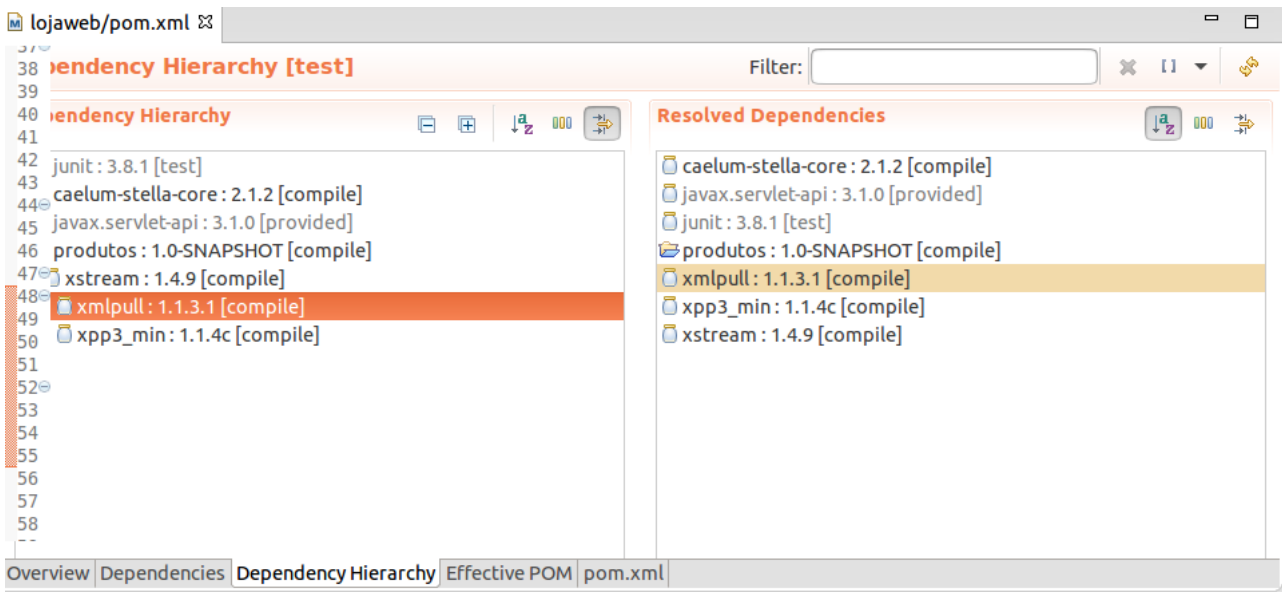


Excluindo dependências

Em "Java Resources > Libraries > Maven Dependencies", há algumas bibliotecas que não declaramos como dependências no `pom.xml`. Elas são resolvidas pelo Maven, pois cada dependência que adicionamos pode conter outras.

No Eclipse, é possível consultar a hierarquia de dependências clicando na aba "Dependency Hierarchy". Podemos ver, por exemplo, que o nosso projeto depende de `produtos`, e que este, por sua vez, depende do `xstream`. O `xstream` depende do `xmlpull` e do `xpp3_min`:



Também é possível analisar a hierarquia de dependências utilizando o comando `mvn dependency:tree`:

```
lucas@lucas-Inspiron-5458:~/Projects/produtos$ mvn dependency:tree
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building produtos 1.0-SNAPSHOT
[INFO] -----
[INFO]
[INFO] --- maven-dependency-plugin:2.8:tree (default-cli) @ produtos ---
[INFO] br.com.alura.maven:produtos:jar:1.0-SNAPSHOT
[INFO] +- junit:junit:jar:4.12:test
[INFO] | \- org.hamcrest:hamcrest-core:jar:1.3:test
[INFO] \- com.thoughtworks.xstream:xstream:jar:1.4.9:compile
[INFO]    +- xmlpull:xmlpull:jar:1.1.3.1:compile
[INFO]    \- xpp3:xpp3_min:jar:1.1.4c:compile
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.378 s
[INFO] Finished at: 2016-03-29T10:57:29-03:00
[INFO] Final Memory: 13M/212M
[INFO] -----
lucas@lucas-Inspiron-5458:~/Projects/produtos$
```

Em alguns momentos, podemos querer excluir uma determinada biblioteca que seja dependência de uma biblioteca do nosso projeto. Um motivo é que as dependências em uma das que estamos utilizando pode ter sido declarada errada, por exemplo.

Exclua o `xstream` do `produtos` no projeto `lojaweb`.

