How to change a Jenkins Declarative Pipeline environment variable?

I'm trying to create some Docker images. For that I want to use the version number specified in the Maven pom.xml file as tag. I am however rather new to the declarative Jenkins pipelines and I can't figure out how to change my environment variable so that VERSION contains the right version for all stages.

This is my code

```
#!groovy

pipeline {
    tools {
        maven 'maven 3.3.9'
        jdk 'Java 1.8'
    }
    environment {
        VERSION = '0.0.0'
    }

    agent any

    stages {
        stage('Checkout') {
            steps {
                git branch: 'master', credentialsId: '290dd8ee-2381-4c5b-8d33-5631d03ee7be', url: 'git@gitlab.crosslang.local:company/SOME-API.git'
                sh "git clean -f && git reset --hard origin/master"
            }
        }
        stage('Build and Test Java code') {
            steps {
                script {
                    def pom = readMavenPom file: 'pom.xml'
                    VERSION = pom.version
                }
                echo "${VERSION}"
                sh "mvn clean install -DskipTests"
            }
        }
        stage('Build Docker images') {
            steps {
                dir('whales-microservice/src/main/docker'){
                    sh 'cp ../../../target/whales-microservice-${VERSION}.jar whales-microservice.jar'
                    script {
                        docker.build "company/whales-microservice:${VERSION}"
                    }
                }
            }
        }
    }
}
```

jenkins jenkins-pipeline

asked May 9 '17 at 13:59
Bram Vandewalle
500 ● 1 ● 7 ● 18

What is exactly your problem? The VERSION in stage 'Build Docker images' doesn't contain the version number which was assigned in `VERSION = pom.version` ? – haschibaschi May 9 '17 at 18:23

That is indeed the problem. It is only available in the 'Build and Test Java code' stage. – Bram Vandewalle May 10 '17 at 7:02

1 Answer

The problem is the single quote of the statement

```
sh 'cp ../../../target/whales-microservice-${VERSION}.jar whales-microservice.jar'
```

single quotes don't expand variables in groovy: http://docs.groovy-lang.org/latest/html/documentation/#_string_interpolation

so you have to double quote your shell statement:

```
sh "cp ../../../target/whales-microservice-${VERSION}.jar whales-microservice.jar"
```

answered May 10 '17 at 13:42
haschibaschi
1,281 ● 11 ● 21