

A-SMGCS 시스템 노드 간 공유 데이터 동기화 및 변경 통지 방법에 관한 연구

Shared Data Synchronization and Change Notification between A-SMGCS System Node

강호영*, 이석찬, 신용학
LS산전(주)

초 록

A-SMGCS 시스템은 HMI 단말 노드, 외부 시스템 연계 노드, 운영 데이터베이스 노드, 서버 노드 등 여러 노드로 구성되어 있으며 이 노드들이 협업하여 운영자에게 원활한 서비스를 제공하기 위해서는 최소한의 공유 데이터가 서로 동기화되고 변경 발생 시 해당 노드에 자동으로 통지될 수 있어야 한다. 본 연구에서는 A-SMGCS 시스템 운영에 필요한 최소한의 공유 데이터 동기화 및 변경 통지 기능 제공을 위해 협업 미들웨어인 ZooKeeper를 활용하고자 한다. 본 연구를 통해 클러스터를 구성하는 복수개의 ZooKeeper에 공유 데이터를 계층적으로 저장하고 특정 데이터에 변경이 발생할 시 자동으로 A-SMGCS 시스템 노드들에 통지가 됨을 확인하였다. 이러한 기능은 A-SMGCS 시스템뿐만 아니라 시스템 노드 간 신뢰성 있는 공유 데이터 실시간 동기화 필요한 다양한 시스템에도 쉽게 적용이 가능하다.

1. 서 론

A-SMGCS 시스템은 최신 기술을 사용하여 다양한 운영 기능의 고수준 통합을 통해 특정한 시정 조건, 교통 밀도, 공항 레이아웃 상에서 적절한 항공 수용력 및 안전성을 제공하는 시스템이다[1].

A-SMGCS 시스템은 HMI 단말 노드, 외부 시스템 연계 노드, 운영 데이터베이스 노드, 서버 노드 등 여러 노드로 구성되어 있으며 이 노드들이 협업하여 운영자에게 원활한 서비스를 제공하기 위해서는 최소한의 공유 데이터가 서로 동기화되고 변경 발생 시 해당 노드에 자동으로 통지될 수 있어야 한다.

본 논문은 A-SMGCS 시스템을 구성하는 노드들이 공유 데이터를 신뢰성 있게 실시간 동기화하는 방법에 관한 것으로 오픈소스 소프트웨어인 ZooKeeper[2]를 활용하여 3개의 공유 데이터 관리 노드로 클러스터를 구성하고 이를 바탕으로 공유 데이터를 계층적으로 저장하여 특정 데이터에 변경이 발생할 시 자동으로 A-SMGCS 시스템 노드들에 실시간 통지가 되도록 확인하였다.

본 연구 결과는 A-SMGCS 시스템뿐만 아니라

시스템 노드 간 신뢰성 있는 공유 데이터 실시간 동기화 필요한 다양한 시스템에도 쉽게 적용이 가능할 것으로 기대한다.

2. 관련 연구

공유 데이터 실시간 동기화와 관련된 오픈소스 소프트웨어에 대해 살펴본다.

ZooKeeper는 분산 어플리케이션을 운용할 때 필요한 네이밍, 구성 관리, 동기화, 그룹 서비스 등을 제공하는 협업 미들웨어이다[2]. Apache 재단이 관리하는 여러 오픈소스 소프트웨어 중 Hadoop[3] 또는 Storm[4]과 같은 분산 처리, Cassandra[5] 또는 MongoDB[6]와 같은 NoSQL 데이터베이스를 운용하기 위해서는 분산형 구성 서비스, 동기 서비스 및 대용량 분산 시스템을 위한 네이밍 서비스 등을 필요로 하는데 이러한 기능을 제공하는 것이 ZooKeeper이다. 특히 ZooKeeper는 복제 서비스를 이용한 고가용성을 제공하여 신뢰성 있는 공유 데이터 실시간 동기화가 가능하다.

3. 본 론

3.1 공유 데이터 실시간 동기화 시나리오

A-SMGCS 시스템을 구성하는 각 노드가 서로 최소한의 데이터를 공유하고 이 데이터에 변경이 발생했을 때 실시간으로 통지되어 전체 시스템을 동기화하는 공유 데이터 실시간 동기화 시나리오는 다음과 같다.

① A-SMGCS 시스템을 구성하는 노드는 크게 서버 그룹과 클라이언트 그룹으로 구성되어 있으며 서버 그룹은 GCSR, ODB, FWS, MVS, ILCMS 서버로 이루어진다. 각 서버는 이중화를 위해 물리적으로 2개의 노드로 나누어진다. 클라이언트 그룹은 CWP와 MWP로 이루어지며 개수의 제한이 없으며 이중화 대상은 아니다.

② A-SMGCS 노드들이 공유하는 최소한의 데이터는 IP 주소이다. 각 서버는 이중화된 2개의 IP 주소 리스트와 Active된 서버 IP가 공유되며 클라이언트는 클라이언트의 IP 리스트가 공유된다.

③ A-SMGCS 노드들이 공유하는 최소한의 데이터는 ZooKeeper 상에 저장되어 있고 각 노드들을 이를 Watch하고 있다.

④ A-SMGCS 노드들 중 특정 노드가 ZooKeeper에 저장된 특정 데이터를 변경할 경우 실시간으로 모든 노드에 통지가 된다.

⑤ A-SMGCS 각 노드들은 ZooKeeper로부터 통지 메시지를 받으면 해당 변경된 노드 정보를 확인하고 각 노드의 운영상 필요로 하는 액션을 수행하거나 상태 정보를 변경한다.

3.2 공유 데이터

A-SMGCS 시스템을 구성하는 각 노드가 서로 공유되어야 하는 데이터는 표 1과 같다. 이 공유 데이터는 변경 발생 시, 실시간으로 특정 노드(해당 데이터를 Watch하고 있는 노드)에 실시간으로 알려줄 수 있어야 한다.

Table 1. A-SMGCS Node Sharing Data

Group		Data Example
SERVER	GCSR	IPList 192.168.0.1,192.168.0.2
		Active 192.168.0.1
	ODB	IPList 192.168.0.3,192.168.0.4
		Active 192.168.0.3
	FWS	IPList 192.168.0.5,192.168.0.6
		Active 192.168.0.5
	MVS	IPList 192.168.0.7,192.168.0.8
		Active 192.168.0.7
	ILCMS	IPList 192.168.0.9,192.168.0.10
		Active 192.168.0.9
CLIENT	CWP	IPList 192.168.0.11,192.168.0.12
	MWP	IPList 192.168.0.13,192.168.0.14

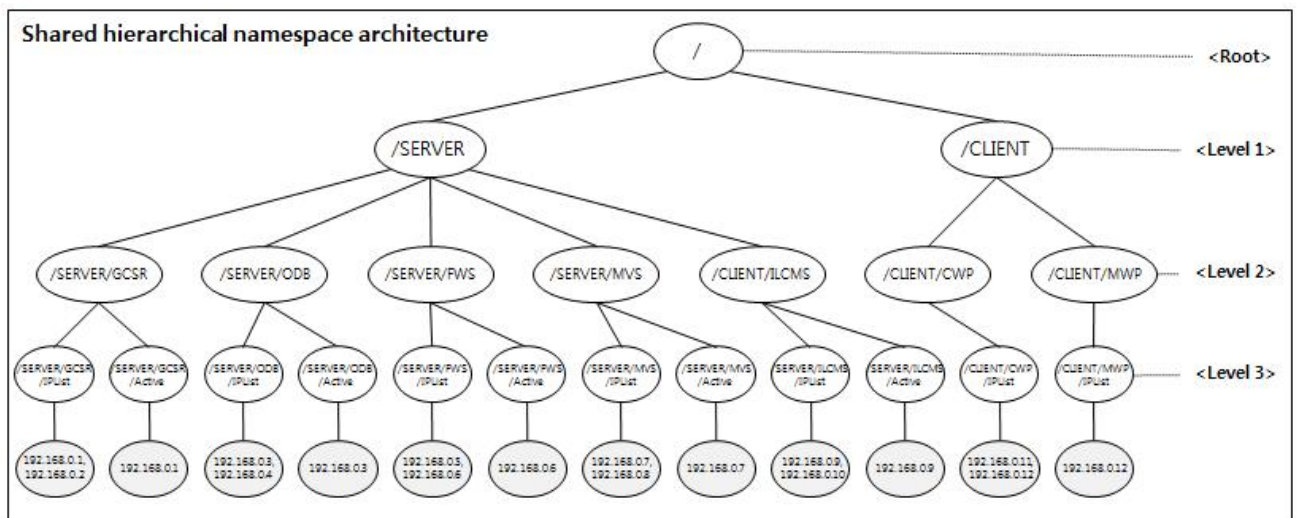


Fig 1. A-SMGCS Node Sharing Data in ZooKeeper

공유 데이터는 협업 미들웨어인 ZooKeeper의 저장 방식인 계층 구조(Shared Hierarchical Data Architecture) 형태로 저장되어야 하므로 ZooKeeper의 저장 형태로 공유 데이터를 도식화 하면 그림 1과 같다.

3.3 시스템 구성

A-SMGCS 시스템을 구성하는 노드들의 구성을 도식화하면 그림 2와 같다. A-SMGCS 시스템은 최소한의 노드들이 공유해야 하는 데이터를 저장하는 ZooKeeper 클러스터와 A-SMGCS 기능을 제공하는 A-SMGCS 시스템 노드들로 구성된다. ZooKeeper 클러스터는 3개의 노드로 구성되며 공유 데이터들이 서로 복제되므로 1개의 노드가 실패되어도 전체 ZooKeeper 클러스터의 공유 데이터는 안전한 상태이므로 신뢰성을 유지할 수 있다. ZooKeeper가 설치되는 노드들의 하드웨어는 표 2와 같으며 높은 사양을 요구하지 않는다.

3.4 공유 데이터 생성

ZooKeeper 클러스터 노드에 공유 데이터를 그림 1과 같은 구조로 생성한다. 생성한 계층적 구조 데이터는 그림 3과 같이 확인할 수 있다.

Table 2. ZooKeeper Node H/W Spec.

Item	Description
CPU	Samsung Exynos5422 Cortex™-A15 1.8Ghz quad core and Cortex™-A7 quad core CPUs.
RAM	2Gbyte LPDDR3 RAM at 933MHz (14.9GB/s memory bandwidth) PoP stacked
USB	USB 3.0 Host x 1, USB 3.0 OTG x 1, USB 2.0 Host x 4
HDMI	HDMI 1.4a for display

```
[zk: localhost:2181(CONNECTED) 65] ls /
[zookeeper, SERVER, CLIENT]
[zk: localhost:2181(CONNECTED) 66] ls /SERVER
[GCSR, FWS, MVS, ODB, ILCMS]
[zk: localhost:2181(CONNECTED) 67] ls /SERVER/GCSR
[IPList, Active]
[zk: localhost:2181(CONNECTED) 68] ls /SERVER/ODB
[IPList, Active]
[zk: localhost:2181(CONNECTED) 69] ls /SERVER/FWS
[IPList, Active]
[zk: localhost:2181(CONNECTED) 70] ls /SERVER/MVS
[IPList, Active]
[zk: localhost:2181(CONNECTED) 71] ls /SERVER/ILCMS
[IPList, Active]
[zk: localhost:2181(CONNECTED) 72] ls /CLIENT
[CWP, MWP]
[zk: localhost:2181(CONNECTED) 73] ls /CLIENT/CWP
[IPList]
[zk: localhost:2181(CONNECTED) 74] ls /CLIENT/MWP
[IPList]
[zk: localhost:2181(CONNECTED) 75]
```

Fig 3. Node Data Structure in ZooKeeper

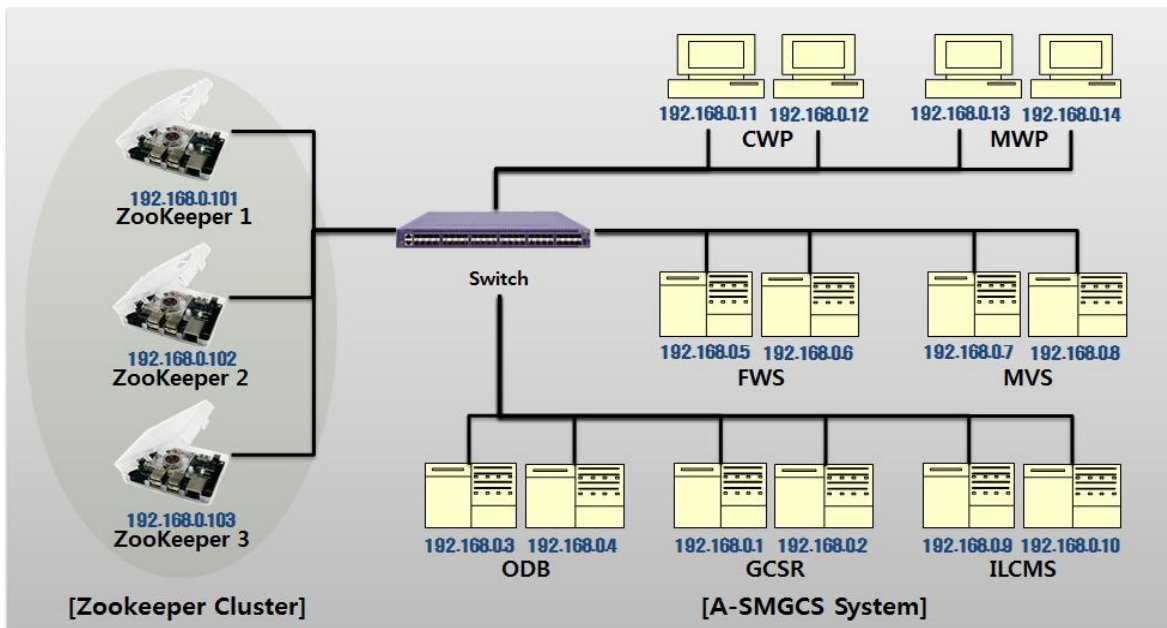


Fig 2. A-SMGCS System Nodes

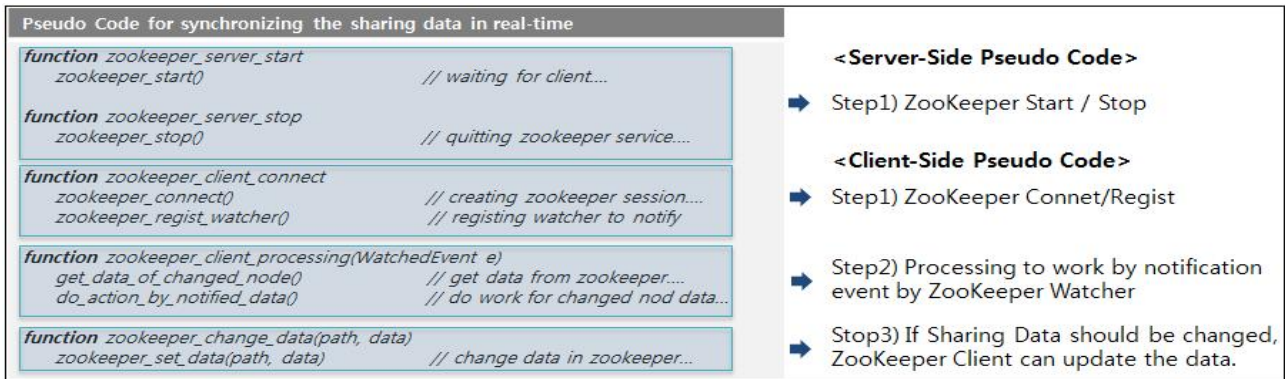


Fig 4. Pseudo Code

3.5 동기화 기능 구현

A-SMGCS 시스템은 C++로 구현되나 본 논문은 ZooKeeper의 적용 가능성을 빠르게 확인하기 위해 ZooKeeper 클라이언트는 Java로 구현하였다. 그림 4는 구현된 Java 클라이언트의 의사 코드(Pseudo Code)이다. 의사 코드는 2가지 부분으로 나눌 수 있다. 첫 번째는 Zookeeper 서비스를 시작하거나 중지하는 부분으로 이는 ZooKeeper 클러스터 노드에서 실행되는 코드이다. 두 번째 부분은 ZooKeeper 클라이언트가 ZooKeeper 서버에 연결하고 실시간으로 변경된 데이터를 통지받기 위해 특정 공유 데이터에 Watcher를 등록하는 코드이다. 세 번째 부분은 ZooKeeper 서버로부터 통지받은 변경 데이터를 처리하는 부분으로 두 번째 부분에서 등록한 데이터에 대해 통지 메시지를 받고 이를 바탕으로 적절한 액션을 수행하는 코드이다. 마지막 부분은 개별 A-SMGCS 시스템 노드에 설치된 ZooKeeper 클라이언트가 특정 공유 데이터를 변경하는 의사 코드이다.

후 기

본 연구는 국토교통부 항공안전기술개발사업 연구비지원(15ATRP-C069188-03)에 의해 수행되었습니다.

참고문헌

- [1] ICAO, 2004, Advanced Surface Movement Guidance and Control Systems (A-SMGCS) Manual, Doc 9830 AN/452, ICAO, Canada, pp. 1-1~5-14.
- [2] Flavio Junqueira, Benjamin Reed, "ZooKeeper Distributed Process Coordination", 2013, O'REILLY Media, USA
- [3] Hadoop, <https://hadoop.apache.org/>
- [4] Storm, <https://storm.apache.org/>
- [5] Cassandra, <http://cassandra.apache.org/>
- [6] MongoDB, <https://www.mongodb.org/>

4. 결 론

본 연구에서는 A-SMGCS 시스템의 각 노드가 필요로 하고 공유되어야 하는 정보를 정의하고 분산 협업 미들웨어인 ZooKeeper를 활용하여 신뢰성 있는 공유 데이터 동기화 기능을 구현하고 적용 가능성을 확인하였다. 향후 후속 연구에서 실제 현장 적용을 위해 성능, 안전성 등 추가적인 비기능 시험을 통해 보완할 계획이다.