

使用Zookeeper改造舊系統

Matt @ JCCConf 2016



About me

- ▶ 何適宇 <Matt>
 - ▶ methodho@gmail.com
 - ▶ <https://github.com/methodho>
 - ▶ <https://www.facebook.com/evermatt>
- ▶ 松凌科技 <<http://www.softleader.com.tw/>>



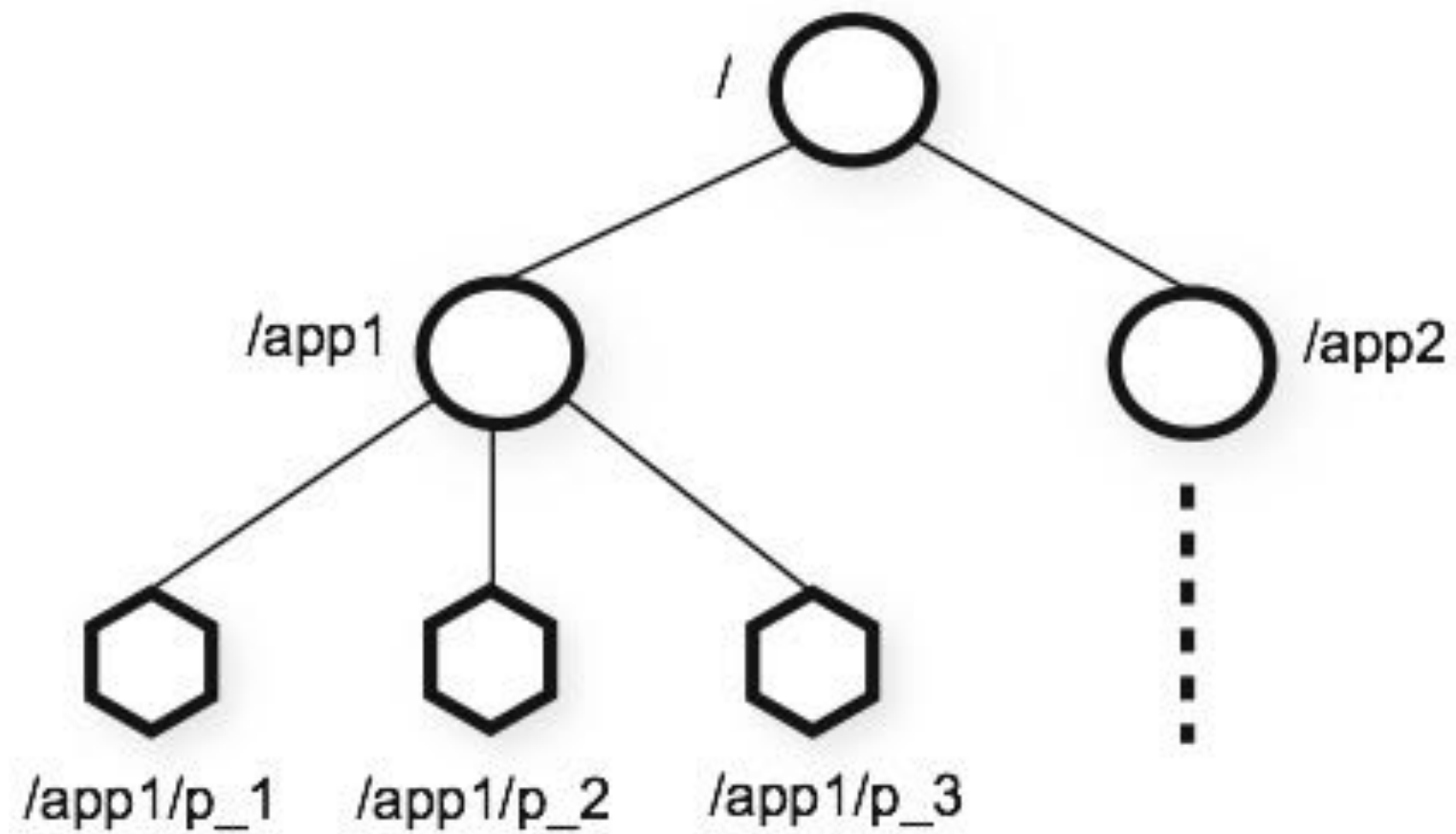
大綱

- ▶ ZooKeeper 介紹
- ▶ 使用 ZooKeeper 前的方法
- ▶ Java programming to ZooKeeper
 - ▶ o.a.z.ZooKeeper (3.4.9)
 - ▶ o.a.c.f.r.l.LeaderLatch (2.11.0)

**Because coordinating
distributed systems is a Zoo**

Apache ZooKeeper wiki

WHAT IS ZOOKEEPER



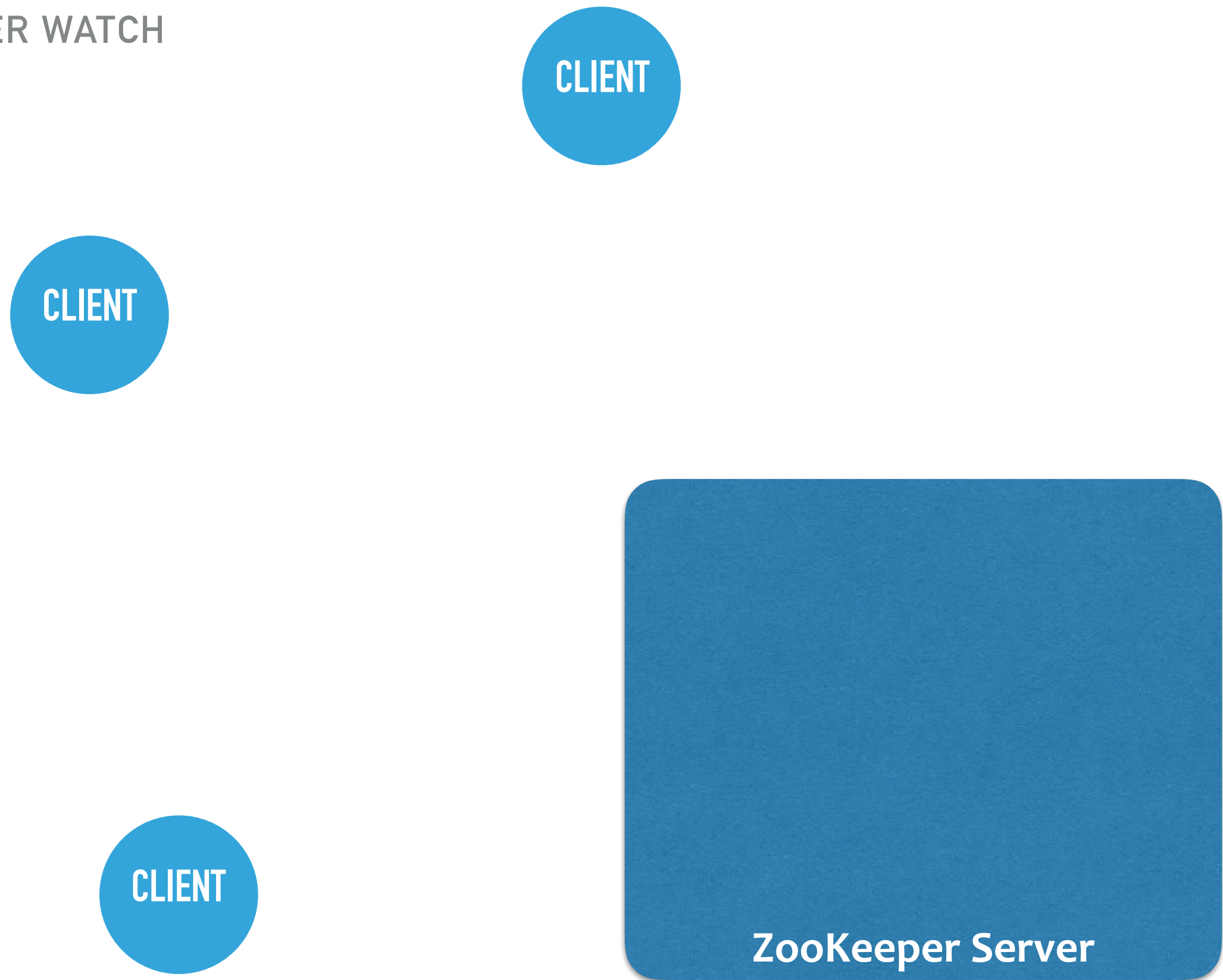
<https://zookeeper.apache.org/doc/current/zookeeperOver.html>

- ▶ 啟動server
 - ▶ `/bin/zkServer.sh start`
- ▶ 連結到 zookeeper server
 - ▶ `/bin/zkCli.sh [-server localhost:2181]`
- ▶ 關閉server
 - ▶ `/bin/zkServer.sh stop`

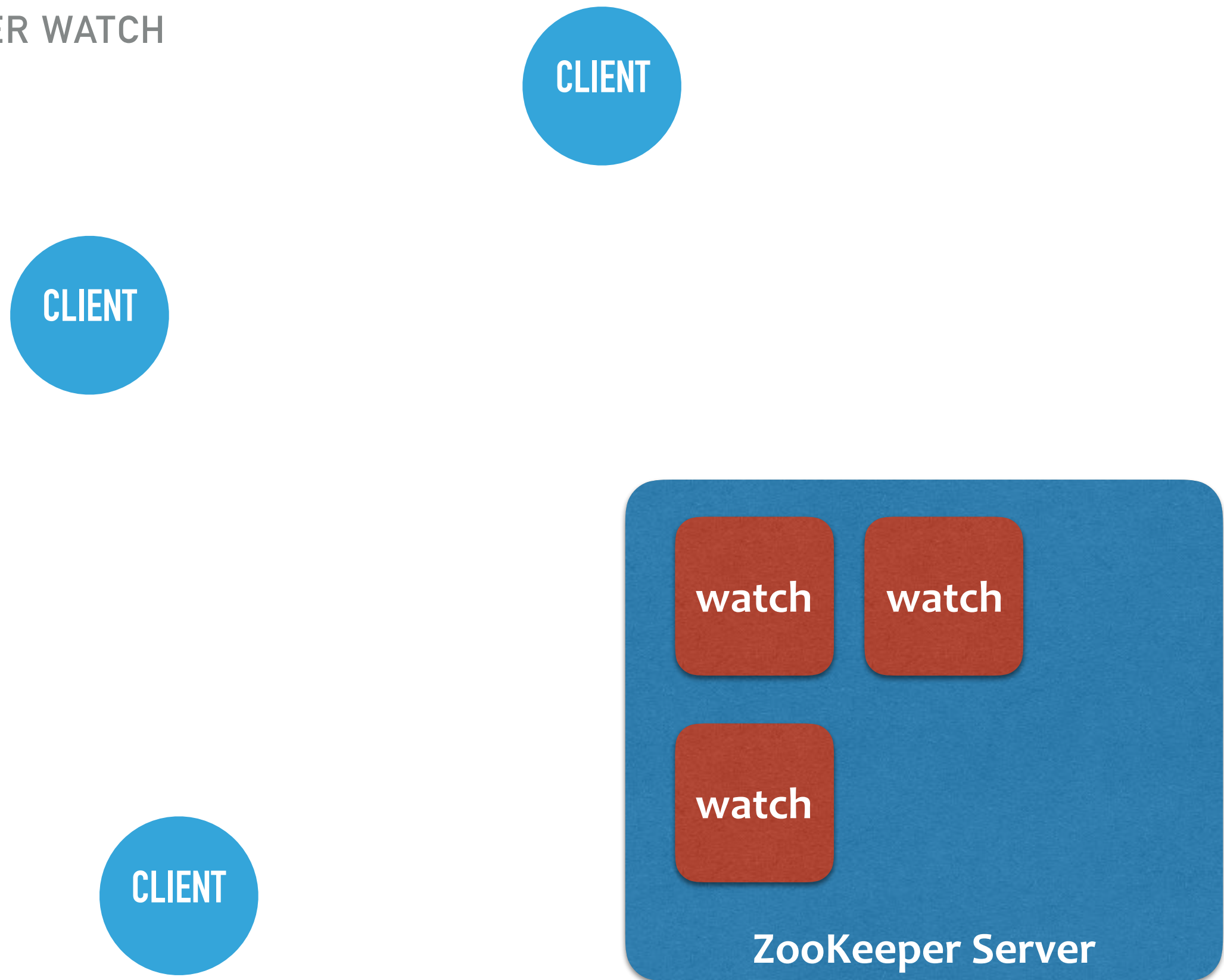
- ▶ 新增 znode
 - ▶ create [-s] [-e] path data
- ▶ 刪除 znode
 - ▶ delete path
 - ▶ rmr path

- ▶ 列出 znode 下的 children
 - ▶ ls path [watch]
 - ▶ ls2 path [watch]
- ▶ 讀寫 znode data (byte[])
 - ▶ set path data [watch]
 - ▶ get path [watch]

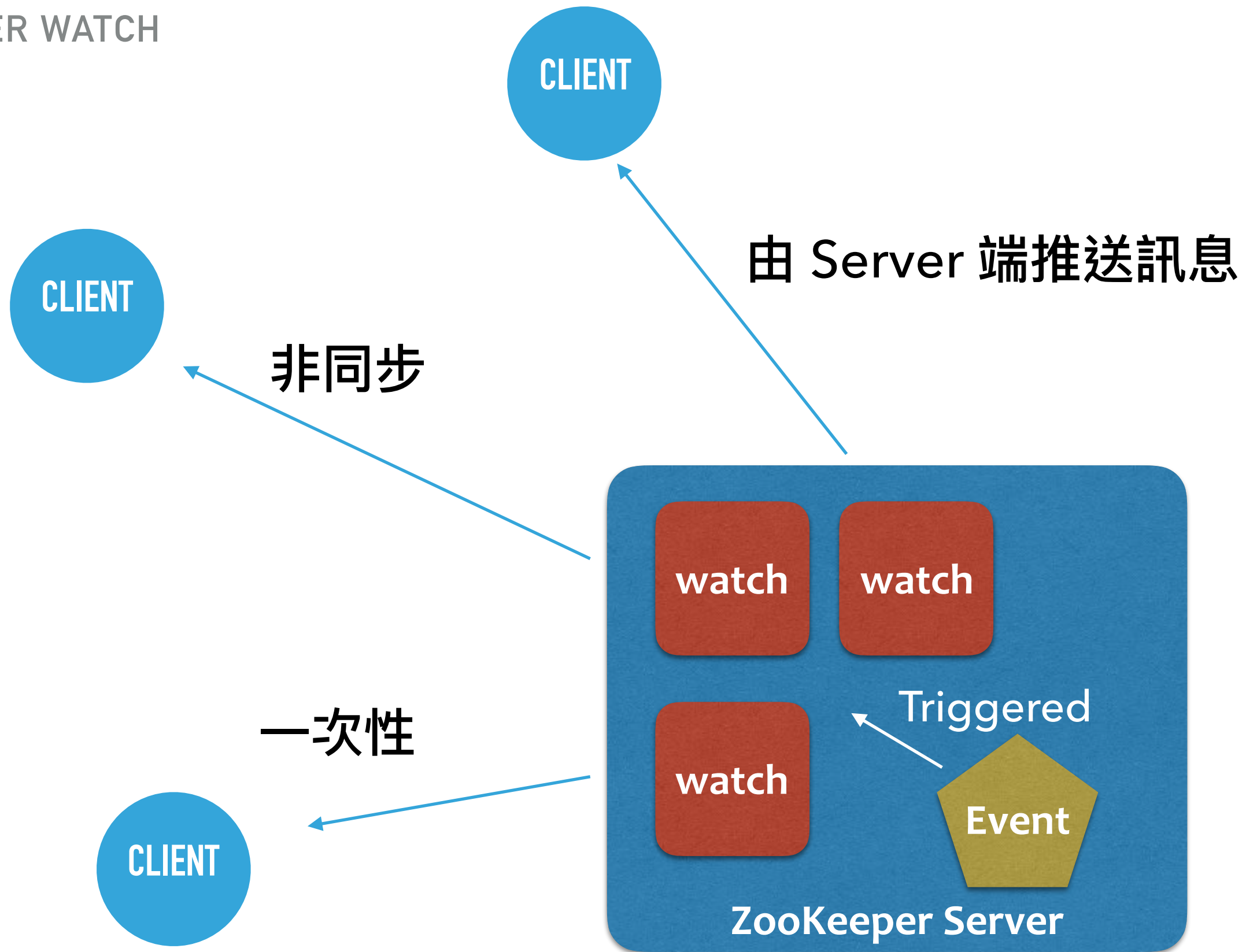
ZOOKEEPER WATCH



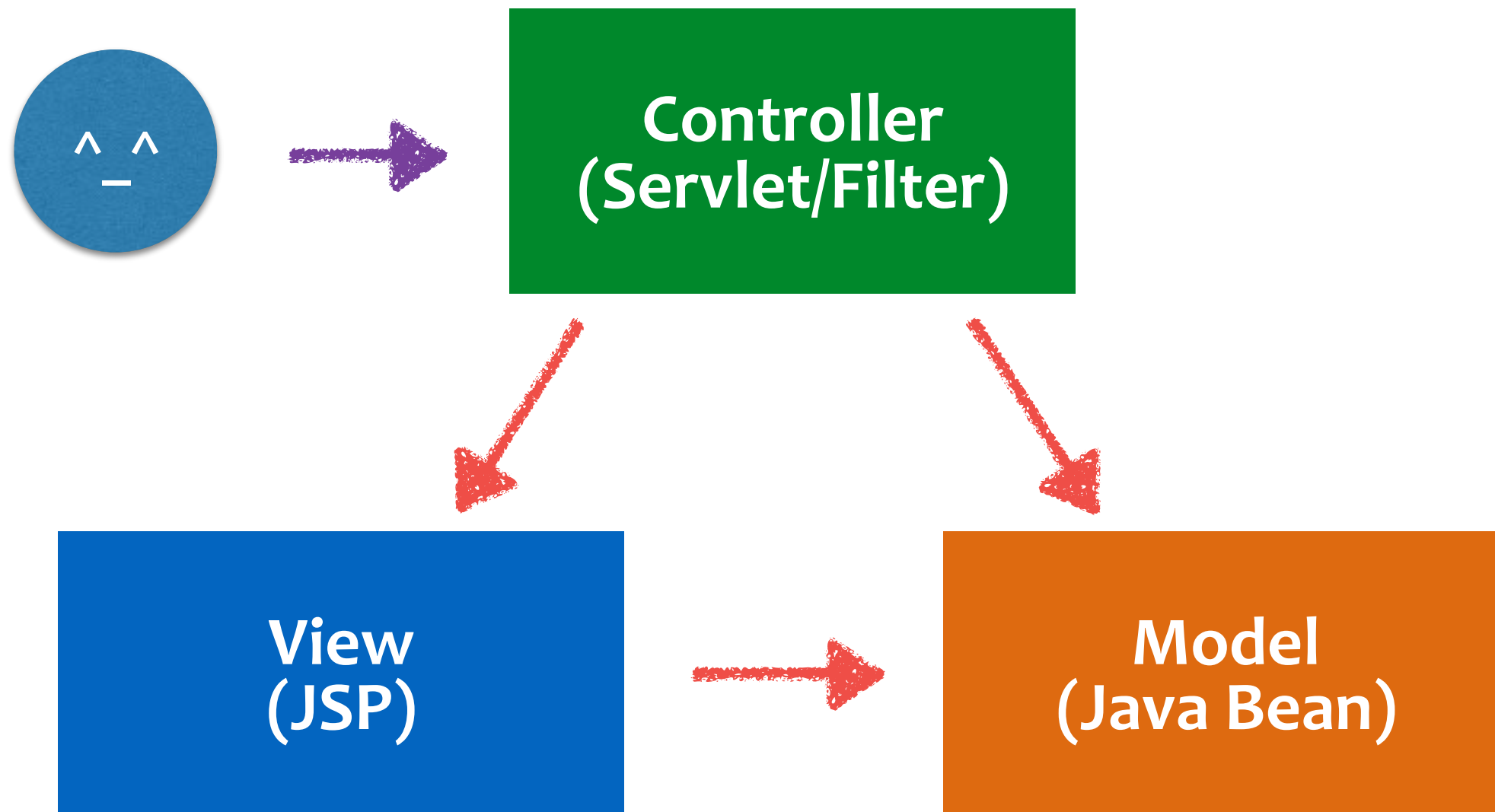
ZOOKEEPER WATCH



ZOOKEEPER WATCH



A MCV EXAMPLE (MODEL 2)

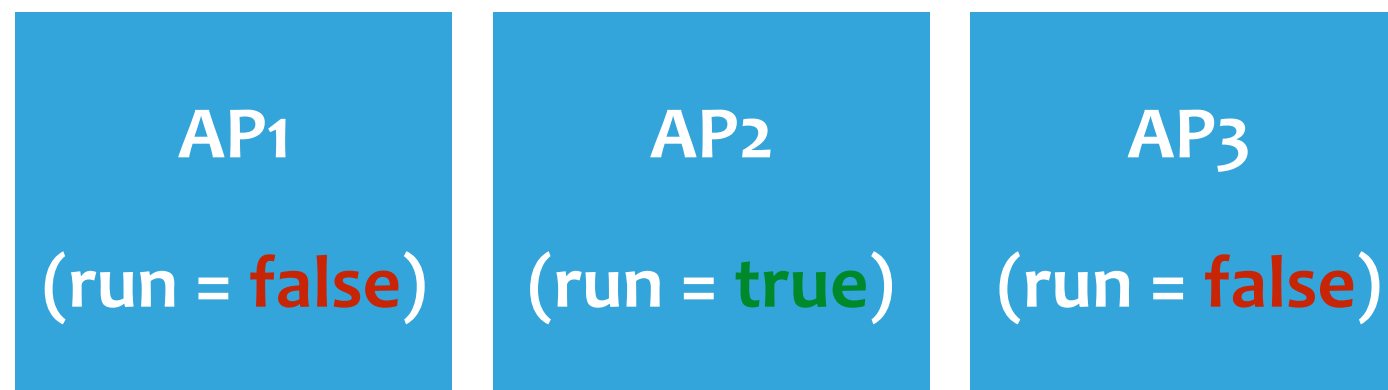




故事是這樣子的...

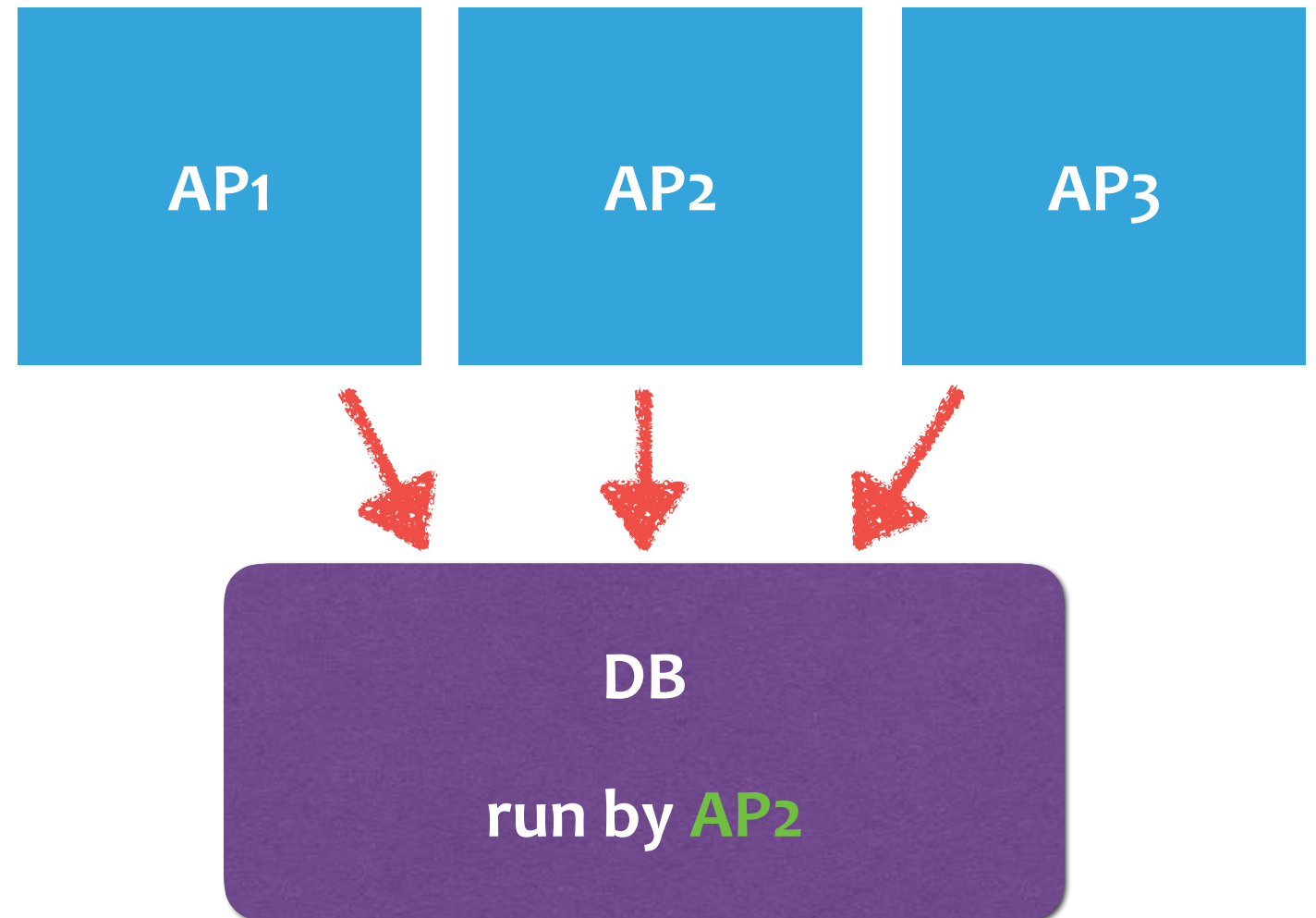
BEFORE ZOOKEEPER

- ▶ 由ap自己控制
 - ▶ hardcode
 - ▶ properties
 - ▶ passes argument ...
- ▶ cons?



BEFORE ZOOKEEPER

- ▶ 將控制點抽離ap
 - ▶ db table control
 - ▶ db lock
- ▶ cons?



我們需要的是...

有一個方法可以實作自己的邏輯：
決定誰要執行

BEFORE ZOOKEEPER



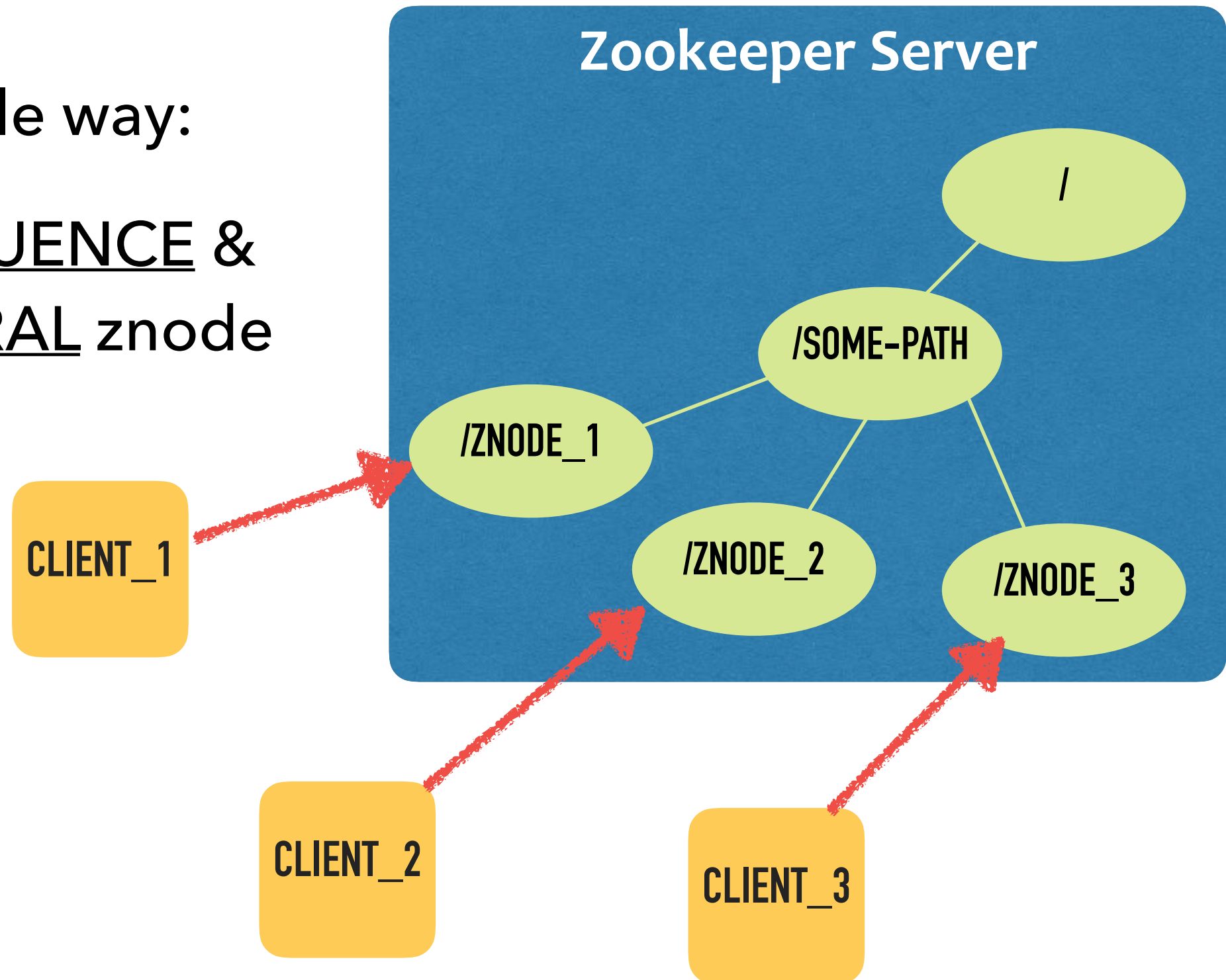
ZooKeeper Recipes

<https://zookeeper.apache.org/doc/trunk/recipes.html>

LEADER ELECTION

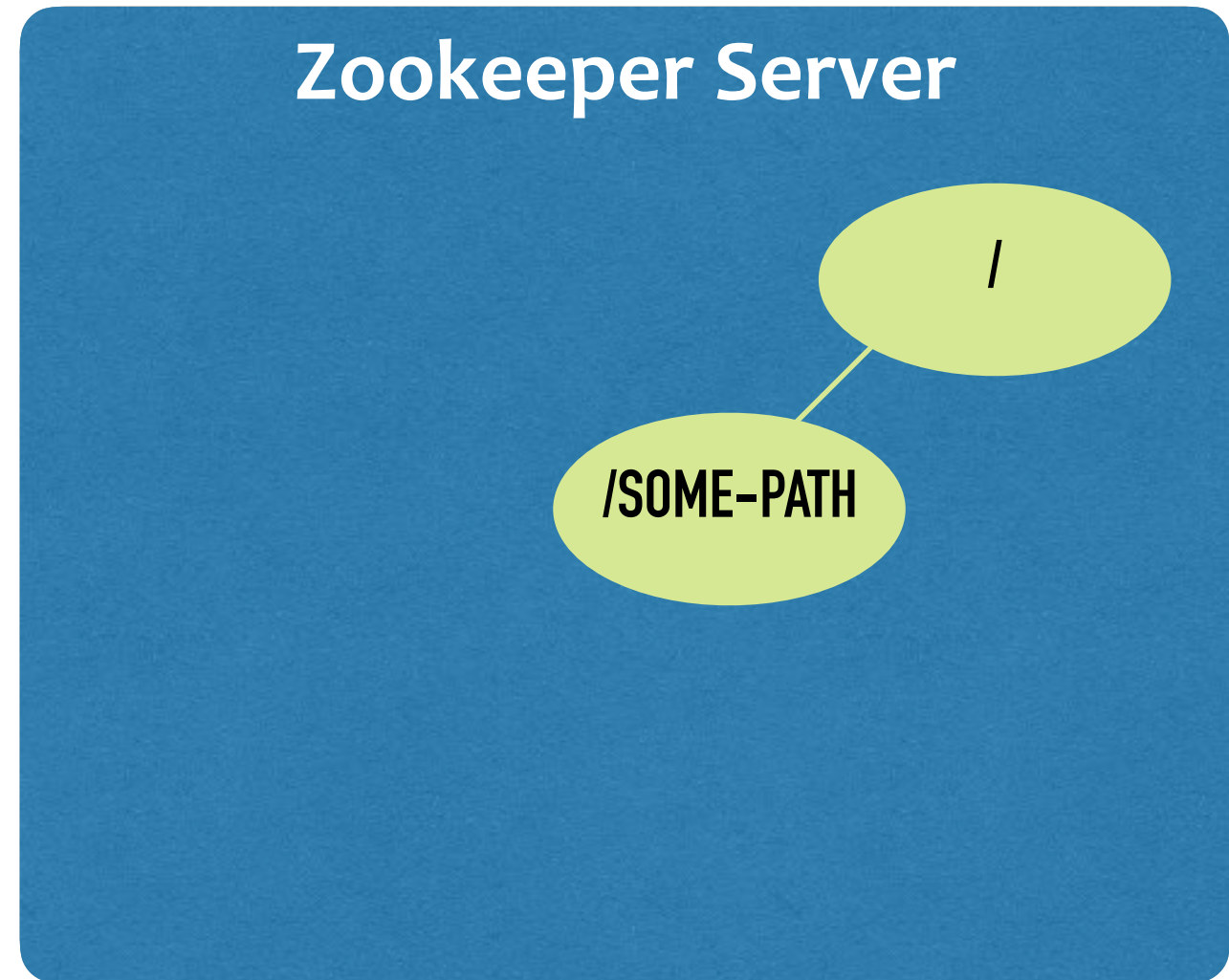
A simple way:

Use SEQUENCE &
EPHEMERAL znode



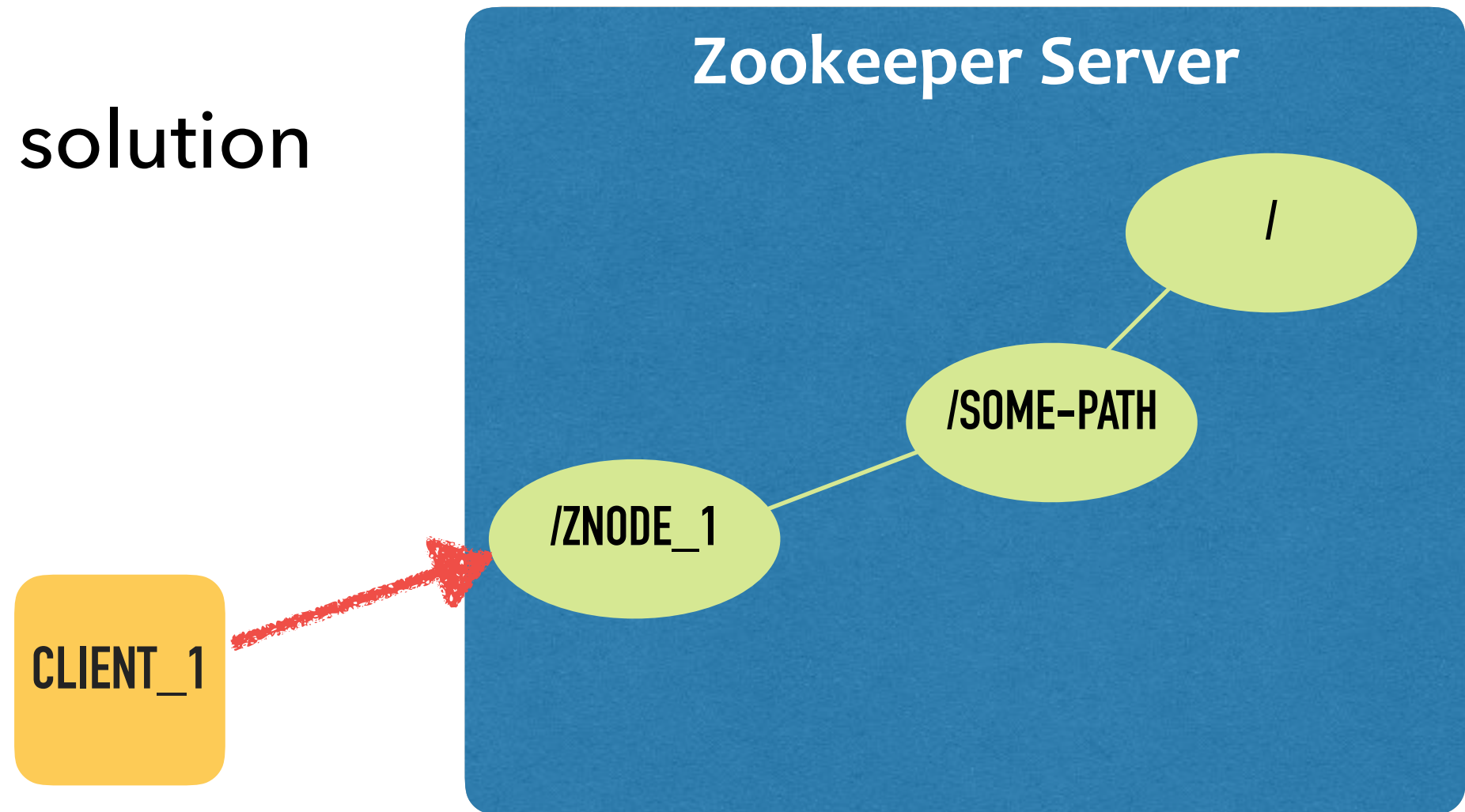
https://zookeeper.apache.org/doc/trunk/recipes.html#sc_leaderElection

Advance solution



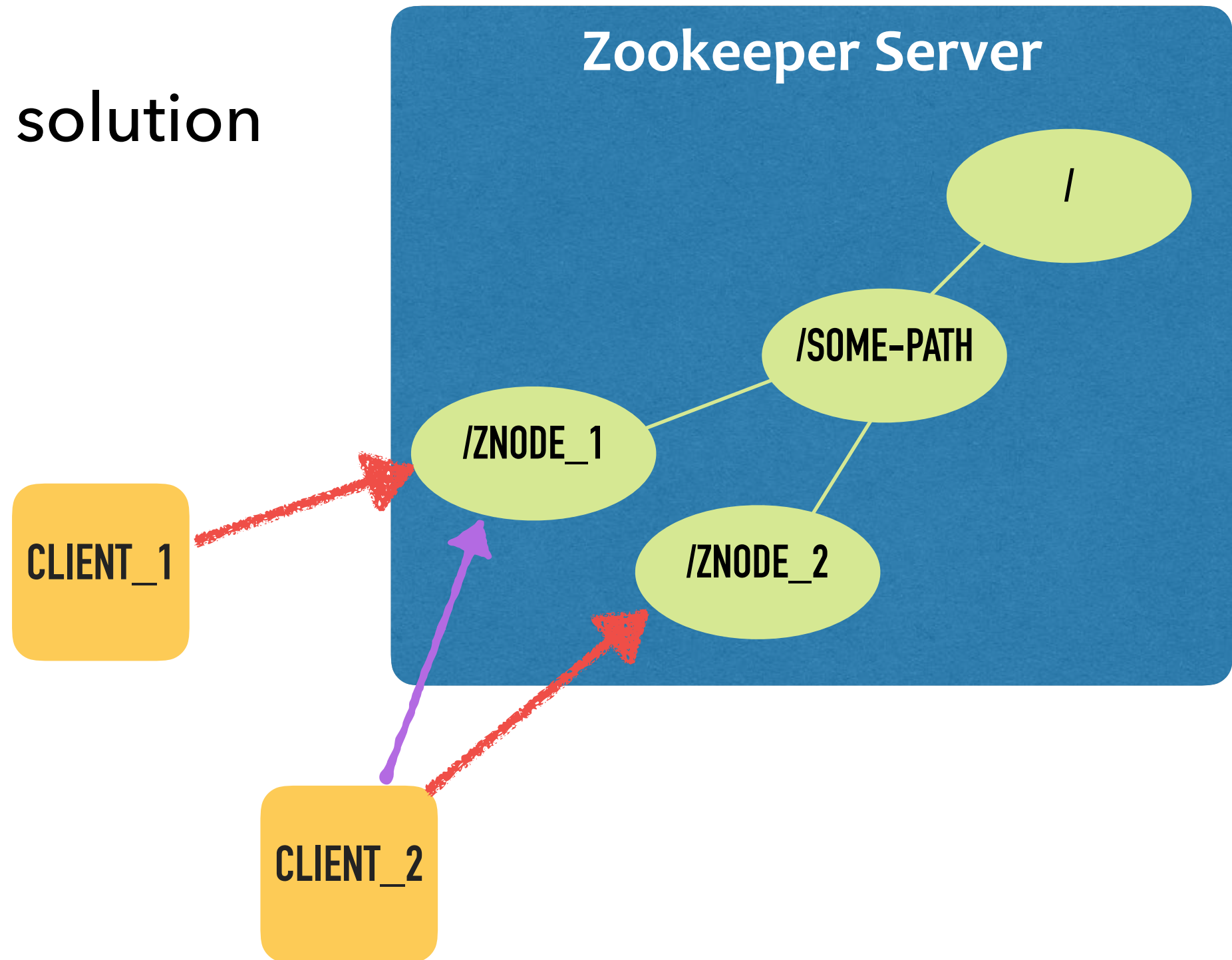
https://zookeeper.apache.org/doc/trunk/recipes.html#sc_leaderElection

Advance solution



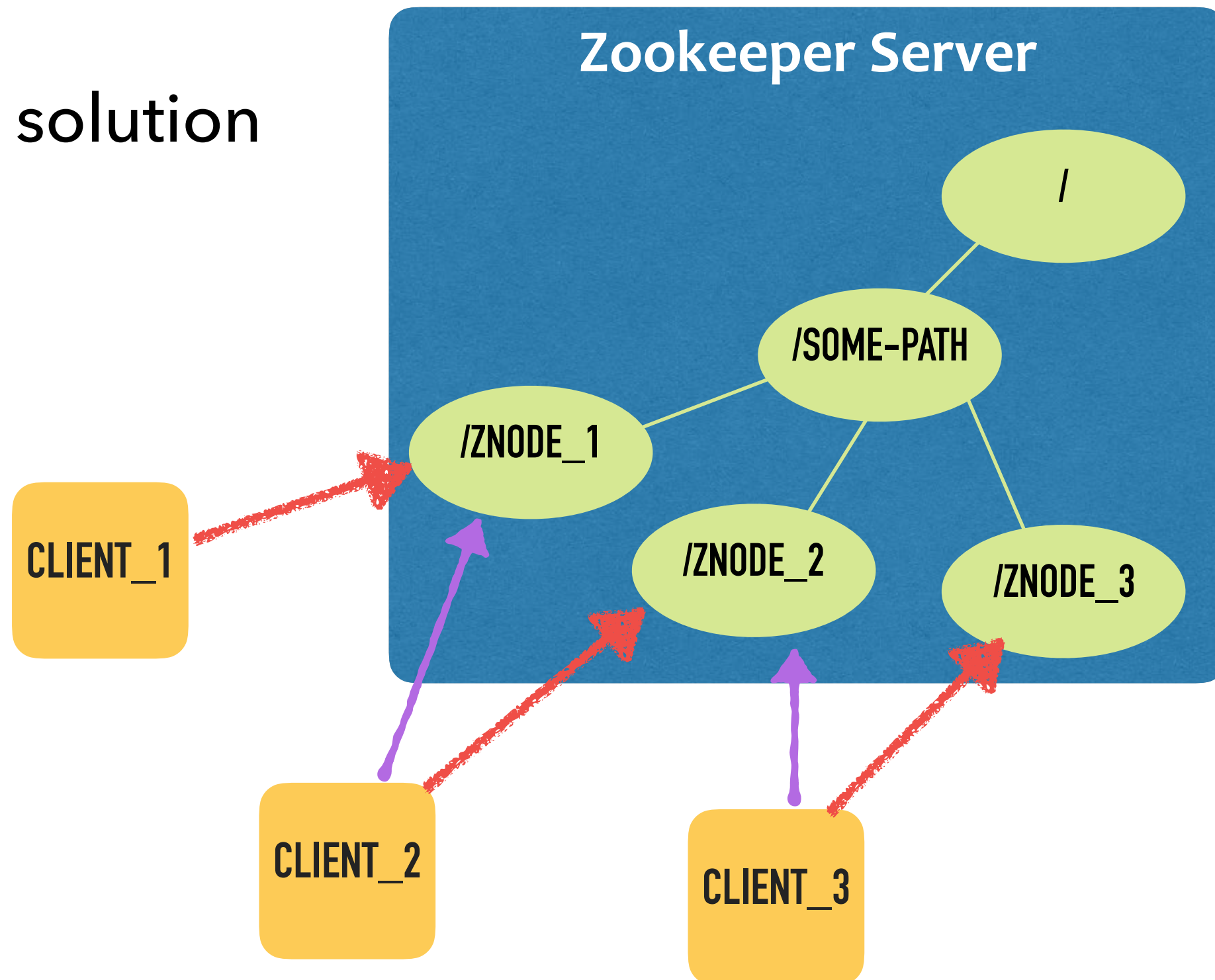
https://zookeeper.apache.org/doc/trunk/recipes.html#sc_leaderElection

Advance solution



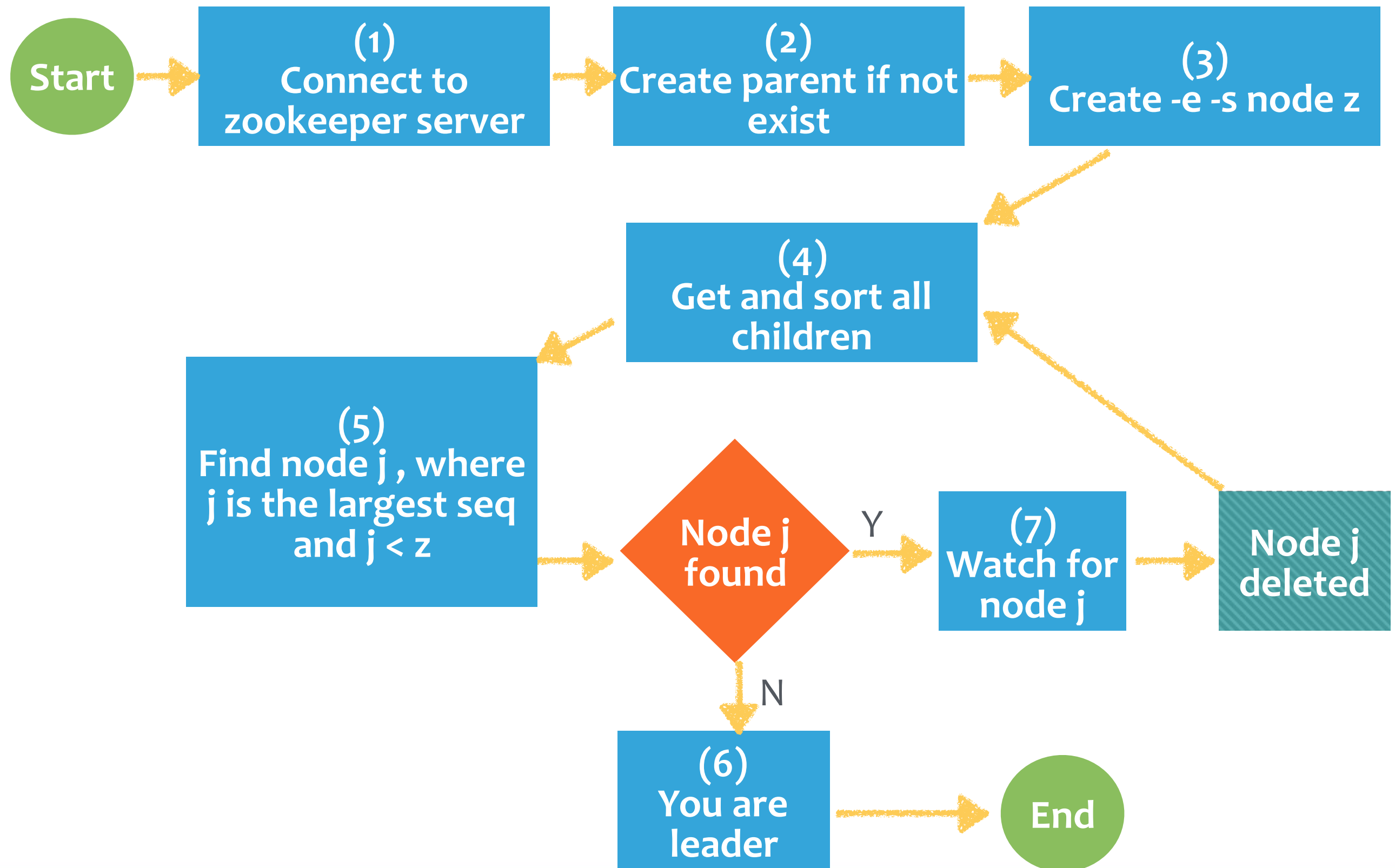
https://zookeeper.apache.org/doc/trunk/recipes.html#sc_leaderElection

Advance solution

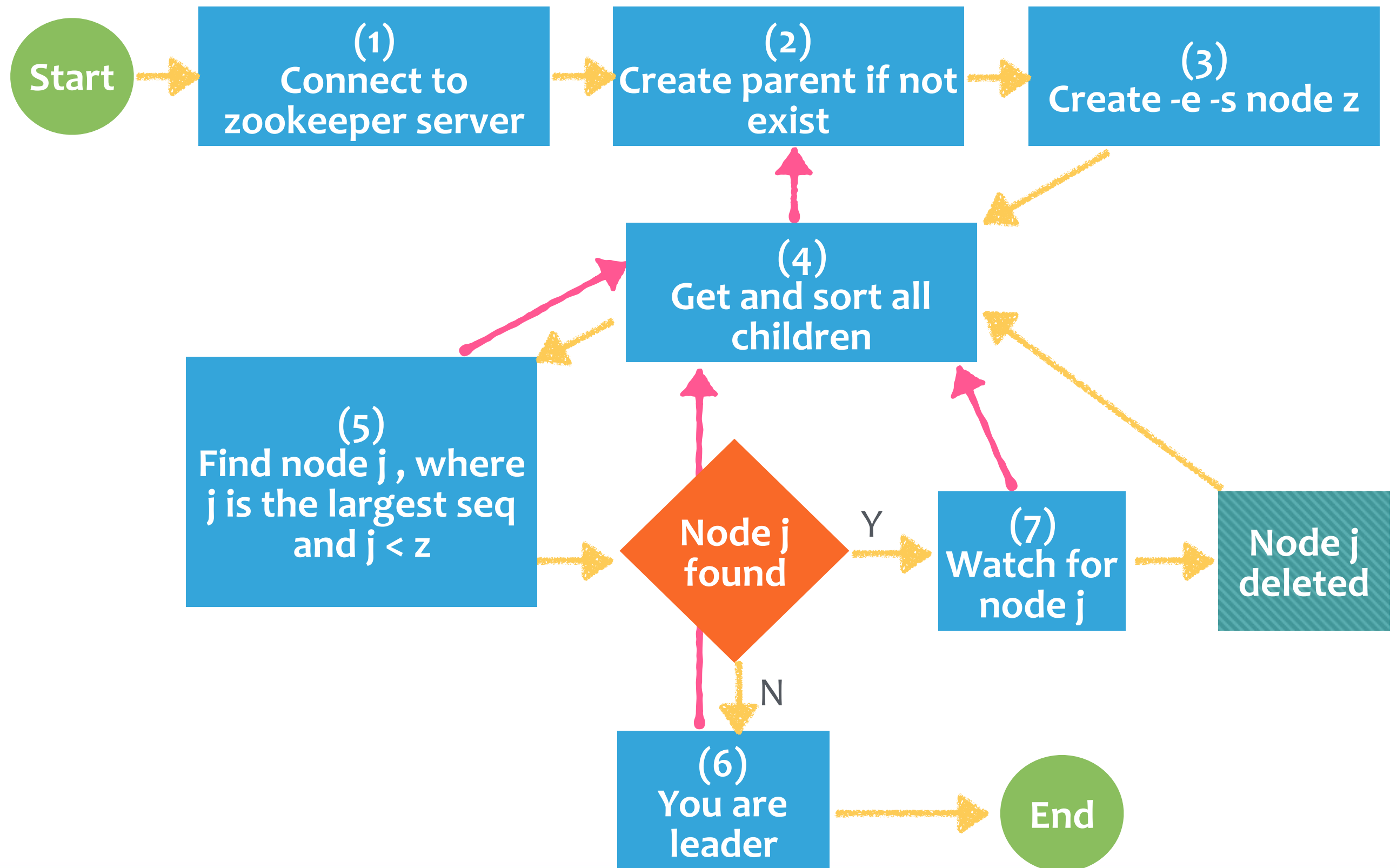


https://zookeeper.apache.org/doc/trunk/recipes.html#sc_leaderElection

IMPLEMENTING LEADER ELECTION



IMPLEMENTING LEADER ELECTION





"Guava is to Java what Curator is to ZooKeeper"
Patrick Hunt, ZooKeeper committer

<http://curator.apache.org/>

PROGRAMMING TO ZOOKEEPER

(1)
Connect to
zookeeper server

```
zk = new ZooKeeper(connectString, sessionTimeout, watcher);
```

```
client =  
    CuratorFrameworkFactory.newClient(connectString, retryPolicy);  
client.start();
```

(2) Create parent if not exist

```
try {  
    zk.create(rootPath, new byte[0], acl, CreateMode.PERSISTENT);  
} catch (ConnectionLossException e) {  
    // try to create again  
} catch (NodeExistsException e) {  
    // no-op  
} catch (KeeperException | InterruptedException e) {  
    // handle exception  
}
```

```
leaderLatch = new LeaderLatch(client, rootPath);  
leaderLatch.start();
```

(3) Create -e -s node z

```
zk.create(rootPath + "/jcconf2016-", new byte[0],
    acl, CreateMode.EPHEMERAL_SEQUENTIAL, callback, null);

private StringCallback callback = new StringCallback() {
    @Override
    public void processResult(int rc, String path, Object ctx, String name) {
        switch (Code.get(rc)) {
            case SESSIONEXPIRED:
            case CONNECTIONLOSS:
                // try to create again
                break;
            case OK:
                // check leader
                break;
            case NODEEXISTS:
                // check leader
                break;
            default:
                // handle exception
        }
    }
};
```

(4)
Get and sort all
children

(5)
Find node j, where
j is the largest seq
and $j < z$

```
zk.getChildren(rootPath, false, attemptToTakeLeadership, null);
```

```
private ChildrenCallback attemptToTakeLeadership = new ChildrenCallback() {  
    @Override  
    public void processResult(int rc, String path, Object ctx, List<String> children) {  
        switch (Code.get(rc)) {  
            case OK:  
                Collections.sort(children); // 排序  
                // 如果 1st index of children 等於 current client >> I'm leader  
                // 如果不是, 取出前一的 index 的 child, 放置 watch 監控  
                zk.getChildren(rootPath + "/" + children.get(index - 1), znodeDeleted);  
                break;  
            default:  
                // handle exception  
        }  
    }  
};
```

(7) Watch for node j

```
private Watcher znodeDeleted = new Watcher() {  
    @Override  
    public void process(WatchedEvent e) {  
        switch (e.getType()) {  
            case NodeDeleted:  
                // check leader  
                break;  
            default:  
                break;  
        }  
    }  
};
```

PROGRAMMING TO ZOOKEEPER

```
leaderLatch.hasLeadership();
```

```
leaderLatch.hasLeadership();
```

剛寫那麼多都是寫 _____ der...



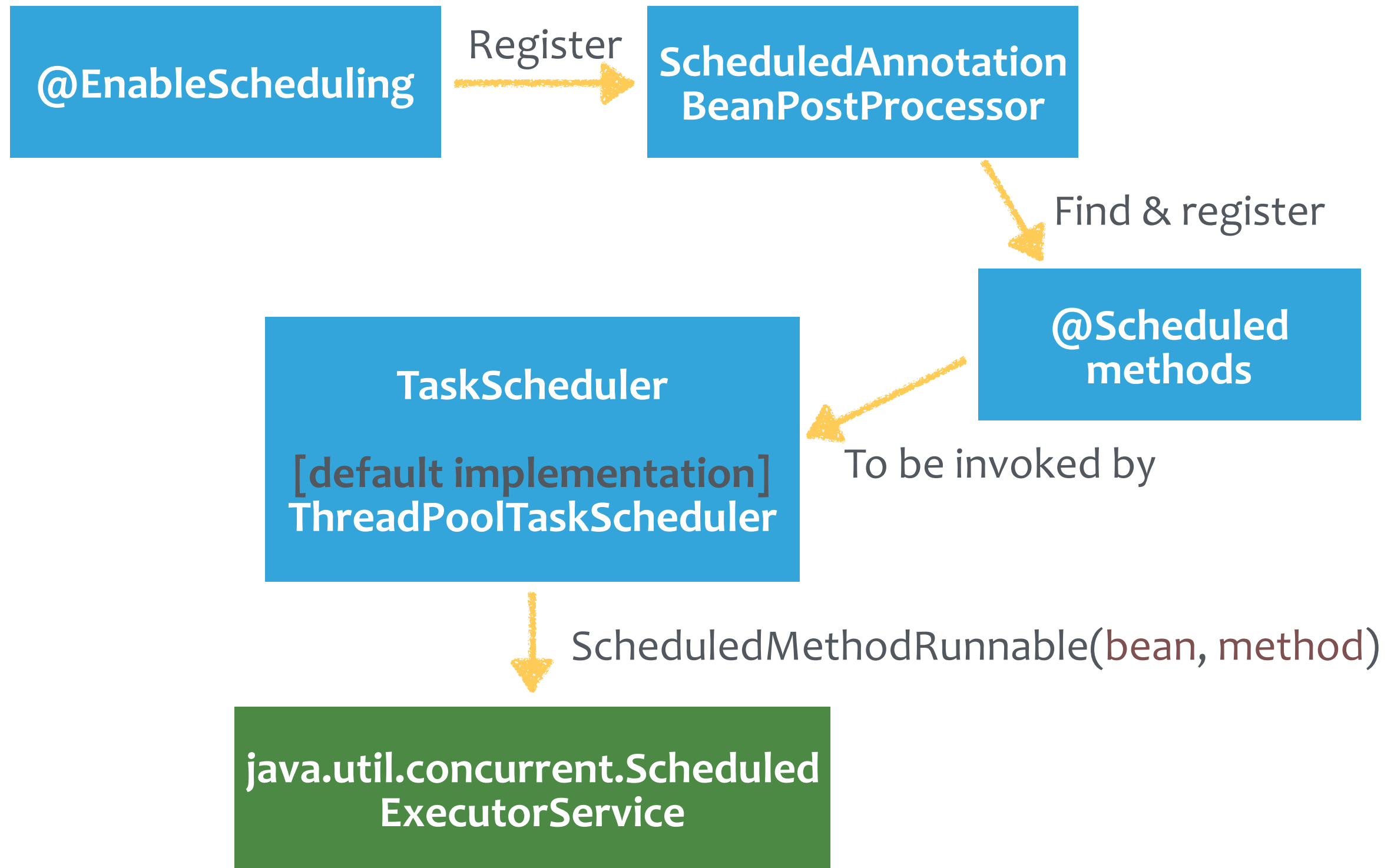
怎麼串接到排程中..?

ADAPTING INTO SCHEDULE FRAMEWORK

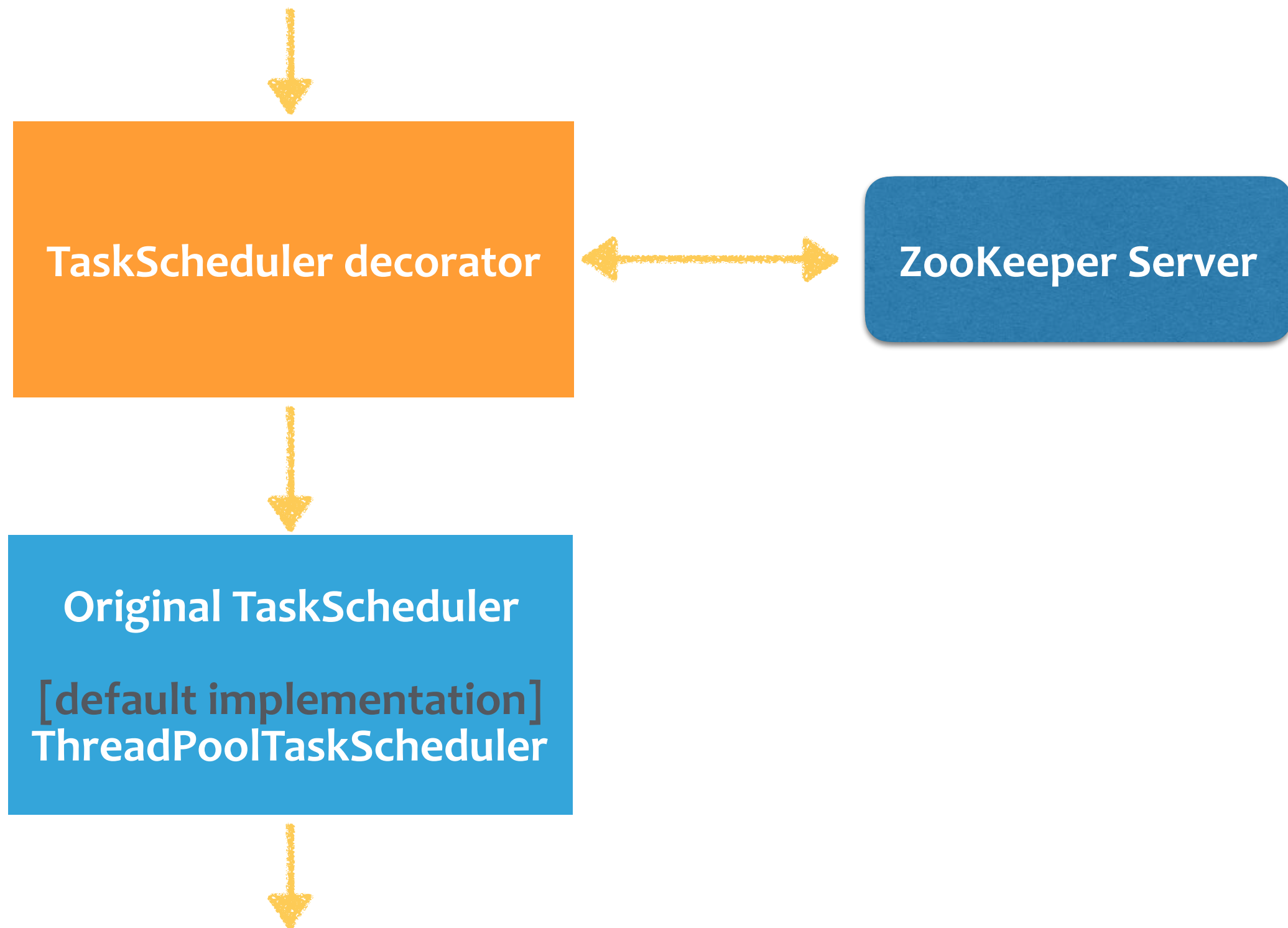
`@EnableScheduling`

`@Scheduled
methods`

ADAPTING INTO SCHEDULE FRAMEWORK



ADAPTING INTO SCHEDULE FRAMEWORK



PROGRAMMING TO SCHEDULE FRAMEWORK

```
@RequiredArgsConstructor
public class ZookeeperTaskScheduler implements TaskScheduler {

    private final TaskScheduler delegate;

    public ScheduledFuture<?> schedule(Runnable task, Trigger trigger) {
        return delegate.schedule(runsIfOwnLeadership(task), trigger);
    }
    public ScheduledFuture<?> schedule(Runnable task, Date startTime) {
        return delegate.schedule(runsIfOwnLeadership(task), startTime);
    }
    // 用同樣的方式實作所有 method...

    private Runnable runsIfOwnLeadership(Runnable runnable) {
        return () -> {
            if (ownLeadership) {
                runnable.run();
            }
        };
    }
}
```

PROGRAMMING TO SCHEDULE FRAMEWORK

```
@Configuration
@EnableAsync
@EnableScheduling
@ComponentScan(basePackages = {"tw.com.softleader.*"})
public static class ScheduledConfig {

    @Bean
    @Primary
    public TaskScheduler taskScheduler() {
        ThreadPoolTaskScheduler taskScheduler = new ThreadPoolTaskScheduler();
        taskScheduler.afterPropertiesSet();
        return new ZookeeperTaskScheduler(taskScheduler);
    }
}
```

PROGRAMMING TO SCHEDULE FRAMEWORK

```
@Service
public class SomeTask {

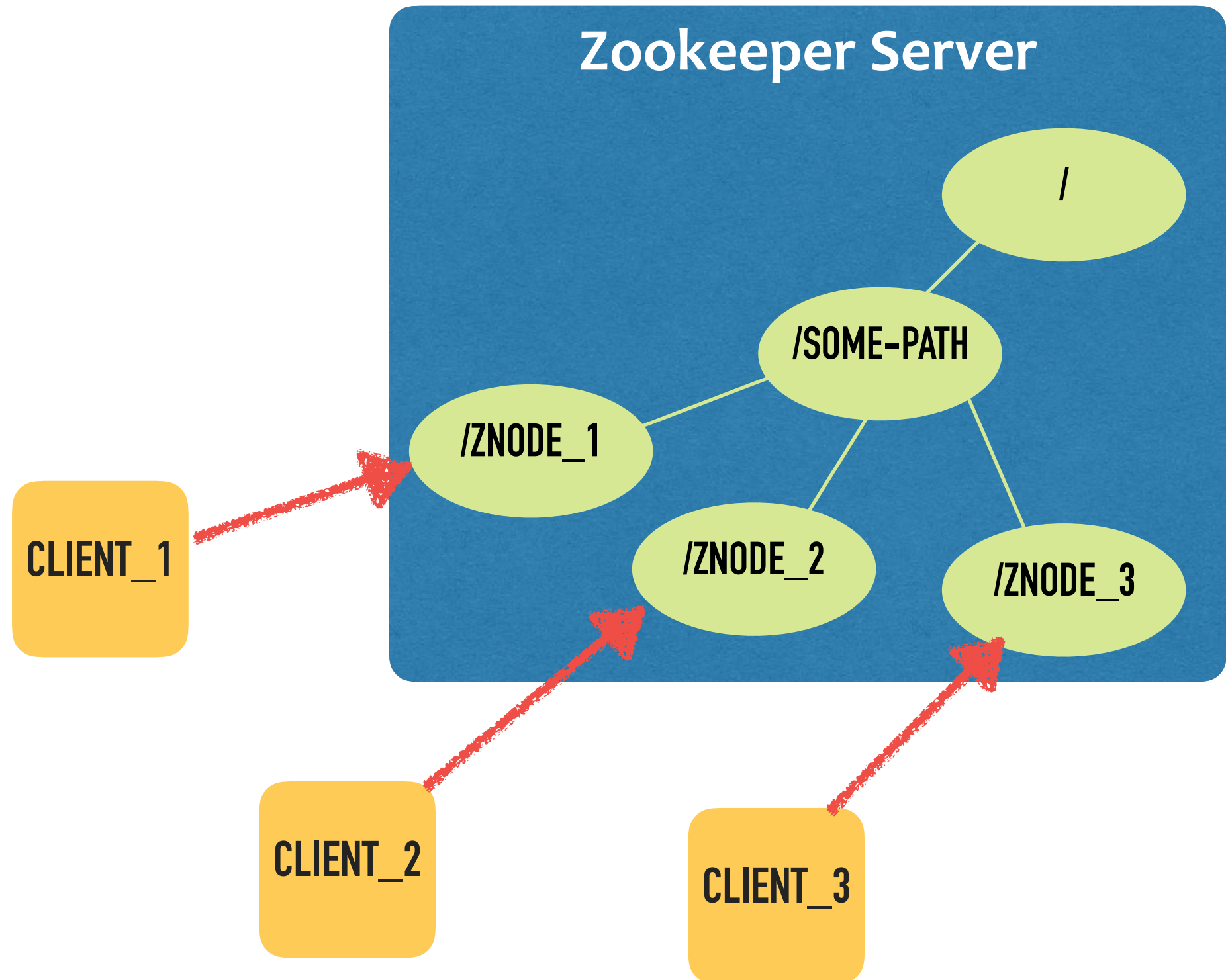
    @Scheduled(fixedDelay = 500)
    public void doMyTask() {
        // 排程內容..

        // 所以我們就並不用修改任何程式碼，如進 method 時先去判斷是否為 leader
    }
}
```

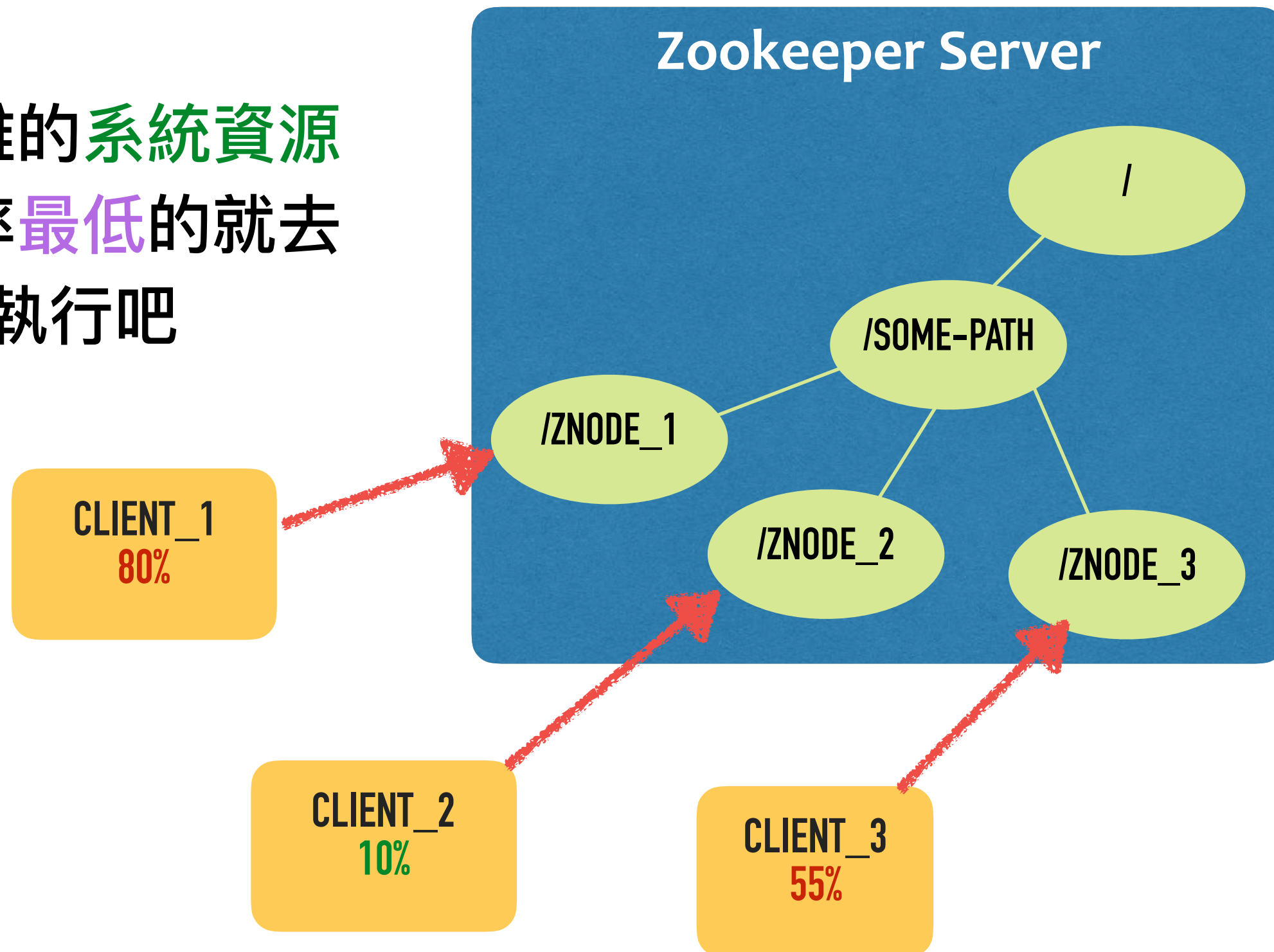
來看看實際的程式碼吧...

還可以做些什麼？

LEADER ELECTION



看看誰的系統資源
使用率最低的就去
執行吧



Please leave your comments ↓

<https://github.com/methodho/jcconf2016-zookeeper>

Please leave your comments ↓

<https://github.com/methodho/jcconf2016-zookeeper>

And we're hiring!

<http://www.softleader.com.tw/web/career.html>