

di.uniba.map.b.adventure.impl.  
OpenObserver.update

di.uniba.map.b.adventure.impl.  
ReadObserver.update

di.uniba.map.b.adventure.impl.  
UseObserver.update

di.uniba.map.b.adventure.parser.  
ParserOutput.getInObject

```
graph LR; A[di.uniba.map.b.adventure.impl.OpenObserver.update] --> D[di.uniba.map.b.adventure.parser.ParserOutput.getInObject]; B[di.uniba.map.b.adventure.impl.ReadObserver.update] --> D; C[di.uniba.map.b.adventure.impl.UseObserver.update] --> D;
```

The diagram illustrates a dependency or data flow. On the left, there are three white rectangular boxes with black borders, each containing a fully qualified class name and a method name. Blue arrows originate from the right side of each of these three boxes and point towards a single gray rectangular box on the right. This gray box also contains a fully qualified class name and a method name. The arrows indicate that the three methods on the left (OpenObserver.update, ReadObserver.update, and UseObserver.update) all depend on or interact with the ParserOutput.getInObject method.