Continuous Delivery for Microservices with SBT and Docker

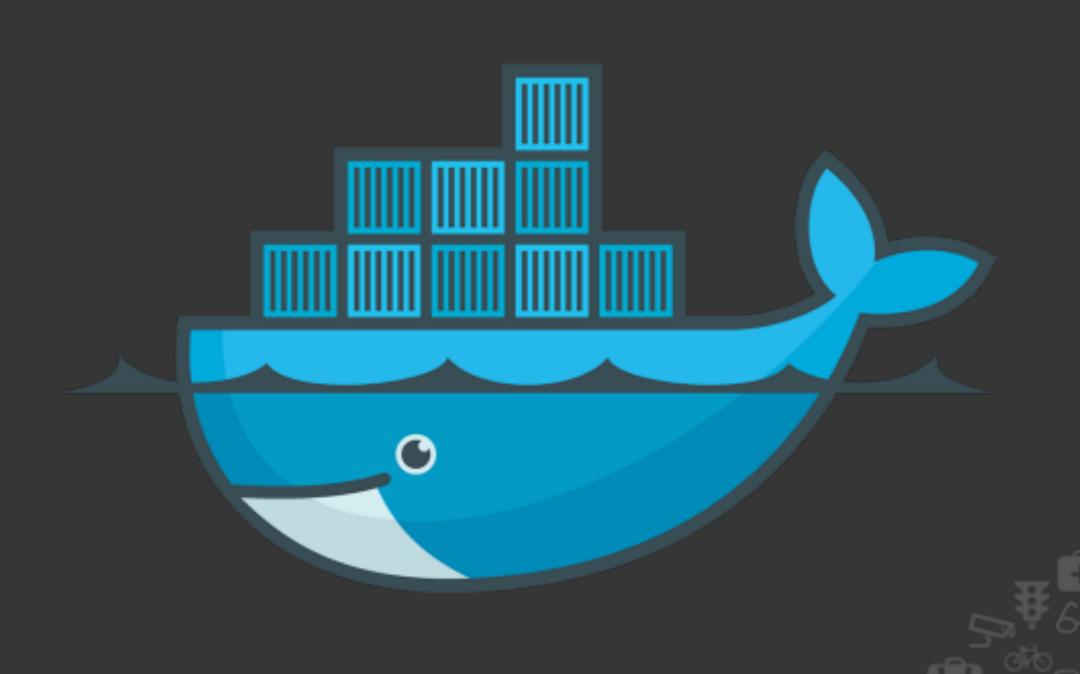
Patterns behind a mnubo internal revolution



Continuous Delivery



Docker



Company SBT plugin

Latest plugin

```
val pluginArtifactoryUrl = new URL(
   "<artifactory URL>/com/mnubo/mnubo-plugin_2.10_0.13/maven-metadata.xml")
val latestPluginVersion =
   Using.urlInputStream(pluginArtifactoryUrl) { stream =>
        """<latest>([\d\.]+)</latest>""".r
        .findFirstMatchIn(IO.readStream(stream))
        .get
        .group(1)
}

resolvers += "Mnubo repository" at "<artifactory URL>"
addSbtPlugin("com.mnubo" % "mnubo-plugin" % latestPluginVersion)
```

Automating the build

```
releaseVersion
                             := identity,
                             := { (ver: String) =>
releaseNextVersion
  sbtrelease
    .Version(ver)
    .map(_.bumpBugfix.string)
    .getOrElse(sys.error("No versions are set!"))
releaseProcess
                             := Seq[ReleaseStep](
  setupRemoteTracking,
  inquireVersions,
  runTest,
  setReleaseVersion,
  publishArtifacts,
  tagWithOverwriteRelease,
  setNextVersion,
  commitNextVersion,
  pushChanges
```

Building Docker images

sbt-native-packager









sbt-docker





Latest version

```
libraryDependencies ++= Seq(
  "com.mnubo" %% "some-lib" % "[1.0.0,)" changing()
  //...
),
```

Managing data stores



https://github.com/mnubo/dbevolv

Contact and questions



julien.letrouit@gmail.com



@jletroui