



# gogo shell

<http://felix.apache.org/site/rfc-147-overview.html>

OSGi Command Line Interface (draft)







# Was ist gogo shell?

```
g! lb -s | grep felix.gogo
  1|Active      | 1|org.apache.felix.gogo.shell (0.10.0)
  2|Active      | 1|org.apache.felix.gogo.runtime (0.12.1)
  3|Active      | 1|org.apache.felix.gogo.command (0.14.0)
```

felix:

install/uninstall, start/stop/resolve, refresh/update  
cd, ls, help, log  
lb, inspect, headers, which, bundlelevel/frameworklevel

gogo:

cat, echo, grep, set, sh/gosh/source, telnetd  
type, format, getopt, each, until, not, tac

obr:

repos add refresh remove list  
list, deploy, info, javadoc/source

⌈⌋⌈⌋' qeb⌋o⌋' ⌈⌋to' ⌋a⌋a⌋oc\⌋o⌋rc⌋  
⌋ebos ⌋q⌋ ⌋e⌋⌋e⌋⌋ ⌋e⌋o⌋e ⌈⌋⌋⌋





# Konzept?

- Command

```
g! cat .project | grep name.*go      # Kommentar
```

- Session Variable \$... oder {...}

```
g! variable = string | 'string with space'      # String
g! variable = 1 | 1.0 | true                     # Long | Double | Boolean
g! list = [1, '2', 3.0, true]                    # [Long, String, Double, Boolean]
g! map = [Jan=1 'Feb'=true Mar=3.0]              # [String=Long, .=Boolean, .=Double]
g! echo $variable ${list}                       # $.. == ${..}
```

- Rückgabewert

```
g! xml = ( cat .project | tac )
```

- Macro/Closure

```
g! macro = { echo Bla $args blub }
g! macro bla
Bla bla blub
```





# Reflection

- Exceptionhandling

```
g! $exception printstacktrace      # ((Exception) e).printStackTrace();  
  
g! ( $exception stacktrace ) 0      # ((Exception) e).getStackTrace()[0]  
g! ( ( ( $exception stacktrace ) 0 ) class ) name # ...  
...((Exception) e).getStackTrace()[0].getClass().getName()
```

- Ein bisschen mehr

```
g! macro = { echo Bla $args blub }  # Closure macro = `echo Bla $args blub`  
g! map = ['1'=Jan '2'=Feb '3'=Mar]  # Map<String, String> map = ...  
g! each ( ( $map ) entryset ) { macro $it } # ...  
... for (Object entry : map.entrySet()) macro.execute(entry);  
Bla 1=Jan blub  
Bla 2=Feb blub  
Bla 3=Mar blub  
null                                # 3x wegen echo-command
```





# FirstCommand

- FirstCommand

```
public class FirstCommand {  
    public void macro(Object arg){  
        System.out.println("macro#obj: Bla " + arg + " blub");  
    }  
}
```

- Als Declarative Service oder programmatisch

```
<component name="FirstCommand">  
    <implementation class="FirstCommand"/>  
    <service>  
        <provide interface="FirstCommand"/>  
    </service>  
    <property name="osgi.command.scope" type="String" value="itemis"/>  
    <property name="osgi.command.function" type="String" value="macro"/>  
</component>
```

```
public class Activator implements BundleActivator {  
    @Override  
    public void start(BundleContext context) throws Exception {  
        Dictionary<String, Object> dict = new Hashtable<>();  
        dict.put("osgi.command.scope", "itemis");  
        dict.put("osgi.command.function", new String[] {"macro"});  
        context.registerService(FirstCommand.class, new FirstCommand(), dict);  
    }  
}
```



# Risks

- Polymorphy & Reflection ( `polymorph cmd-order` )
- `Cmd-Return` vs `System.out/System.in`
- `System.out` reading in ,|' mode
- Bad-help
- Missing eclipse-console tooling ( tab-, history-support)
- OBR: duplicate Capability/Requirement Impl





# Links

- GitHub: <https://github.com/jwausle/gogo.shell/blob/master/gogo.shell>
- Overview: <http://felix.apache.org/site/rfc-147-overview.html>
- Cmds: <https://github.com/jwausle/gogo.shell/blob/master/gogo.shell.doc/help/README.MD>
- OSGi-Spec: <http://www.osgi.org/download/osgi-4.2-early-draft.pdf>
- Cmd-History-UpdateSite: <https://github.com/jwausle/de.jwausle.support.org.eclipse.ui.console/raw/master/de.jwausle.support.org.eclipse.ui.console.update.site/update-site/2.1.0>