

# Module renaming

Motivation:	Get nightly site and javadoc builds
Contact:	<a href="#">Martin Desruisseaux</a> , <a href="#">Cédric Briçon</a>
Tracker:	<a href="#">GEOT-995</a>
Attachment:	none

## Proposal Complete.

The modules have now been renamed. Jars often start with 'gt-' as in 'gt-main'.

## Description

Maven behaves better when we stick to its default behavior. Among other it expects module names to match directory names. Previous module names have a `gt2-` prefix in front of them, which isn't present in directory name. The consequence is broken URL in the generated web site. It is possible to workaround by providing explicitly `<scm>` or `<url>` elements in every `pom.xml` files, but this is tedious, error-prone and we always end up with modules with wrong settings. Given the large amount of modules in GeoTools, less manual settings we have, better it will be.

However if we blindly remove `gt2-` prefix in every module names, we get potential conflict in JAR names like GeoTools `h2-2.5-SNAPSHOT.jar` with the `h2-1.0-SNAPSHOT.jar` database driver: only the version number distinguish them, which is fragile. In order to keep some prefix in the JAR name, a possible approach is to use the `<finalName>` element in `pom.xml` like below:

```
<build>

<finalName>gt-${artifactId}-${version}</finalName>

</build>
```

However as of Maven 2.0.8, prefix specified that way are not propagated to the online repository and in the JAR downloaded by users. A custom plugin defined in GeoTools, `jar-collector`, can take care of copying the JAR in a target directory with the right prefix. Every users (**not** just the ones building GeoTools) can get this flat directory providing they put the following in their own `pom.xml`:

```

<build>
  <plugins>
    <plugin>
      <groupId>org.geotools.maven</groupId>
      <artifactId>jar-collector</artifactId>
      <version>2.5-SNAPSHOT</version>
      <executions>
        <execution>
          <goals>
            <goal>collect</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

## Original proposal

- Remove the `gt2-` prefix from all module names.
- Declare a prefix in `<finalName>` elements instead.

## Pros

- The most straightforward module and directory names we can get.
- No redundancy between `<artifactId>` and `<groupId>` (the need to prefix `<artifactId>` despite different `<groupId>` suggests that something is not working with Maven model...)
- Gives us an opportunity to distinguish between library, plugin, *etc.* through different prefix (e.g. `gt-lib-`, `gt-plg-`, *etc.*).

## Cons

- By default, JARs downloaded from Maven by users do not include the prefix, thus leading to potential JAR filename conflicts. The `jar-collector` mentioned above solve those conflicts, but not every users will accept this non-standard plugins.

## Alternative proposal

- Reinsert the `gt-` prefix (without the 2 character), but **only** for leaves (not for parent modules containing child

modules).

- Insert the `gt-` prefix in directory name as well. We would get for example `modules/library/gt-metadata` (note that the `gt-` prefix appears only once).

Because only leaves produce JAR, it doesn't matter if the parents are not prefixed.

### **Pros:**

- Remove the need for an explicit `<finalName>` declaration, so we get a total matching between our naming scheme and Maven defaults.
- We can distinguish easily between modules that contains a JAR and modules that contains other modules, since only the former are prefixed by `gt-`.
- May be the less trouble-prone approach for a quiet future.

### **Cons:**

- Would force us to rename about 80 directories in GeoTools code base.
- Users will need to remount the GeoTools project in their IDE.
- More verbose directory names, e.g. `modules/library/gt-metadata`, a little bit tedious for those who use a lot the command line.
- We lost the cabability to distinguish between library, plugins, *etc.* from the JAR filename.