Distributed Systems, Advanced Course
Project Report

KTH Royal Institute of Technology
School of Information and Communication Technology
Student:Fanti Machmount Al Samisti (fmas@kth.se)
Student:Pradeep Perris (weherage@kth.se)

March 3, 2016

Contents

1 Introduction **2**

2 Design Overview **2**
2.1 System Component 2
2.2 Node Toplogy 4

3 System Abstraction and Implmentation **4**
3.1 Perfect Point to Point Link 4
3.2 Best Effort Broadcast 4
3.3 (N,N) Atomic Registry 4
3.4 Reconfiguration 5

4 System Simulations and Scenarios **5**

5 Conclusions **5**

1 Introduction

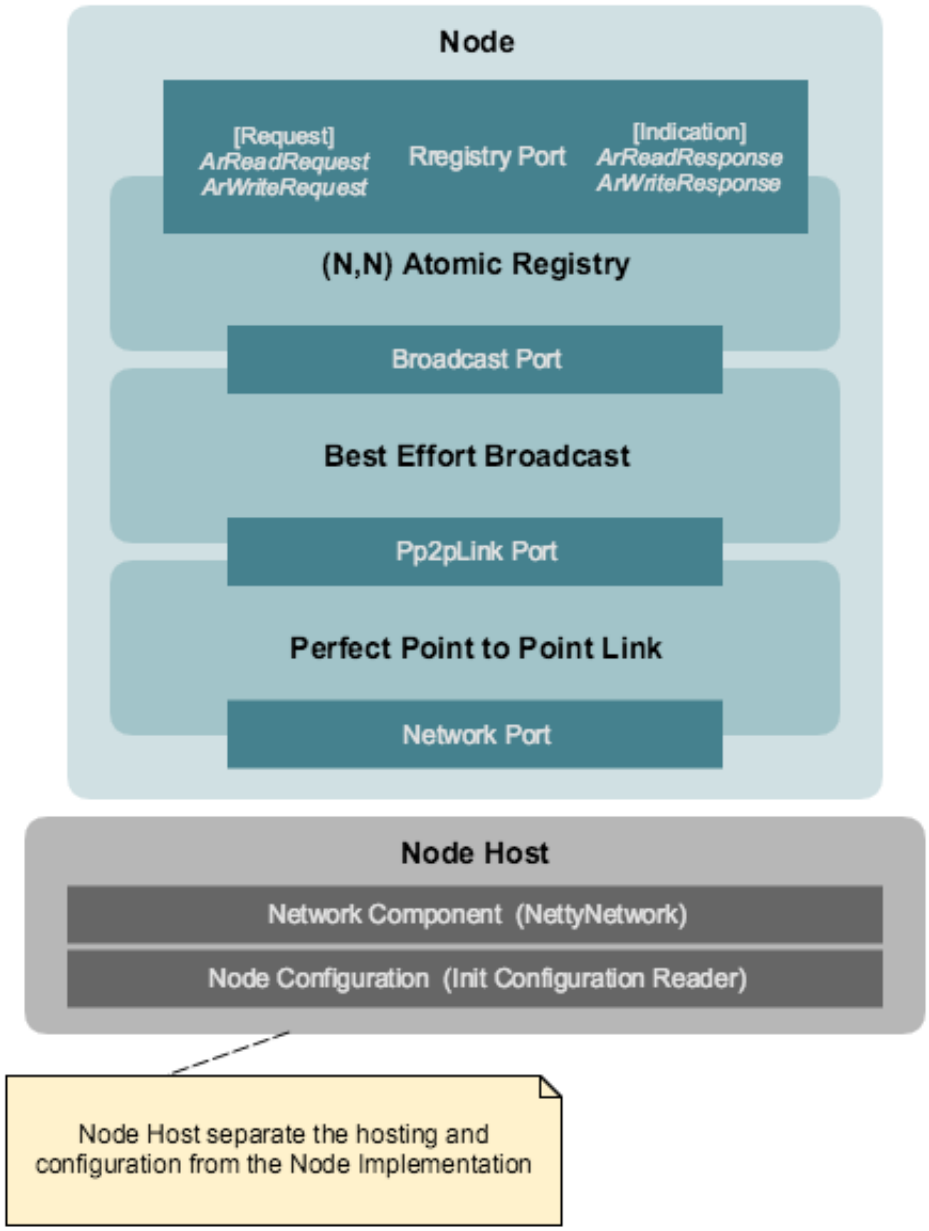
The goal of this project is to design and implement a distributed key-value store in Kompics. We have used well known distrubuted abstraction model to achive this task:

Our model employs a (static) membership protocol. Data is partisioned using a Hashing function and replicated within each membership nodes. The data consitency among the replicas during in both reads and updates are achived with (N,N) Atomic Register.

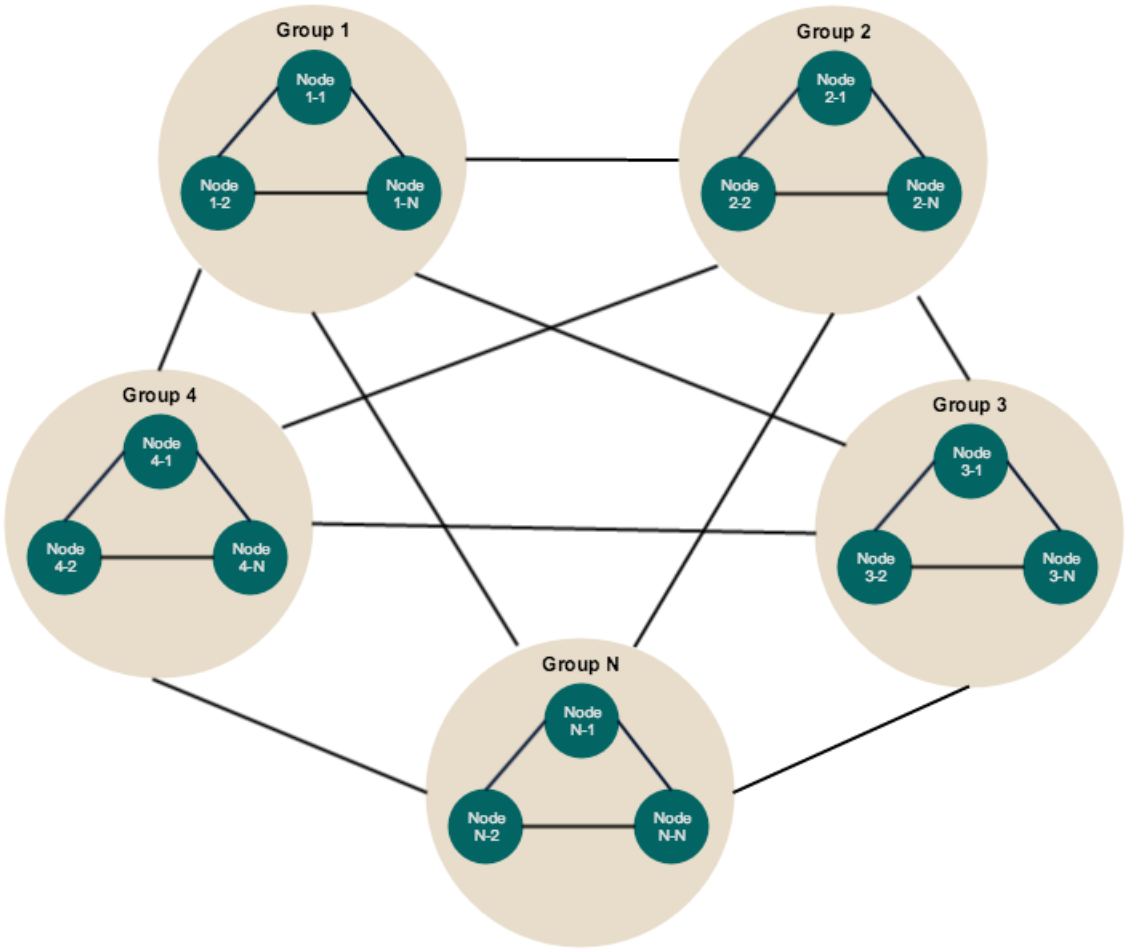
2 Design Overview

2.1 System Component

The following figure depicts the overall design of the system.


Node
[Request]
ArReadRequest
ArWriteRequest
Rregistry Port
[Indication]
ArReadResponse
ArWriteResponse
(N,N) Atomic Registry
Broadcast Port
Best Effort Broadcast
Pp2pLink Port
Perfect Point to Point Link
Network Port
Node Host
Network Component (NettyNetwork)
Node Configuration (Init Configuration Reader)
Node Host separate the hosting and configuration from the Node Implementation

2.2 Node Toplogy



3 System Abstraction and Implmentation

The report should not be too long (≈ 2-3 pages).

3.1 Perfect Point to Point Link

The report should not be too long (≈ 2-3 pages).

3.2 Best Effort Broadcast

The report should not be too long (≈ 2-3 pages).

3.3 (N,N) Atomic Registry

The report should not be too long (≈ 2-3 pages).

3.4 Reconfiguration

The report should not be too long (≈ 2-3 pages).

4 System Simulations and Scenarios

The report should not be too long (≈ 2-3 pages).

5 Conclusions

The report should not be too long (≈ 2-3 pages).

What have you learnt from the problem presented? Was it useful?