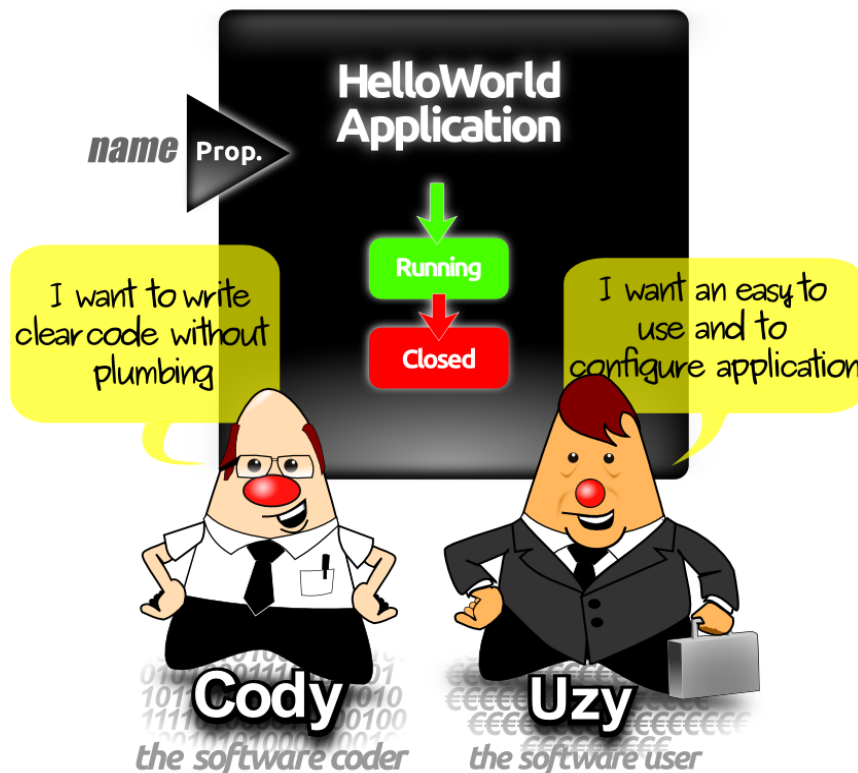


Control your software application with C³

C³ is a Java library providing a component abstraction capacity, as its main feature. The use of components and a component model is highly recommended to structure your software. Before defining your software's component structure, you need to define what the users of your application will be able to do and how they will be able to do it. In order to enable this definition, C³ provides an `Application` class, whose presentation and description is the subject of this article.

An application can be seen as a black box, created by a “coder” and used by a “user”. Their needs are different. The coder wants to write the simplest program that renders the requested functions. The user wants to be able to configure the application easily and to have access to a decent documentation. A very simple example is used throughout this article, the famous “HelloWorld” application, whose functionality is simply to perform greetings. Let us see together what C³ might provide for such an application.



Application class provides the following features :

- **application state-machine** : the software coder can benefit from the simple C³ application state machine. Let's us start with only two states “Running” and “Closed”. By sub-classing the abstract Application class, the compiler will force you to implement the events. At run-time, your events will be called before each state change,
- **application property** : the application shall be configurable by the software user with Java properties (other configuration type might be added in future version). The annotation called `@AppProperty` handles simply this functionality in a declarative manner and C³ does the plumbing for you.
- **generic help with auto-documented mechanism** : no plumbing code needs to be added to display the application name, the “about” information and the properties. C³ discovers automatically all the application features and displays them, when the software user adds the “-help” parameter to its software launch command.

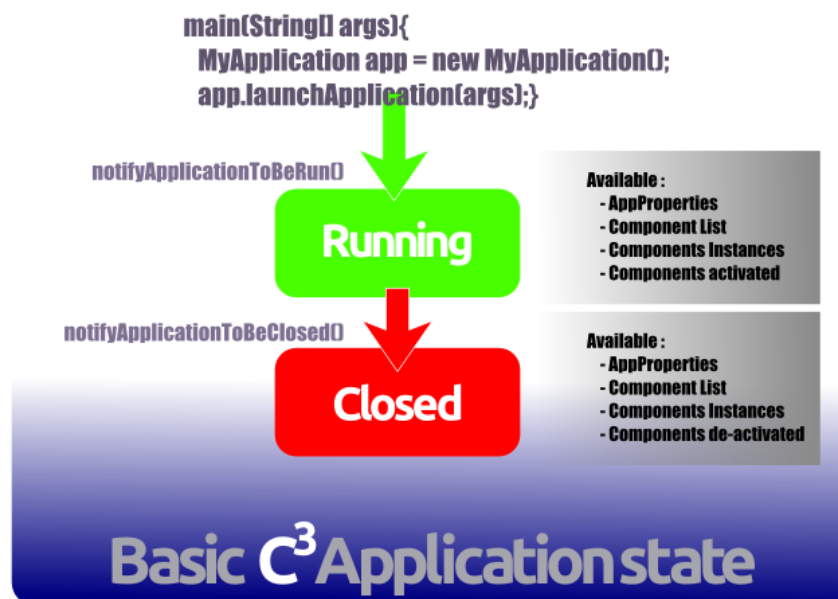


Application annotation

```
@AppInformation (version="1.0", about="This a Hello world application with C3")
public class HelloWorldApplication extends Application {
    ...
}
```

C³ provides an application annotation mechanism in which the version can be set and an “about” text added. This information will be accessible to the software user.

Application state machine



For our simple example, C³ provides an application state-machine with only two states: “Running” and “Closed”. Just before reaching one of the two states, C³ will notify you with the corresponding event: `notifyApplicationToBeRun` or `notifyApplicationToBeClosed`.

The first event is called when all `AppProperties` are ready, and when the components are created. A future paper will focus on components. Basically, this event has to be used to configure and to start your application.

The second event is called when the application is stopped. C³ detects, for you, that your application is closing and notifies this. The Event can be used to properly close resources such as a socket, a database, a file, etc.

```
@AppInformation (version="1.0", about="This a Hello world application with C3")
public class HelloWorldApplication extends Application {
    @Override
    public void notifyApplicationToBeRun(String applicationDescription, String
componentsDescription) {
        System.out.println("Application started");
    }

    @Override
    public void notifyApplicationToBeClosed() {
        System.out.println("bye bye, application closed");
    }
}
```

AppProperty annotation

Java language provides system property that can be set either on Java command line with “-D” prefix or by loading a property file. A property is a key/value pair, where each value is accessed through the associated key.

C³ provides annotations in order to automatically link a property to your class application member. The principle is basic, by being declarative. This way the code is more functional and less technical. What you have to do is to add, as class member, the needed configuration inside your application class. For instance, let's say that an application needs to be configured with a name to greet. All you have to do is to add, inside your own Application's sub class, a member like :

```
@AppProperty
String name = "World";
```

This is the default value and it will be used, if no « name » property set

The above piece of code can be read as: this is a property called “name” and its default value is “world”. Automatically the “name” system property is available to configure your application.

Min and max value can be added for some numerical properties

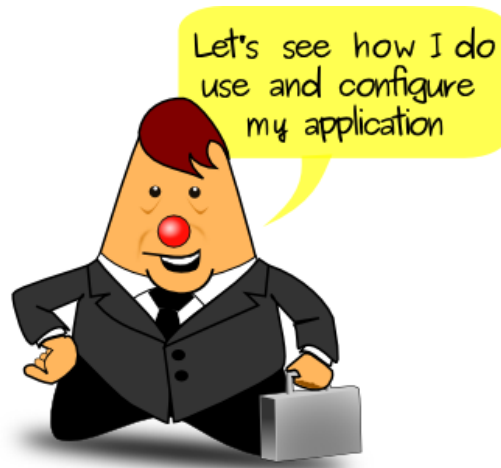
```
@AppProperty(comment = "primitive value", min = 3, max = 7)
int intValue1 = 5;
```

Application member types supported are listed in the following table :

Type	comment
int	<i>min and max values can be defined</i>
java.lang.Integer	<i>min and max values can be defined</i>
double	<i>min and max values can be defined</i>
java.lang.Double	<i>min and max values can be defined</i>
java.lang.String	
java.awt.Color	<i>hexadecimal color value</i>
java.nio.file.Path	<i>check if the file exists, if not raise and error on system.error</i>

Extra types can be easily added by implementing the generic interface `ObjectFactoryFromStringValue` which shall return the defined generic type from a string. When the factory is written just add it with the following method at your application creation.

```
public void addNewObjectFactory(Class<?> objectFactoryType,
ObjectFactoryFromStringValue<?> factory)
```



Application command line

C³ provides default command parameters such as the keyword “-help”, “-h”, “--help” or “--h”. When called with one of this parameters, C³ writes on the system output :

- the application name of the application,
- “about” information,
- version of the application
- the list of properties, with their types.

You can either use your own code typed from the samples or download the ready to use classes

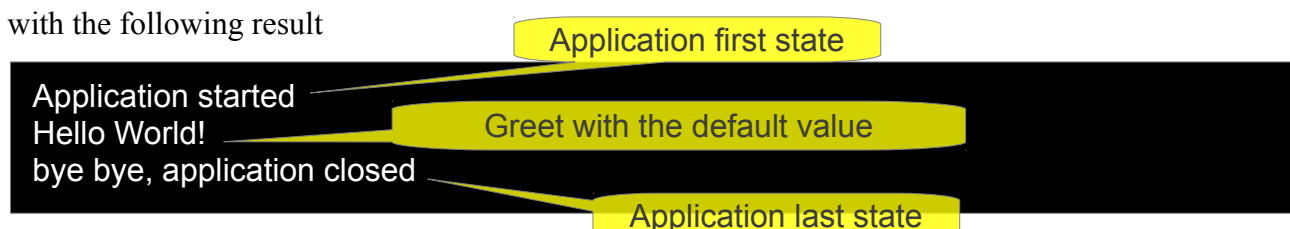
At <http://c3.capcaval.org/download.html> download :

- C3.jar
- HelloWorldApplication.zip

Unzip the file, and go inside the root directory. You need to run with a Java7 jre. After launch the following command line :

```
java -cp ../c3.jar c3sample.HelloWorldApplication
```

with the following result



In order to know the available application properties used the -help parameter as follows :

```
java -cp ../c3.jar c3sample.HelloWorldApplication -help
```

the result should be :

```
*****
Application Name : HelloWorldApplication
Version : 1.0
About : This a Hello world application with C3
-----
The available properties are :
  name
  comment : Application which salute anyone.
  type : String
*****
```

Application name

Application Version

Application information

This application got only one Property named name

Configure your application

Of course the default value, “world” in our case can be changed by using the “name” property. Just type the following command line:

```
java -cp ../c3.jar -Dname="Uzy" c3sample.HelloWorldApplication
```

Use « -D » property followed by the property name

The result should be :

```
Application started
Hello Uzy!
bye bye, application closed
```

« world » default value is not used anymore, replaces by the configured one

Thanks to Laurent, Yannick and Olivier for their review.

Mik Arber

Founder of C³ : c3.capcaval.org

JUG Leader : finistjug.fr

Blog : miksblog.capcaval.org