

# Distributed Programming II

A.Y. 2016/17

## Assignment n. 2 – Neo4JXML REST API Documentation

Neo4JXML provides the possibility to create a set of nodes connected by relationships. Each node can have properties (i.e. name-value pairs) and labels (i.e. strings).

### PART1: RESOURCES

Neo4JXML REST API includes the following resources:

localhost:8080/Neo4JXML/rest/	base URL of the service
localhost:8080/Neo4JXML/rest/application.wadl	WADL description of the service
localhost:8080/Neo4JXML/rest/resource/	the main resource of the service
localhost:8080/Neo4JXML/rest/resource/nodes	the set of nodes
localhost:8080/Neo4JXML/rest/resource/node/id	the node identified by id
localhost:8080/Neo4JXML/rest/resource/node/id/label	the set of labels of node id
localhost:8080/Neo4JXML/rest/resource/node/id/label/idl	the label of node id named idl
localhost:8080/Neo4JXML/rest/resource/node/id/relationship	the relationships of node named id
localhost:8080/Neo4JXML/rest/resource/relationship	the set of all relationships
localhost:8080/Neo4JXML/rest/resource/relationship/idr	the relationship named idr

### PART2: OPERATIONS

#### Create Node with properties

*Example request*

- **POST** localhost:8080/Neo4JXML/rest/resource/node/
- **Content-Type:** application/xml
- **BODY:**

```
<node xmlns="http://www.example.org/Neo4J">
  <property name="name" value="Client0"/>
</node>
```

*Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<node xmlns="http://www.example.org/Neo4J" id="15">
  <property name="name" value="Client0"/>
</node>
```

## Get Node by ID

### *Example request*

- **GET** localhost:8080/Neo4JXML/rest/resource/node/0
- **Content-Type:** application/xml

### *Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<node xmlns="http://www.example.org/Neo4J" id="0">  
  <property name="name" value="Client0"/>  
</node>
```

## Delete Node by ID

### *Example request*

- **DELETE** localhost:8080/Neo4JXML/rest/resource/node/0
- **Content-Type:** application/xml

### *Example response*

- **204:** NO CONTENT
- **Content-Type:** text/plain

## Delete all Nodes

### *Example request*

- **DELETE** localhost:8080/Neo4JXML/rest/resource/nodes
- **Content-Type:** application/xml

### *Example response*

- **204:** NO CONTENT
- **Content-Type:** text/plain

## Get all Nodes

### *Example request*

- **GET** localhost:8080/Neo4JXML/rest/resource/nodes
- **Content-Type:** application/xml

### *Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<nodes xmlns="http://www.example.org/Neo4J">
  <node id="0">
    <property name="name" value="Client0"/>
  </node>
  <node id="1">
    <property name="name" value="server0"/>
    <labels>
      <value>Server</value>
    </labels>
  </node>
</nodes>
```

## Create Label

*Example request*

- **POST** localhost:8080/Neo4JXML/rest/resource/node/15/label
- **Content-Type:** application/xml
- **BODY:**

```
<labels xmlns="http://www.example.org/Neo4J">
  <value>Client</value>
</labels>
```

*Example response*

- **204:** No Content
- **Content-Type:** text/plain

## Get Label by Node ID

*Example request*

- **GET** localhost:8080/Neo4JXML/rest/resource/node/15/label

*Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<label xmlns="http://www.example.org/Neo4J">
```

```
<value>Client</value>
</label>
```

## Delete Label by ID

*Example request*

- **DELETE** localhost:8080/Neo4JXML/rest/resource/node/15/label/Client

*Example response*

- **204:** NO CONTENT
- **Content-Type:** text/plain

## Create Relationship connecting a Node to another Node

*Example request*

- **POST** localhost:8080/Neo4JXML/rest/resource/node/15/relationship
- **Content-Type:** application/xml
- **BODY:**

```
<relationship dstNode="2" type="Link" xmlns="http://www.example.org/Neo4J" />
```

*Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<relationship xmlns="http://www.example.org/Neo4J" dstNode="2" type="Link"
srcNode="15" id="3"/>
```

## Get Relationship by ID

*Example request*

- **GET** localhost:8080/Neo4JXML/rest/resource/relationship/3

*Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<relationship xmlns="http://www.example.org/Neo4J" dstNode="2" type="Link"
srcNode="15" id="3"/>
```

## Delete Relationship by ID

*Example request*

- **DELETE** localhost:8080/Neo4JXML/rest/resource/relationship/3

*Example response*

- **204:** NO CONTENT
- **Content-Type:** text/plain

## Get the Paths that connect a Node to another Node with given Node ID

*Example request*

- **GET** localhost:8080/Neo4JXML/rest/resource/node/0/paths?dst=2

*Example response*

- **200:** OK
- **Content-Type:** application/xml
- **BODY:**

```
<paths xmlns="http://www.example.org/Neo4J">
  <path dstNode="2" srcNode="0" length="1">
    <node>0</node>
    <node>2</node>
    <relationship>0</relationship>
  </path>
</paths>
```