Distributed System Project

Project Exercise 1

Write a program to Implements the bully election algorithm

UserId : msingh
Name : Maninder Pal Singh
Student Number: 014343397

Introduction

A bully election algorithm is implemented using a Java Program. The algorithm can run on Nodes of the Ukko Cluster as well as Local System.

The bully election algorithm is used to dynamically electing a coordinator by ID number. The node with the highest ID number is selected as the coordinator.

The Nodes other than coordinator keep on pinging to the coordinator to check the state of coordinator. If any Node discovers, the current coordinator is down because of message timeouts, it performs the election in a system to find out new coordinator in a system.

The Node who discovers, the current coordinator is in down state, the Node broadcasts an election message (inquiry) to all other Nodes. If the Node hears from highest Index ID Node, the Node will drop an Election and the Other Node with highest index ID will continue the Election process in a System. If the Node does not hear from any other higher ID Node, it will win the Election and declare itself as a new Coordinator for the System. The process of Election continues till the Highest ID Node declares itself as new Coordinator for the System.

Assumptions

- Message delivery between processes should be reliable
- Prior information about other Node id's must be known
- All the IP address mention in the configuration_file are unique.
- Election initiated by only single node. When the Node send message to other nodes, Other Node who receive message whose id is greater than current Node, will bully the existing Election and start a fresh Election. In a system, only single election is going on at any moment.
- Server Node is an alias used for Coordinator of the System. Coordinator is a current Leader for the System consists of number of nodes.
- The system used predefined set of nodes, and other nodes in a system are known.

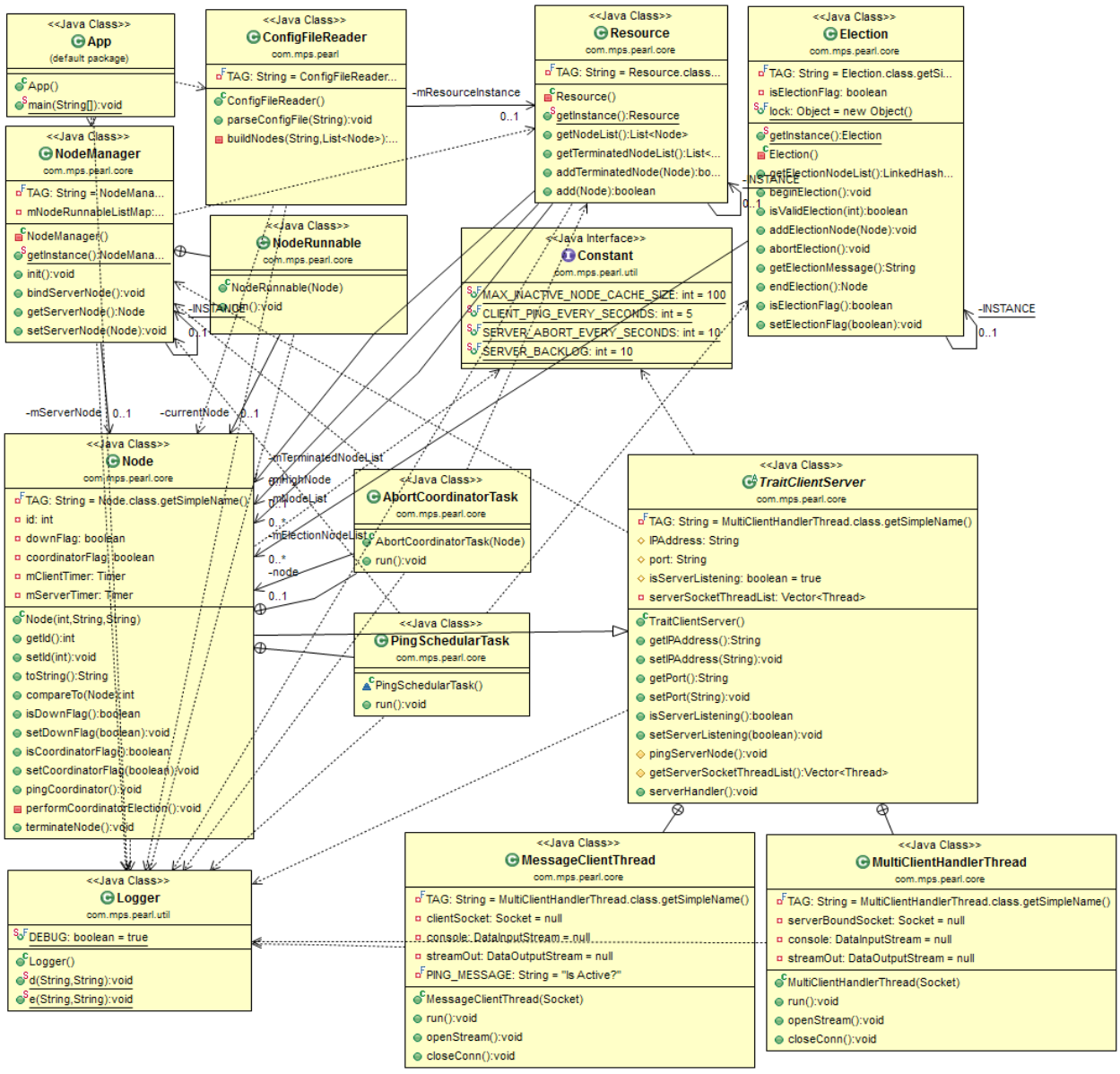
Project Structure

BullyElectionAlgorithmVer1/
----- ----- ----- ----- ----- ----- src
----- ----- ----- ----- ----- ----- configuration_file.txt
----- ----- ----- ----- ----- ---------- ----- {Java Source Code}
----- ----- ----- ----- ----- ----- classDiagram
----- ----- ----- ----- ----- ----- ----- ----- {Class Diagram of the Code}
----- ----- ----- ----- ----- ----- doc
----- ----- ----- ----- ----- ----- ----- ----- {Java API Documentation}
----- ----- ----- ----- ----- ----- readme.txt
----- ----- ----- ----- ----- ----- sources.txt

Class Diagram

The Java API documentation has been added into the project folder to understand the structure of the Project. The documentation can be obtained in the following project path:

BullyElectionAlgorithmVer1\doc\index.html



Compile and Run

App.java file is entry point for the App. Run the program using App.java file. Before running the source code, compile all java files into bytecode.

Step 1>> Go to Project source folder, and perform following commands to compile the source code.

 # Linux

$ find -name "*.java" > sources.txt

$ javac @sources.txt

{ sources .txt will contain a list of all the files in a project folder }

Step 2>> Run the Application Code::

$ java App configuration_file.txt

The App.java is the entry point for the Application. The Struture of the Command should be ::

<java> <App> <Configuration file name>

Note :: the configuration file should be under src folder of the Project Source code

Configuration

This App implements Bully Election Algorithm. App.java file is entry point for the App. Run the program using App.java file. Before running the source code, compile all java files into bytecode.

Configure Nodes for the App in a configuration_file.txt file.

Add the nodes in a file in the following format:

<Index> <IP Address> <Port>

eg: 12345 192.168.123.234 4457

Advance Debugging

Logger.java helps to perform advance debugging and see the calls proceeding between systems. Enable the DEBUG flag to perform advance debugging of the App.

```
public static final boolean DEBUG = true;
```