

자바 기반 CoAP 설치 가이드

요약 : JAVA 기반의 Californium Framework를 사용하여 Coap 프로토콜을 사용한 통신을 할 수 있는 환경을 구축한다.

실행환경 : Windows 7 Enterprise 64bit, JDK 1.7.0_67, Eclipse Luna X64
(문서 작성자의 실행환경이며, 실제로는 Java를 실행할 수 있는 대부분의 환경에서 구동 가능하다)

경북대학교 컴퓨터학부
2009038063 이상준
2009038004 김동민
2009038042 장준희
2009038053 최예찬
2011105108 홍진경

- 1) <https://github.com/mkovatsc/Californium> 에서 오른쪽 Download ZIP 버튼을 눌러 원하는 폴더에 CoAP Source를 다운받은 후 압축을 해제한다.

GitHub This repository Search Explore Features Enterprise Blog Sign up Sign in

mkovatsc / Californium ★ Star 57 🍴 Fork 41

Californium (Cf) CoAP framework (Java implementation) <http://people.inf.ethz.ch/mkovatsc/californium.php>

867 commits 12 branches 4 releases 8 contributors

branch: coap-18 Californium / +

Repository freeze.

mkovatsc authored on Jul 4 latest commit 9a4e43a84f

californium-proxy	Final release for central.	3 months ago
californium	Final release for central.	3 months ago
cf-api-demo	Extended CoapClient API.	7 months ago
cf-benchmark	Final release for central.	3 months ago
cf-browser	Final release for central.	3 months ago
cf-client	Final release for central.	3 months ago
cf-coapbench	Final release for central.	3 months ago
cf-helloworld-client	Final release for central.	3 months ago
cf-helloworld-server	Final release for central.	3 months ago
cf-osgi	Final release for central.	3 months ago
cf-plugtest-checker	Final release for central.	3 months ago
cf-plugtest-client	Final release for central.	3 months ago
cf-plugtest-server	Final release for central.	3 months ago
cf-proxy	Final release for central.	3 months ago

Code

Issues 2

Pull Requests 0

Pulse

Graphs

HTTPS clone URL

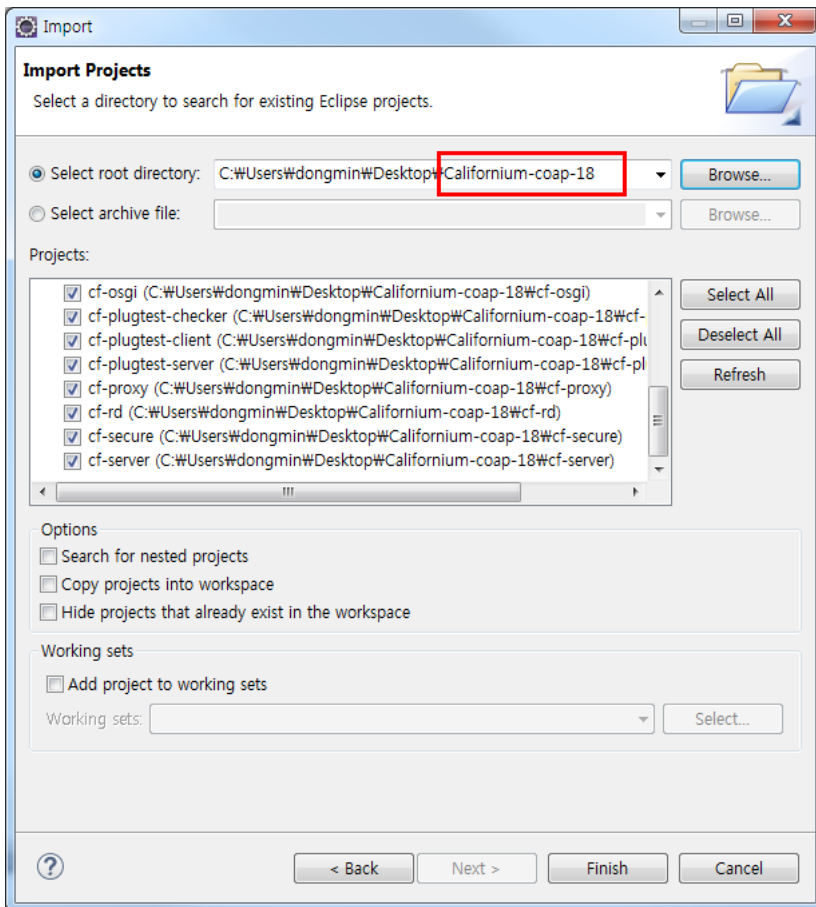
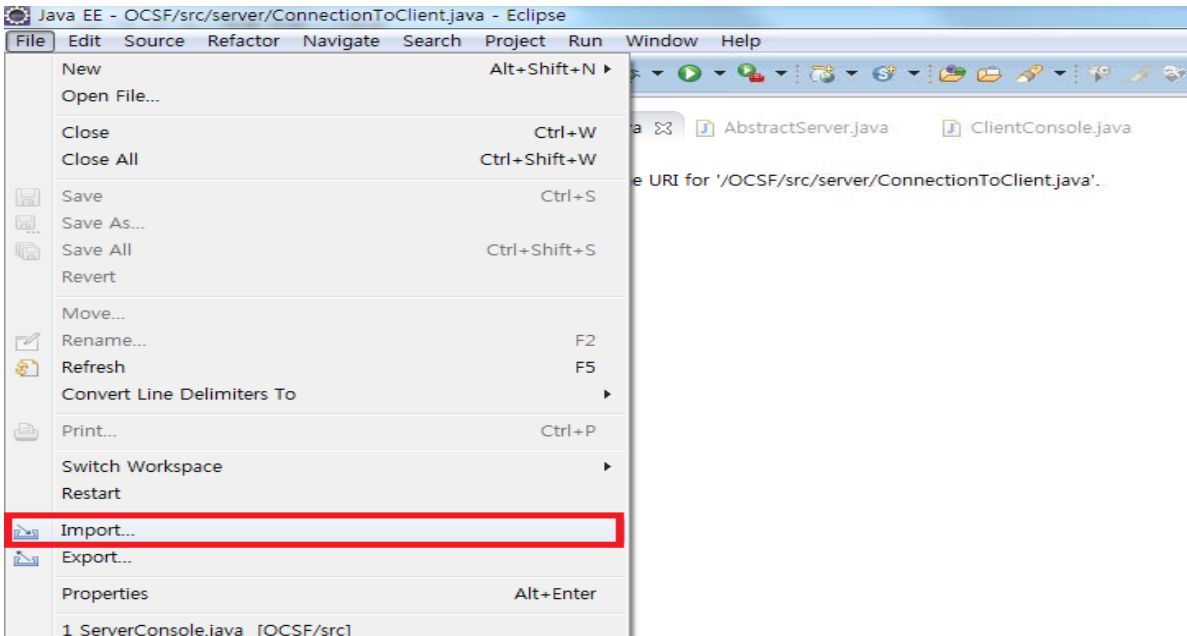
<https://github.com/mkovat>

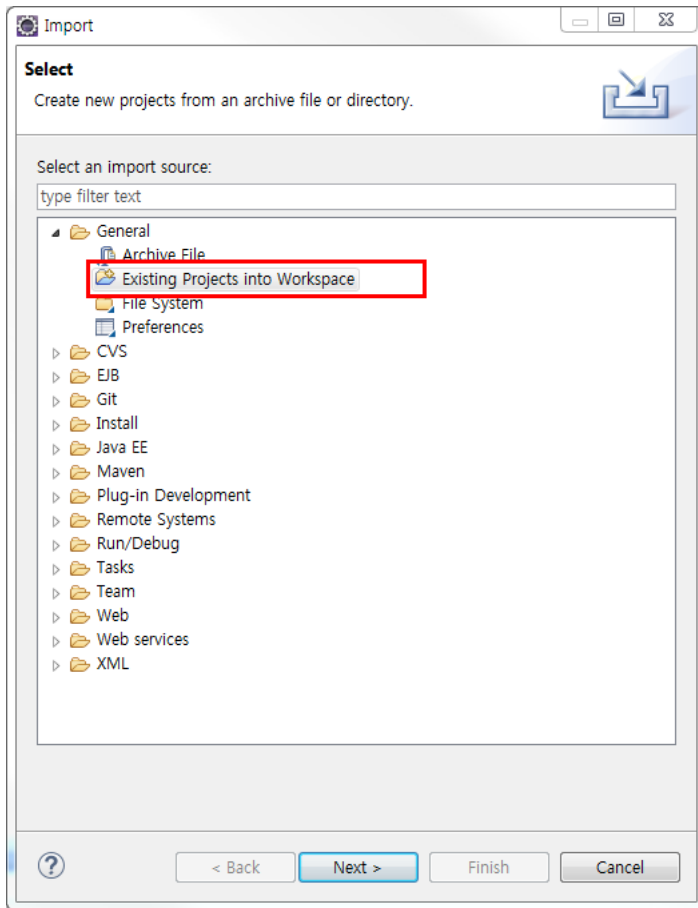
You can clone with HTTPS or Subversion.

Clone in Desktop

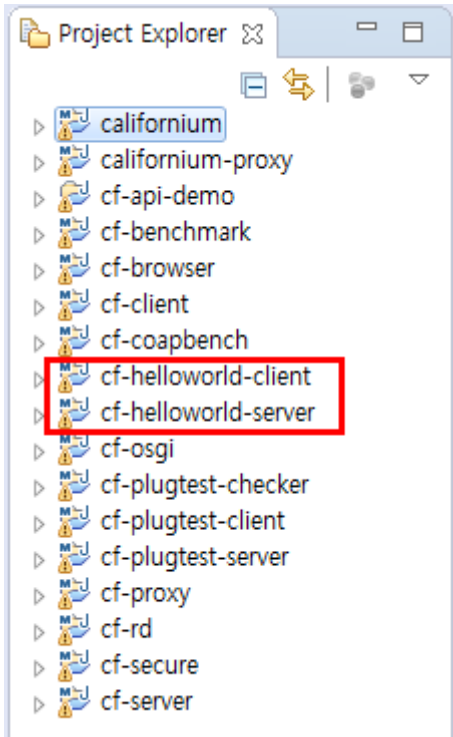
Download ZIP

2) Eclipse 실행 후 압축 해제한 CoAP Source를 위와 같은 순서로 Import한다.

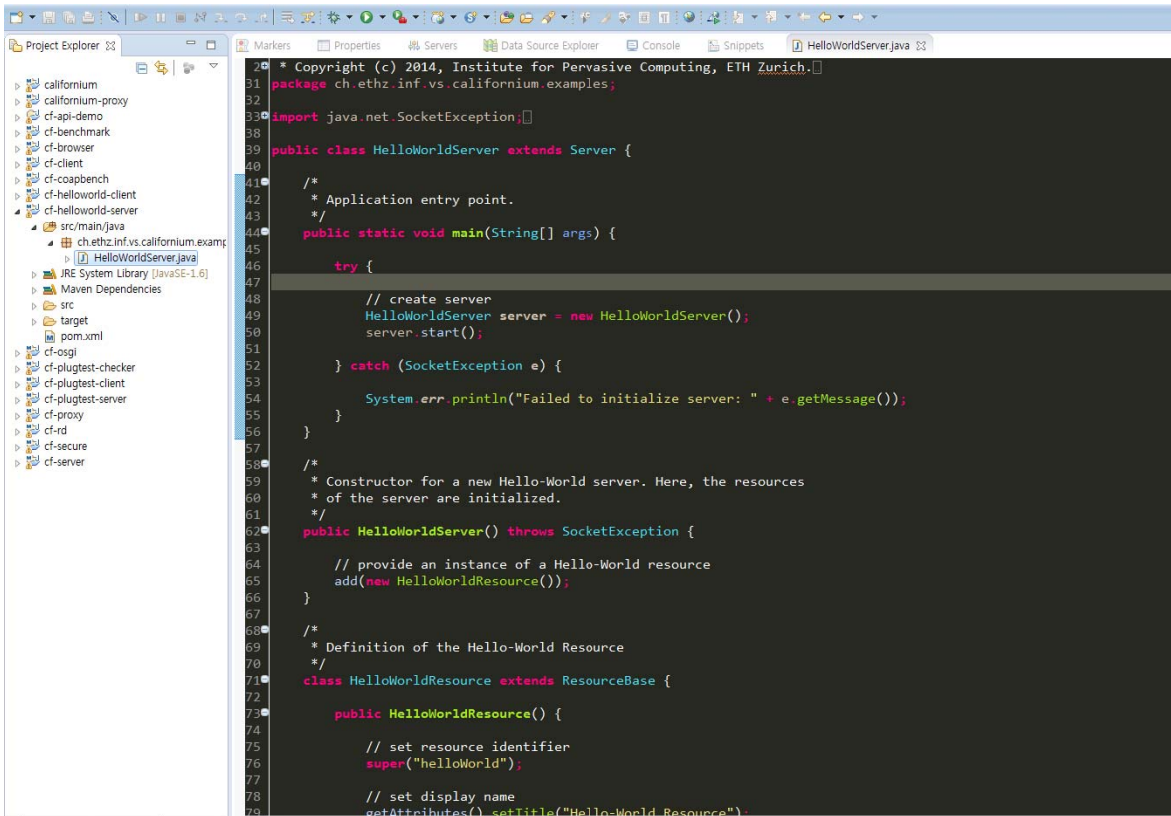




3) Import하게 될시 Project Explorer에서 위와 같은 모습을 확인 할 수가 있다. 이 소스들 중에서 Echo Test를 하는데 필요한 Source는 cf-helloworld-client, cf-helloworld-server 두가지이다.



4) 먼저 서버를 실행해 보자. 위와 같이 Source를 Open한 후 실행을 시키자

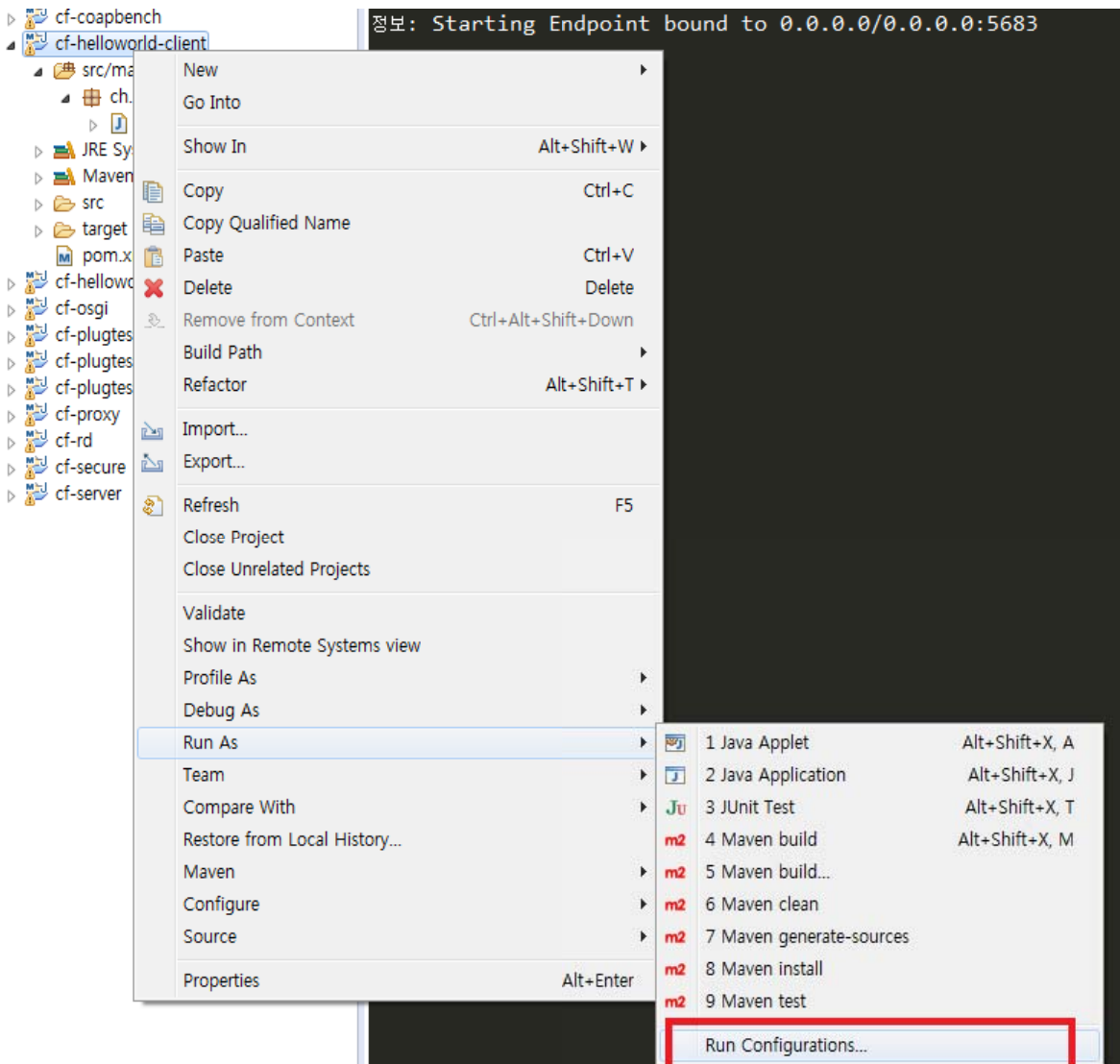


5) 서버 실행 시 위의 그림과 같은 실행 결과를 확인할 수가 있으며 현재 상황은 Server가 Client의 접속을 기다리고 있는 상황이다. (기본 포트 UDP/5683)

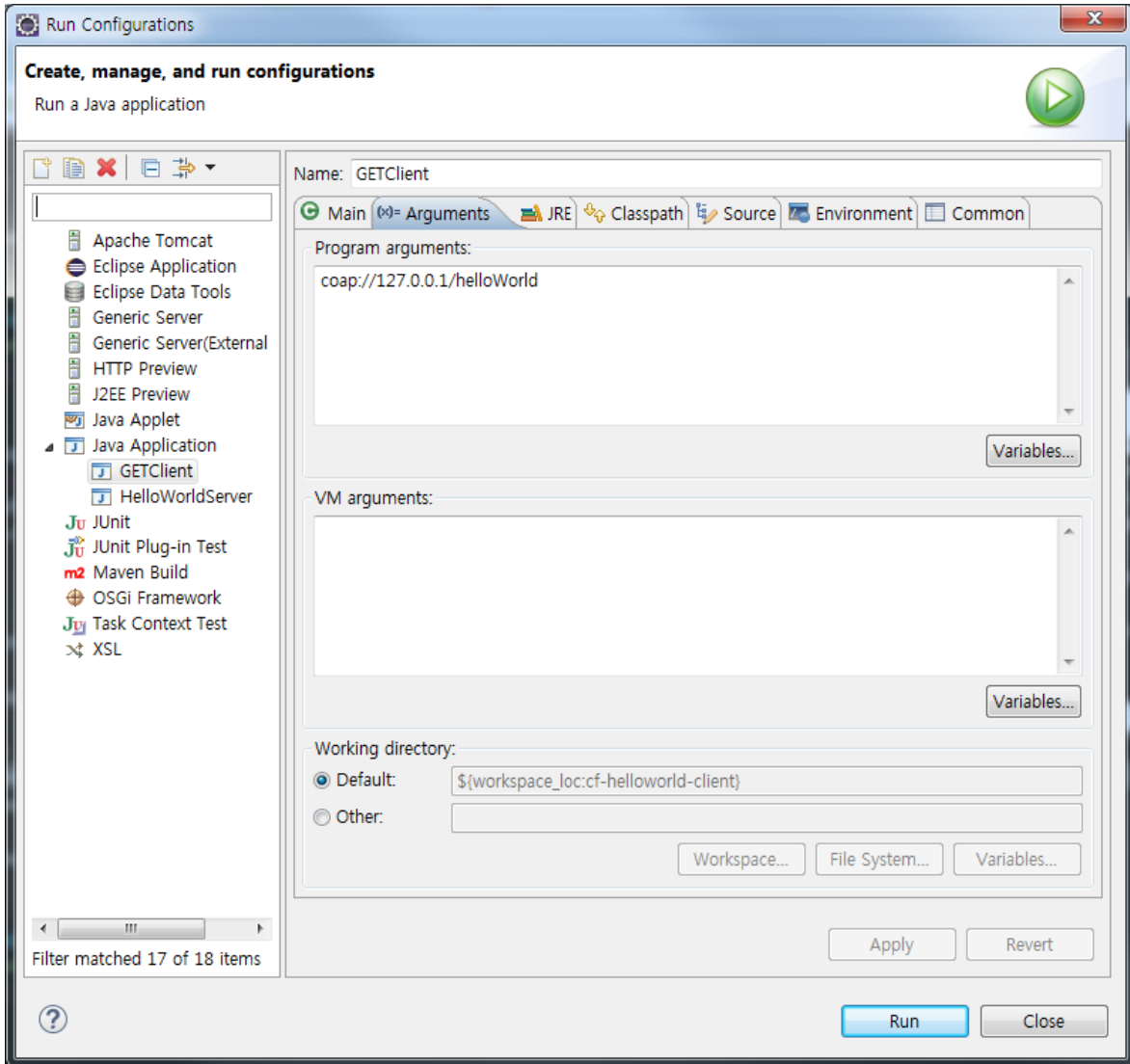
```

HelloWorldServer [Java Application] C:\Program Files\Java\jre1.8.0_20\bin\javaw.exe (2014. 10. 15. 오후 8:07:32)
10월 15, 2014 8:07:32 오후 ch.ethz.inf.vs.californium.network.config.NetworkConfig createStandardWithFile
정보: Create standard properties with file Californium.properties
10월 15, 2014 8:07:32 오후 ch.ethz.inf.vs.californium.server.Server start
정보: Starting server
10월 15, 2014 8:07:32 오후 ch.ethz.inf.vs.californium.server.Server start
정보: No endpoints have been defined for server, setting up default endpoint at port 5683
10월 15, 2014 8:07:32 오후 ch.ethz.inf.vs.californium.network.CoAPEndpoint start
정보: Starting Endpoint bound to 0.0.0.0/0.0.0.0:5683
  
```

6) 이제 Client를 실행하는 법을 알아보자. 실행법이 간단한 Server와 달리 Client의 실행법은 실행시 매개변수를 전달하여야 하기 때문에 차이가 있다. 먼저 위와 같이 Project Explorer창의 cf-helloworld-client소스 위에서 오른쪽 클릭후 Run As->Run Configurations 를 선택한다.



6) Run configurations 창이 열리게 되면 오른쪽 창에서 우리가 실행하고자 하는 GETClient를 선택 후 Arguments 탭으로 이동한다. 이 탭에서 실행 시 전달할 매개변수를 지정할 수가 있다. Server와 Client의 Echo Test를 위해선 이곳에 coap://127.0.0.1/helloWorld를 입력한 후 Run으로 Client를 실행한다.



7) 위와 같은 방법으로 실행 시 Client Console에서 'Hello World!' 라는 문장을 Server로부터 receive한 것을 확인할 수가 있다.

