csma MAC protocol, a CBR application. The example can be directly downloaded from HERE [./example.xml].

Note

```
<?xml version='1.0' encoding='UTF-8'?>

<worldsens xmlns="http://worldsens.citi.insa-lyon.fr">

<simulation nodes="500" duration="10s" x="100" y="100" z="0"/>

<entity name="free-space" library="propagation_freespace">

</entity>

<entity name="interf" library="interferences_none">

</entity>

<entity name="none" library="modulation_none">

</entity>

<entity                                    name="omnidirectionnal"
library="antenna_omnidirectionnal">

<default loss="0" angle-xy="random" angle-z="random"/>

</entity>

<entity name="radio" library="radio_half1d">

<default       sensibility="-92"       dBm="10"       channel="0"
modulation="none"/>

</entity>

<entity name="mac" library="mac_dcf_802_11">

</entity>

<entity name="cbr" library="application_cbr">

</entity>

<entity name="static" library="mobility_static">

<default x="random" y="random" z="random"/>

</entity>

<environment>

<propagation entity="free-space" range="200"/>
```

```
<interferences entity="interf"/>
<modulation entity="none"/>
</environment>
<bundle name="sensor" default="true">
<mobility entity="static"/>
<antenna entity="omnidirectionnal">
<up entity="radio"/>
</antenna>
<with entity="radio">
<up entity="mac"/>
<down entity="omnidirectionnal"/>
</with>
<with entity="mac">
<up entity="cbr"/>
<down entity="radio"/>
</with>
<with entity="cbr">
<down entity="mac"/>
</with>
</bundle>
</worldsens>
```

Chapter 4. Credits

Credit for the icons goes to the Tango Desktop Project [http://tango.freedesktop.org/].