



CCBUS

Connect Tool

Table of Contents

1 Introduction.....	1
2 Requirements.....	1
3 Installation.....	1
4 Tutorial.....	3
4.1 Compilation.....	3
4.2 Java project structure.....	3
4.3 React project structure.....	3
4.4 Supported types.....	4
4.5 Demo projects.....	4

1 Introduction

Presents tool for compilation from Java code to Babel javascript (extension of EcmaScript 6.0), Java API and React API for cross-platform development.

2 Requirements

- Java 8
- Maven tool
- Npm tool.
- React > 16.*

3 Installation

- Checkout Java Api in you java project folder. Source is shared between all projects

```
terminal$ svn checkout svn://192.168.0.2/projects/ccbus/ccbusConnectLib/trunk ccbusConnectLib
```

```
terminal$ cd ccbusConnectLib
```

```
terminal$ mvn install
```

Add module as dependency in you Maven project



```
<dependency>
  <groupId>ccbus.connect</groupId>
  <artifactId>ccbus-connect</artifactId>
  <version>1.0.0</version>
</dependency>
```

It could be imported as module in your maven project, but still it have to be installed in mvn repository with "mvn install"

- Checkout React Api as relative path to your react projects:

```
terminal$ svn checkout svn://192.168.0.2/projects/ccbus/ccbusReact/ccbus/trunk ccbus
```

Add following modules as dependencies in you react project

```
"dependencies":
{
  "axios": "^0.19.0",
  "ccbus": "../ccbus"
},
```

```
terminal$ npm install
```

- Checkout Ccbus compilation tool

```
terminal$ svn export svn://192.168.0.2/projects/ccbus/ccbusTool/trunk/release/ccbus-tool.jar .
```

#Make the bin executable

```
[For linux] terminal$ sudo chmod +x ccbus-tool.jar
```

- ✓ Show help:

```
java -jar ccbus-tool.jar -h
```

- ✓ Generate configuration file:

```
java -jar ccbus-tool.jar -g
```

- ✓ Set up configuration file , example below:

```
[Server]
projectPath=/home/user/dev/java/ccbusdemo
relativeSrcPath=src/main/java/
rootPackage=ccbus.demo

[Client]
projectPath=/home/user/dev/react/ccbusdemo
relativeSrcPath=src
#platform=web|mobile
platform=web

[Ccbus]
projectPath=/home/user/dev/java/ccbusConnectLib
relativeSrcPath=src/main/java/
```

[Server] - Maven (Spring) project



[Client] - React project

[Ccbus] - Ccbus Java API

✓ Set up project environments:

```
java -jar ccbus-tool.jar -i
```

This command will create proper structure in your spring and react project.

React project should work with absolute paths.

4 Tutorial

4.1 Compilation

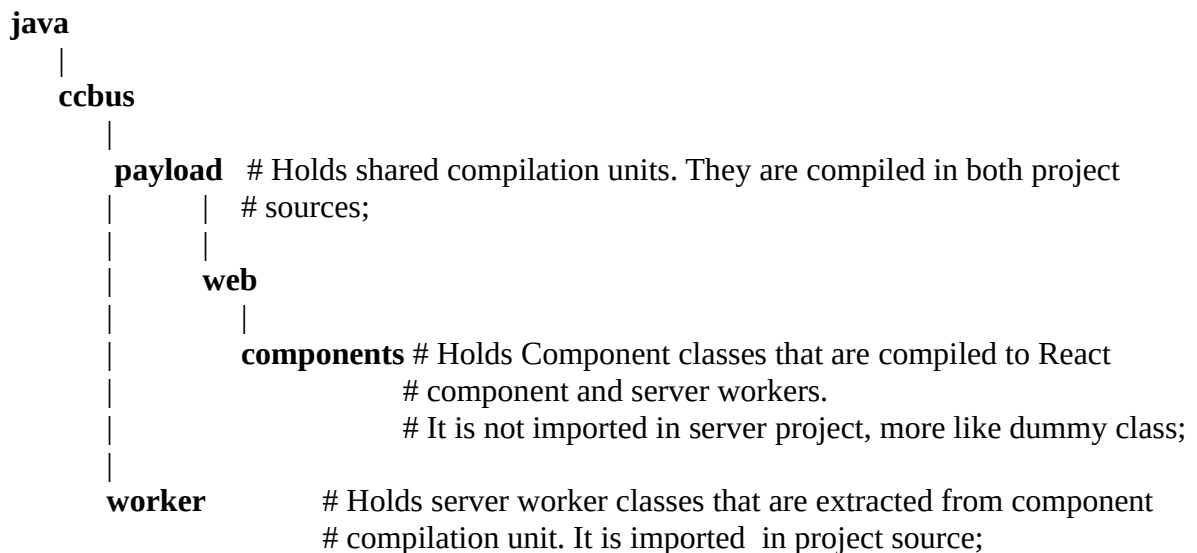
Every project should have separate configuration file. Put compilation tool with configuration file in your project root path.

Compile file with:

```
java -jar ccbus-tool.jar src/main/java/ccbus/demo/ccbus/web/components/EcmaComponentWorker.java
```

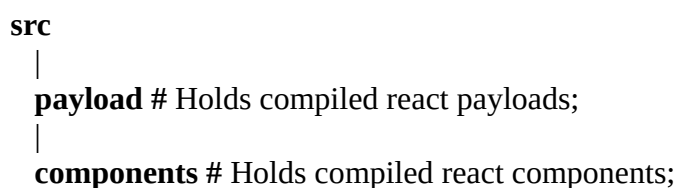
4.2 Java project structure

Bold text presents directory.



4.3 React project structure

Bold text presents directory.





```
|  
package.json # Project configuration file. Set up dependencies...  
|  
jsconfig.json # Project configuration file. Set up project to use absolute path.
```

4.4 **Supported types**

Component class and Payload class support following types:

- All java primitive types.
- String - length, matches, replace, split, join, compareTo
- HashMap - put, containsKey, get, remove
- ArrayList - size, add, get, remove, contains, indexOf, subList
- List - size, add, get, remove, subList
- Function - apply
- BiFunction - apply
- Object - Not available methods
- joda DateTme - toString

4.5 **Demo projects**

For demo project please checkout following projects:

`ccbusdemo/client/ccbusdemo/`

`ccbusdemo/server/ccbusdemo/`

Happy coding :)