

# Japanese Dynamic Groovy

Groovy GroovyObject ExpandoMetaClass

- Using invokeMethod and getProperty
- Using methodMissing and propertyMissing
- Evaluating the MetaClass runtime
- Using ExpandoMetaClass to add behaviour
- Customizing MetaClass for a single instance
- Runtime mixins

Japanese Compile-time Metaprogramming - AST Transformations

```
class Dog {
    def bark() { println "woof!" }
    def sit() { println "(sitting)" }
    def jump() { println "boing!" }
}

def doAction( animal, action ) {
    animal."$action"() //action name is passed at invocation
}

def rex = new Dog()

doAction( rex, "bark" ) //prints 'woof!'
doAction( rex, "jump" ) //prints 'boing!'
```

```
def max(int i1, int i2) {
    Math.max(i1, i2)
}
def numbers = [1, 2]
assert max( *numbers ) == 2
```

GString

```
someObject."$methodName"(*args)
```