Towards Variability-Aware Smells

Lack of Annotation

DEFINITION:

`Lack of Annotation` variability-aware smell is identified when there is annotation omission in the source code for optional features defined in the feature model.

EXAMPLE:

Considering the feature model and the optional features *AStorage* and *LLStorage* and *Locking* in Figure 7.2, Listing 7.4 shows a example of `Lack of Annotation` smell, because the methods *push(E e, Lock lock)* and *pop(Lock lock)* implement the feature *Locking*, however they are not annotated for the feature *Locking*.

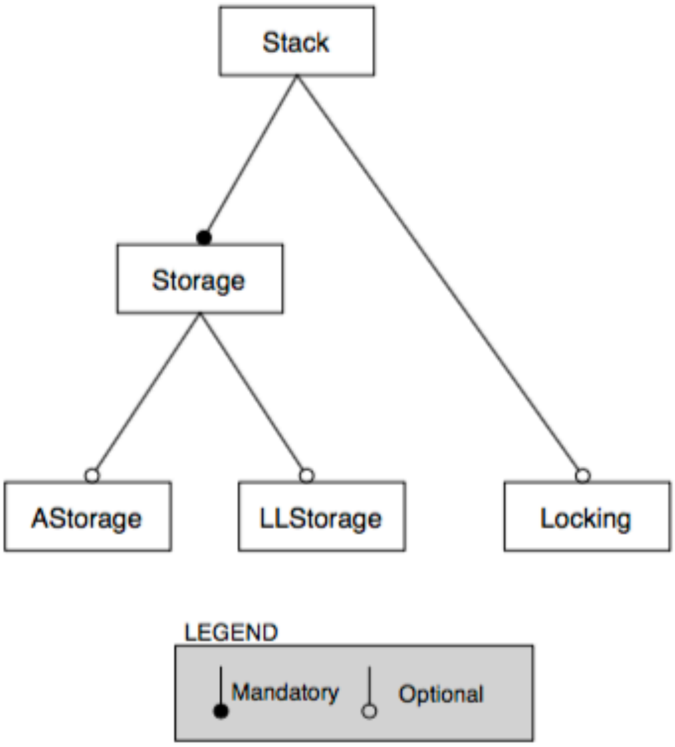
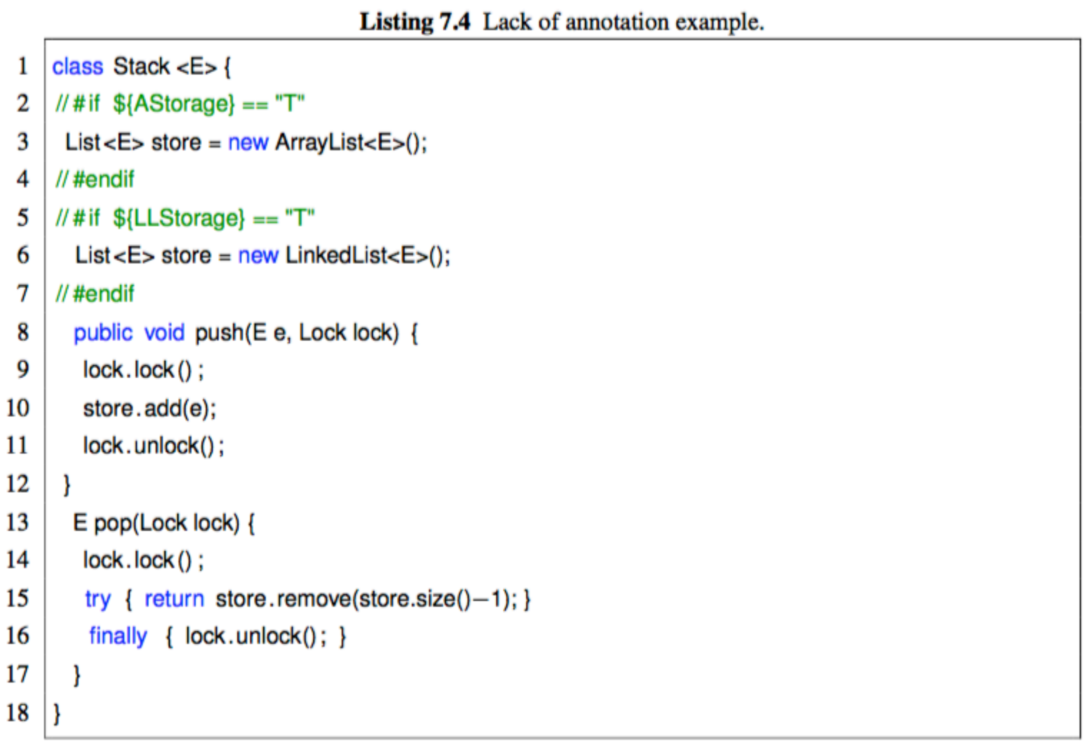


Figure 7.2 Bad annotation example.



Listing 7.4 Lack of annotation example.

```
class Stack <E> {
//#if ${AStorage} == "T"
 List<E> store = new ArrayList<E>();
//#endif
//#if ${LLStorage} == "T"
  List<E> store = new LinkedList<E>();
//#endif
   public void push(E e, Lock lock) {
    lock.lock();
    store.add(e);
    lock.unlock();
  }
   E pop(Lock lock) {
    lock.lock();
    try { return store.remove(store.size()−1); }
     finally { lock.unlock(); }
   }
}
```

PROBLEM:

This smell is a barrier to feature implementation mapping and it impacts negatively in program comprehension, maintenance and consequently in evolution. `Lack of Annotation` smell is very hard to detect, because it requires a deep knowledge about the business domain implemented.