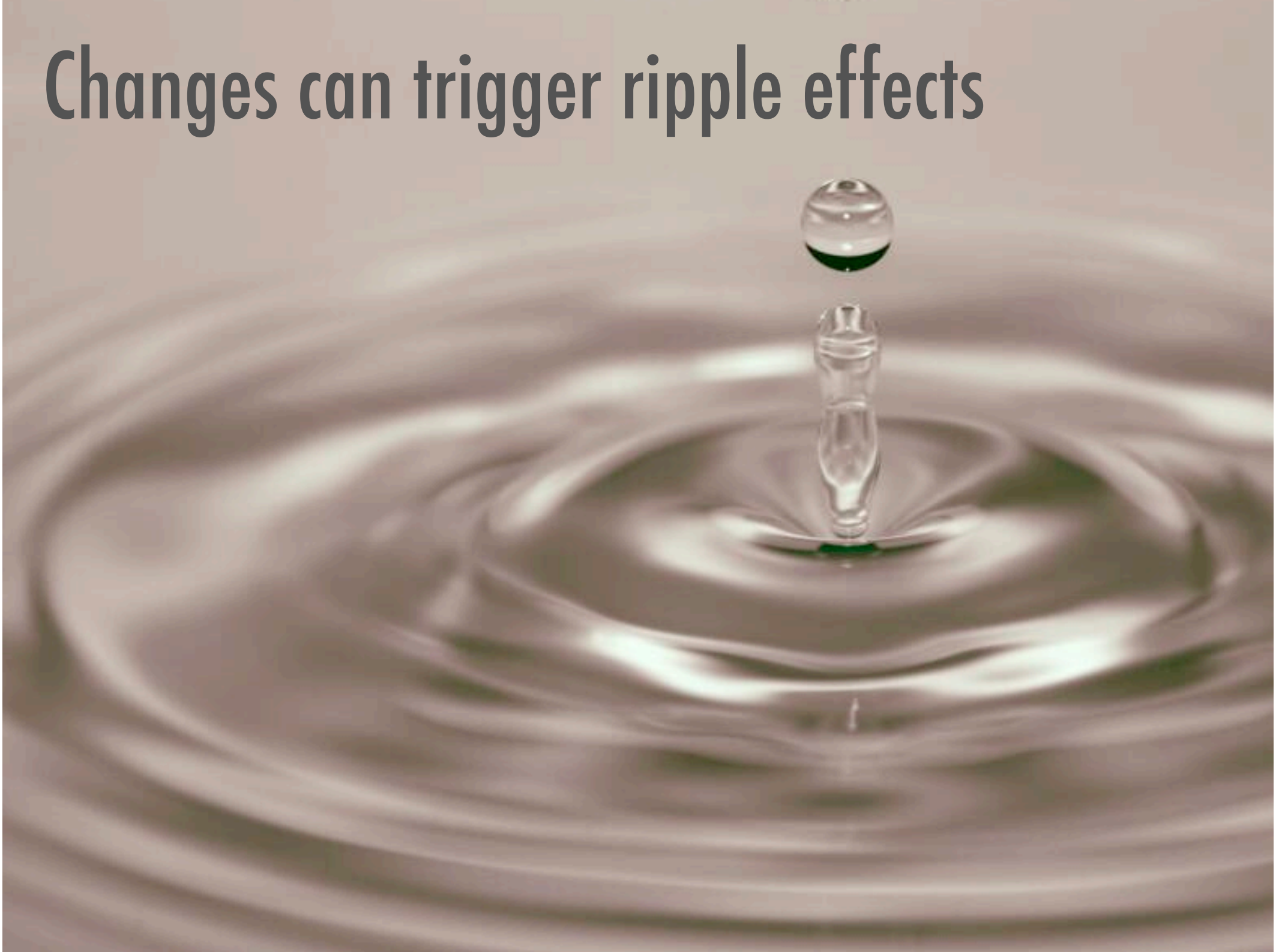
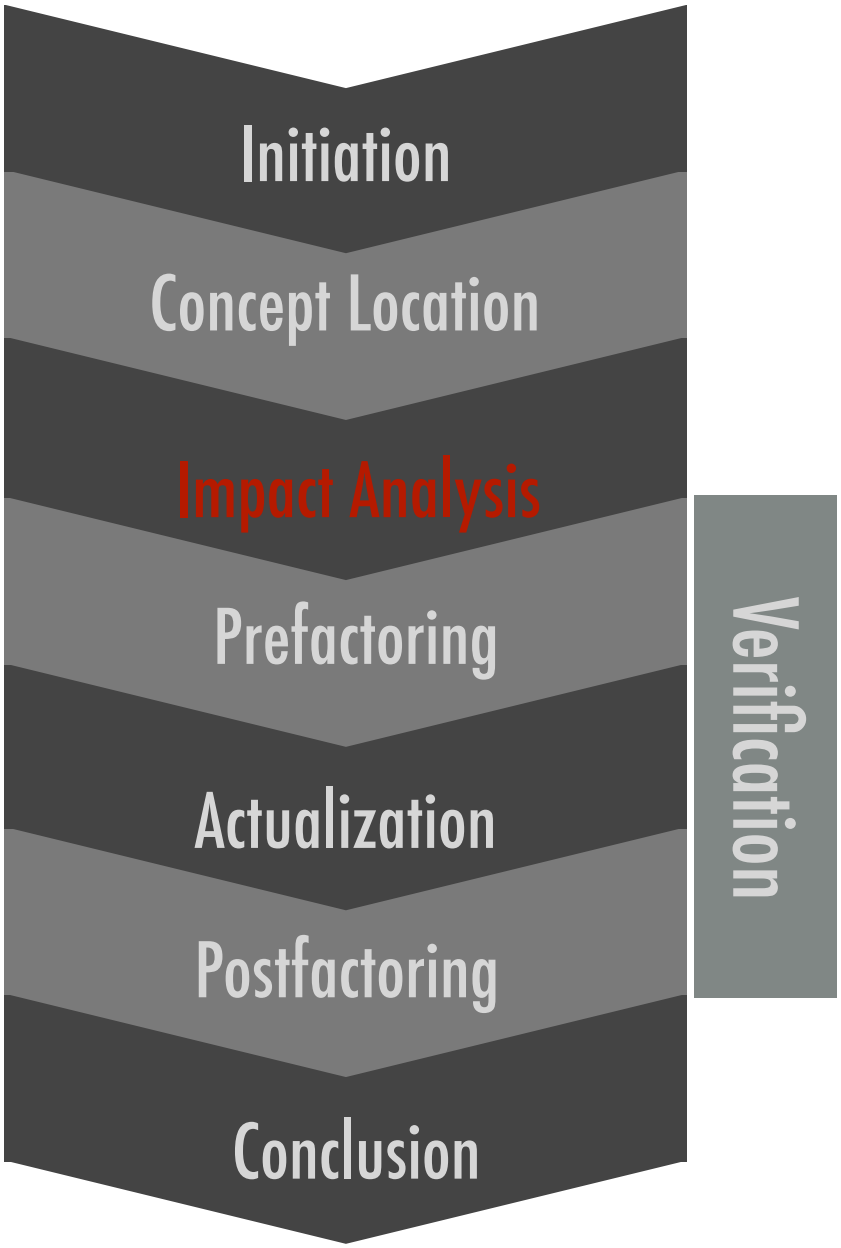
Impact Analysis

Prof. Romain Robbes

Changes can trigger ripple effects



Impact analysis estimates actual changes



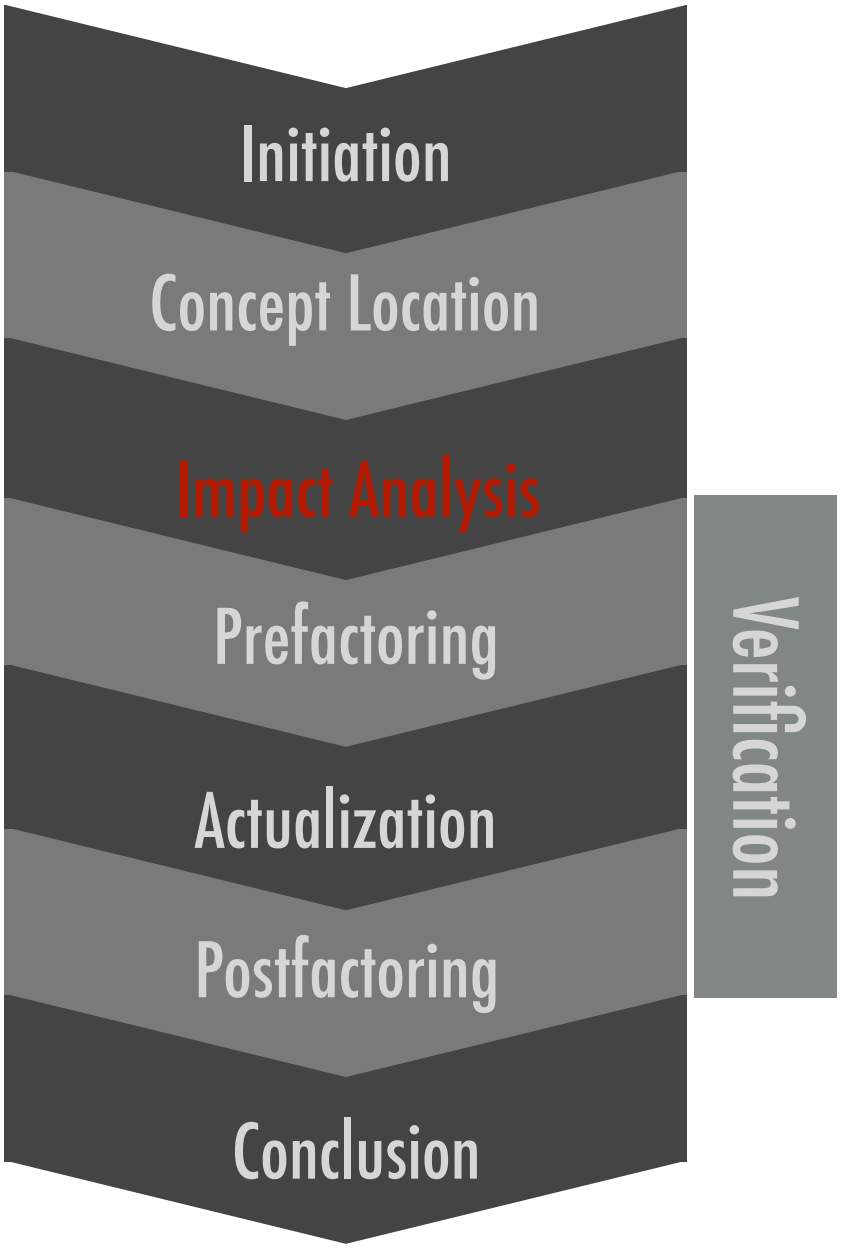
If the change is not localized, the programmer determines which other modules are affected by the change.

Outline

From concept location to impact analysis
Class interaction graphs
The impact analysis process

From concept location to impact analysis

Impact analysis (IA):

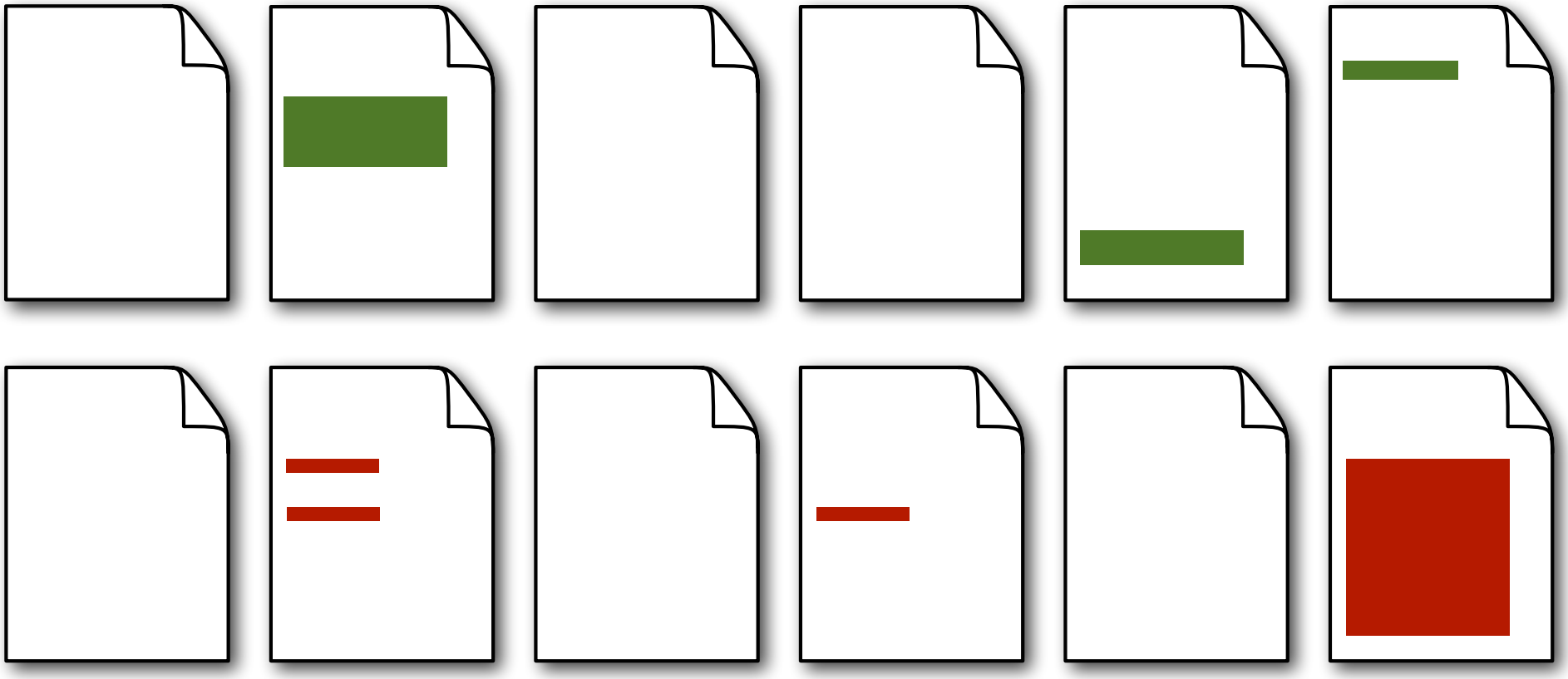


Determines the **strategy** and **impact** of the change

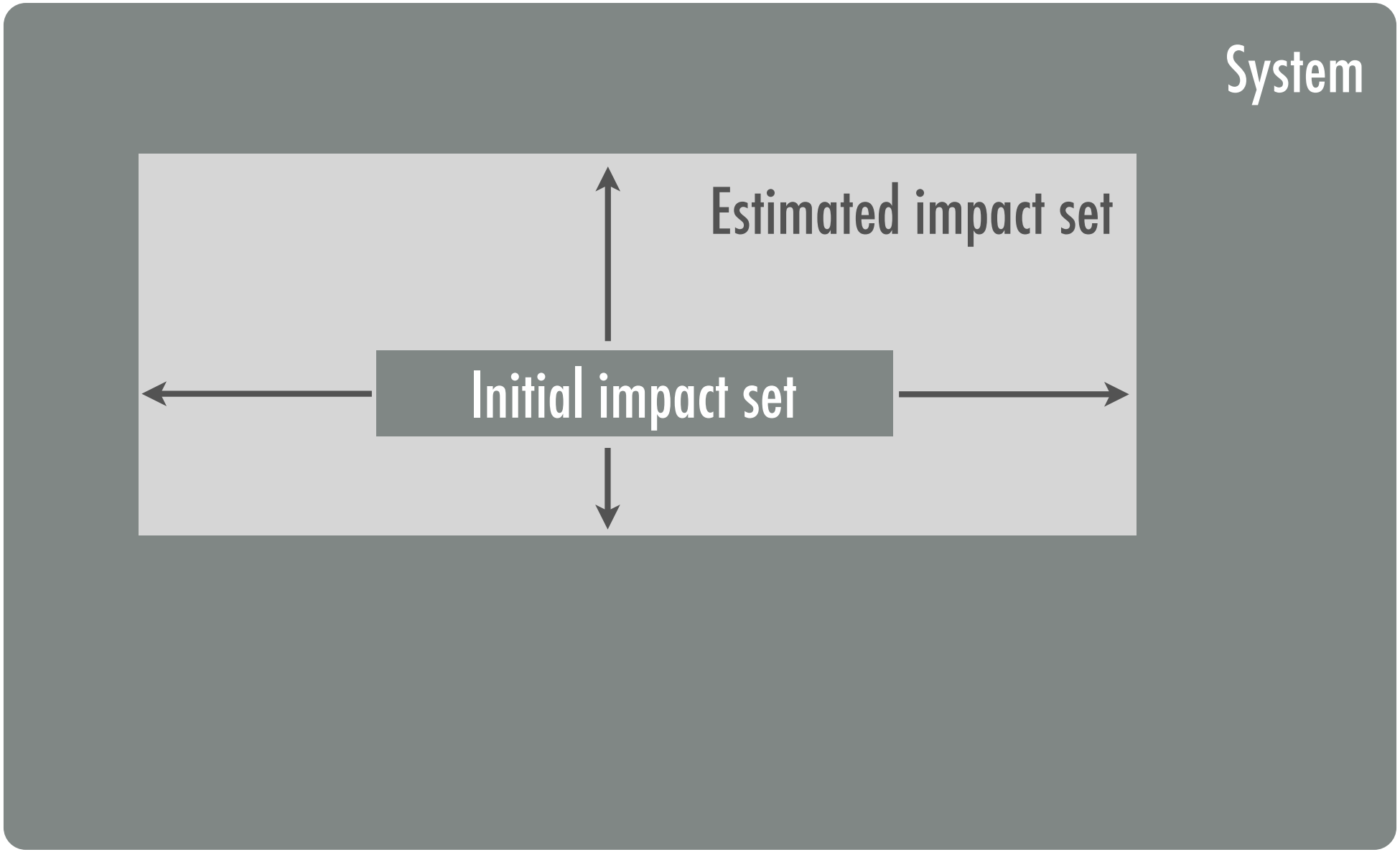
Classes identified in concept location make up the **initial** impact set

Class dependencies are analyzed; impacted classes are added to the **estimated** impact set

Concept location vs Impact analysis

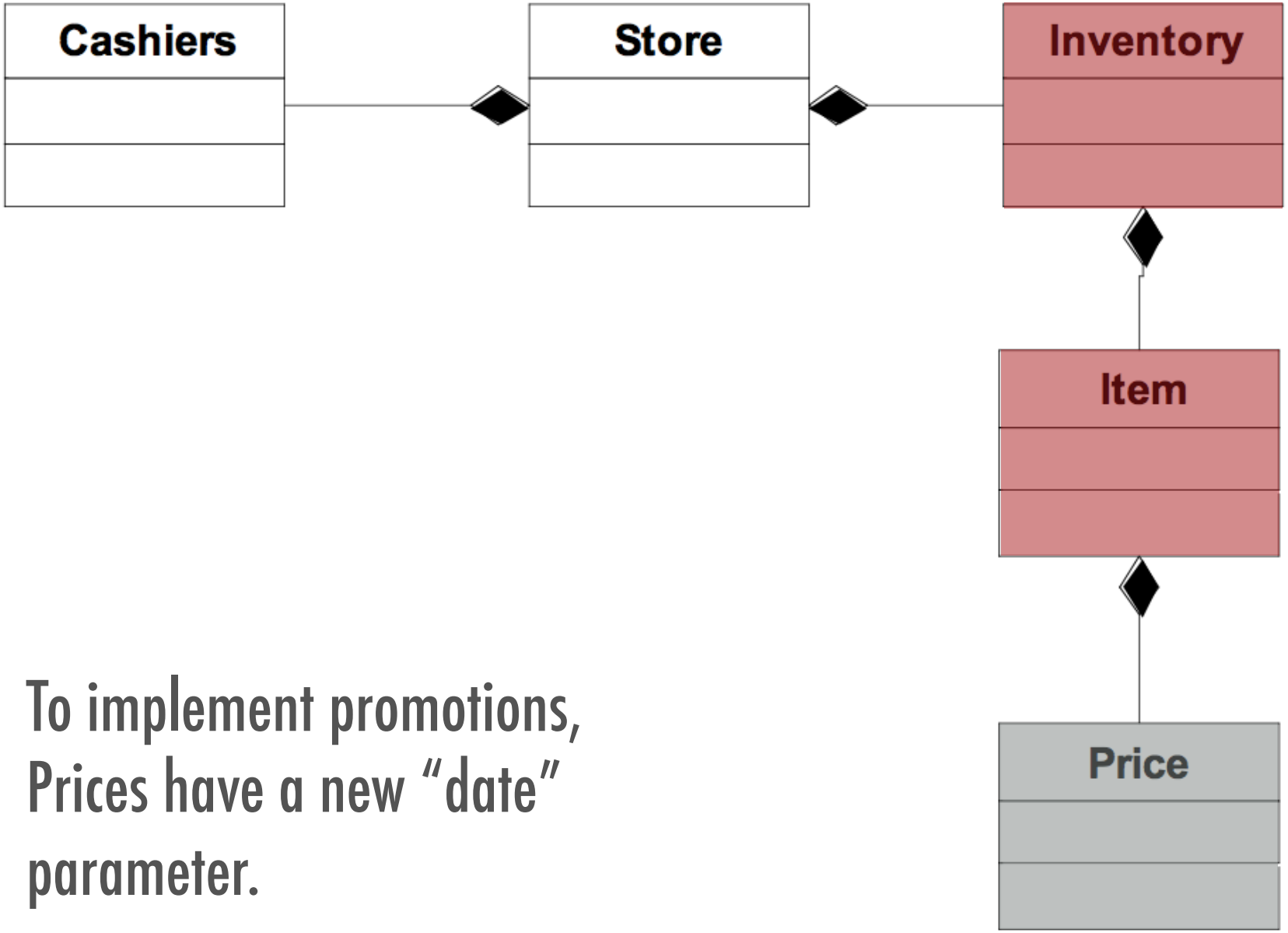


The impact set of a change request



Estimated vs actual impact set

Example: point-of-sale application



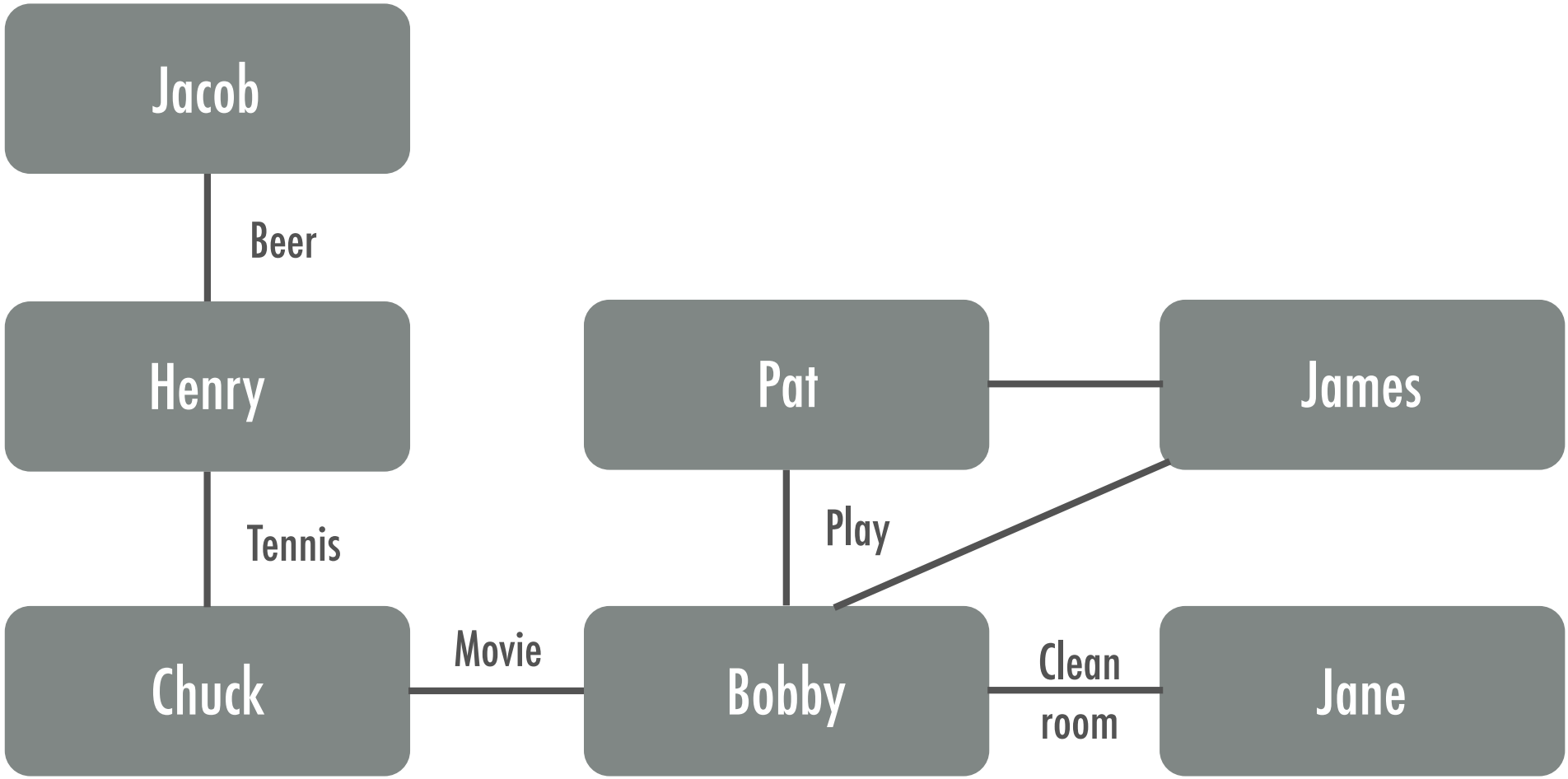
To implement promotions, Prices have a new "date" parameter.

Concept location vs impact analysis

Concept location is repeated until a suitable starting point is found

Impact analysis is repeated until ... when?

There's a change in Jacob's schedule



Class interaction graphs

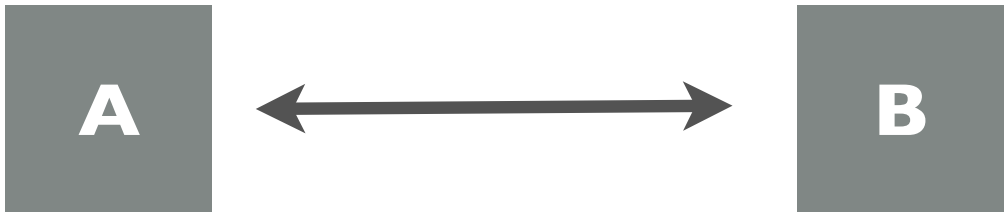
Class interactions

Two classes **interact** if they have something in **common**

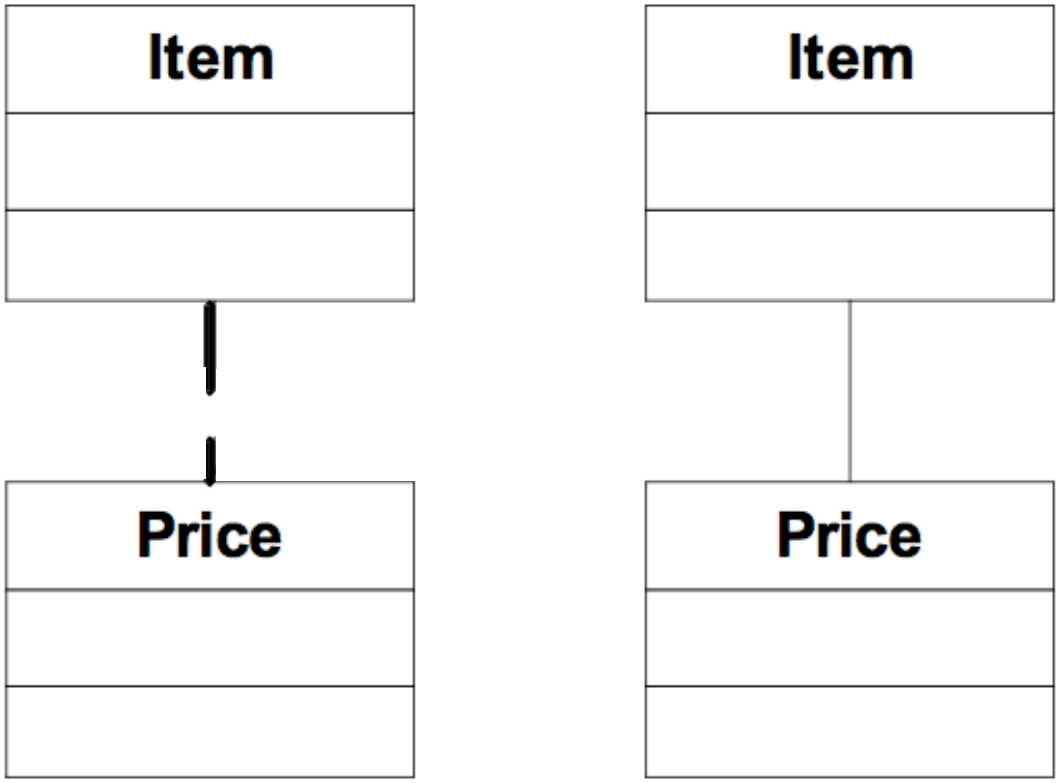
One **depends** on the other; there is a contract between them

They **coordinate**: share the same coding, schedule, etc.

Interactions propagate change in **both directions**



Class interaction vs class dependency graph



Dependency

Interaction

However, indirect interactions are possible (ex: mailman)

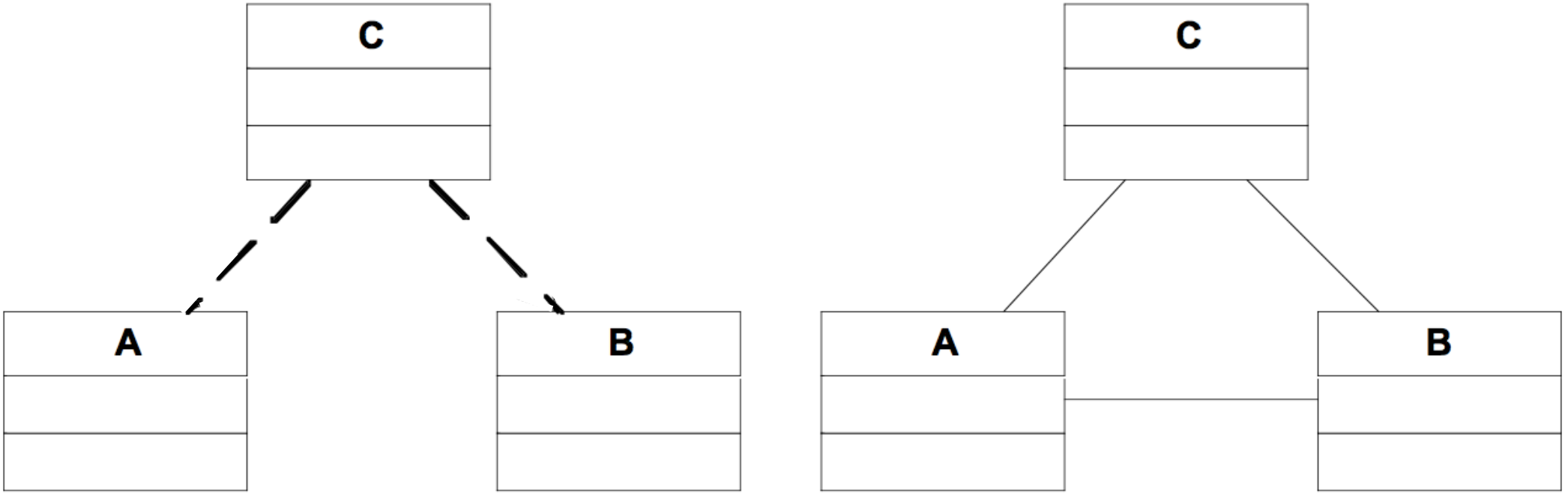
Global variables

Shared assumptions ("blue" is 1)

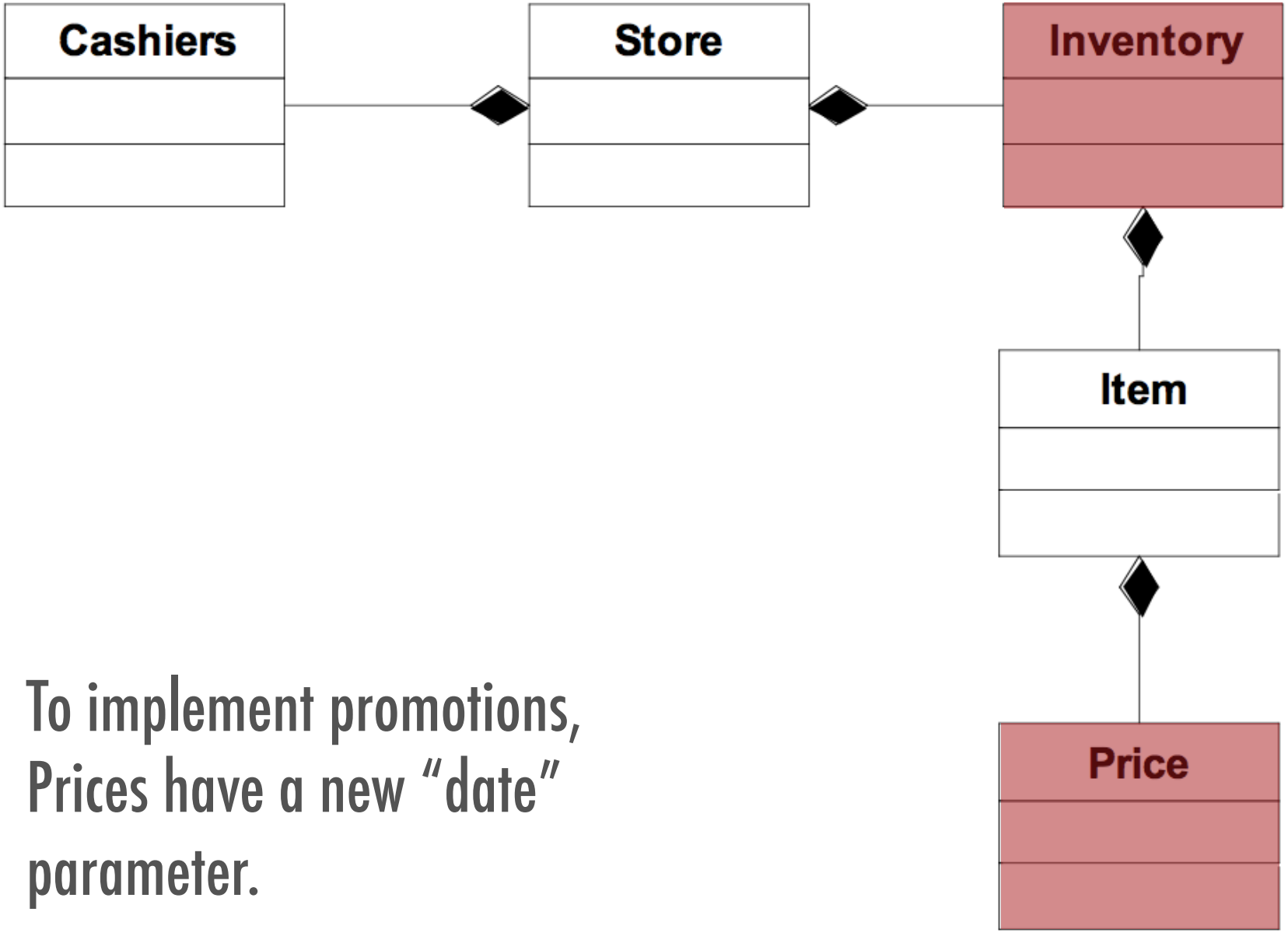
Classes composing functionality

```
class A {
        int getUserColor();
}

class B {
        void paintScreen(int color);
}

class C {
        A a;
        B b;

        void askThenPaint()
        {
            b.paintScreen(a.getUserColor());
        }
}
```

In cases of indirect interactions (coordinations), the two graphs differ



Neighborhood of a class



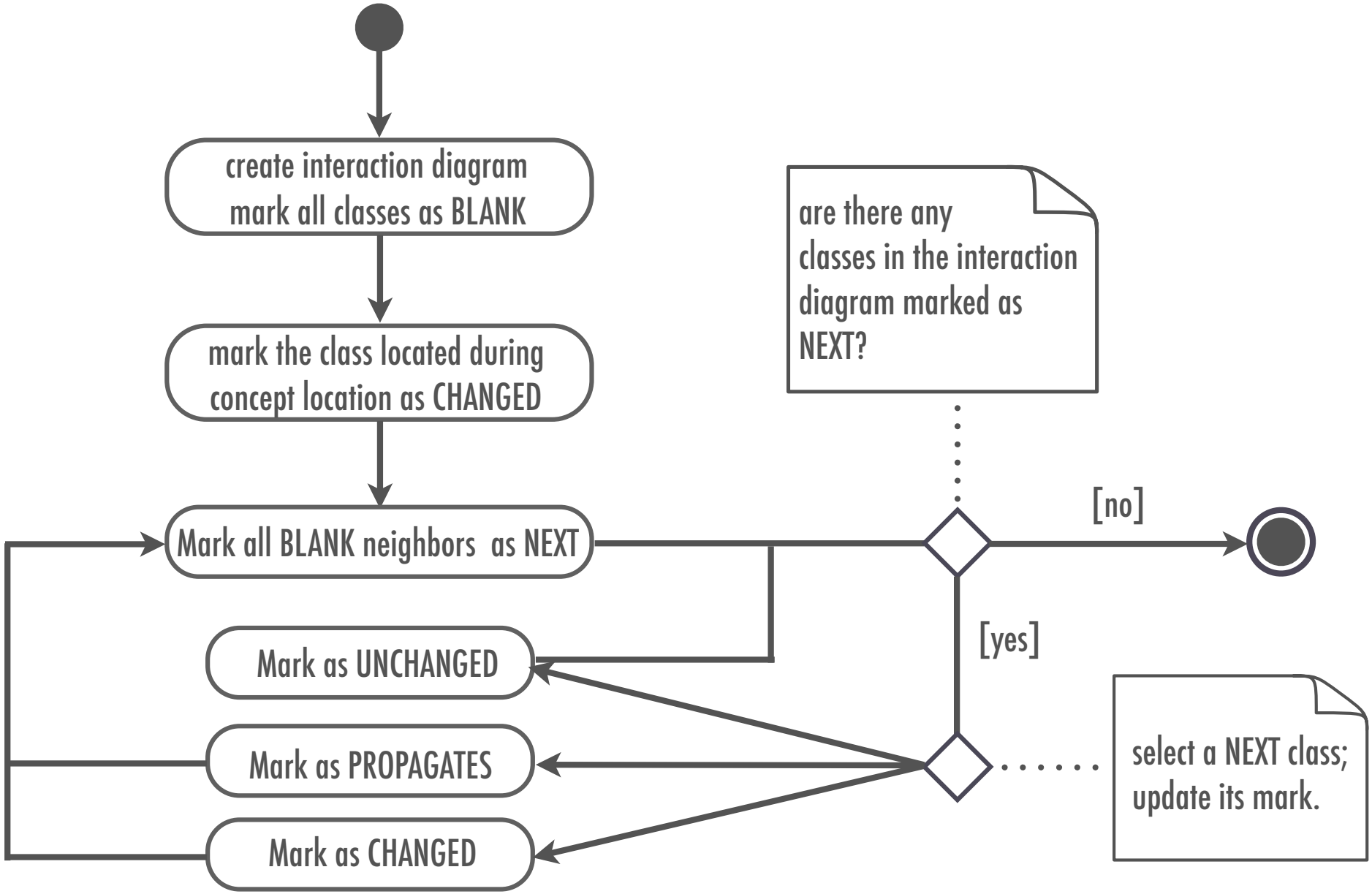
To implement promotions, Prices have a new "date" parameter.

The impact analysis process

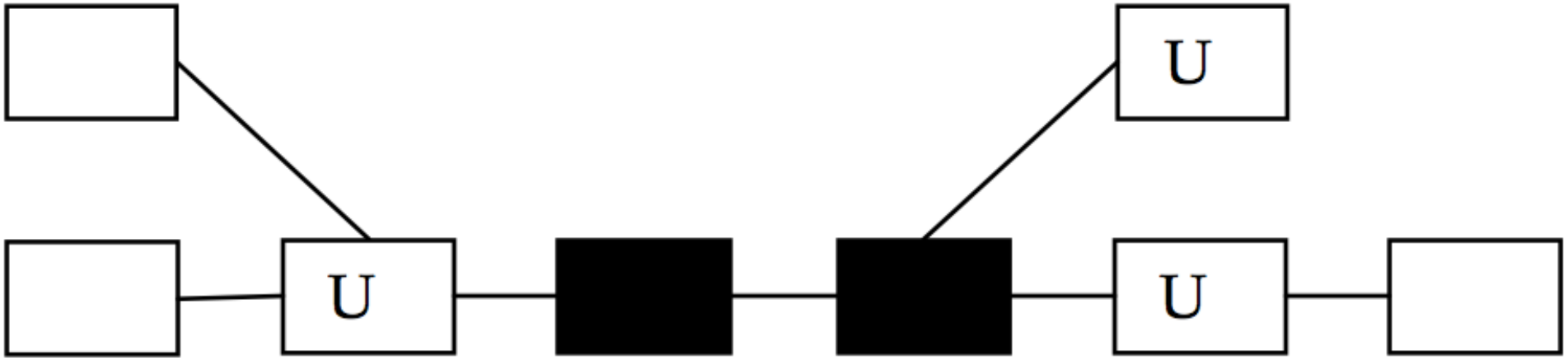
Marking classes

Mark	Meaning
BLANK	Not inspected yet: status unknown
CHANGED	Class belongs to the impact set. All BLANK neighbors become NEXT
UNCHANGED	Class does not belong to the impact set
NEXT	Scheduled for inspection
PROPAGATING	Does not belong to the impact set, but neighbors may still change. All BLANK neighbors become NEXT

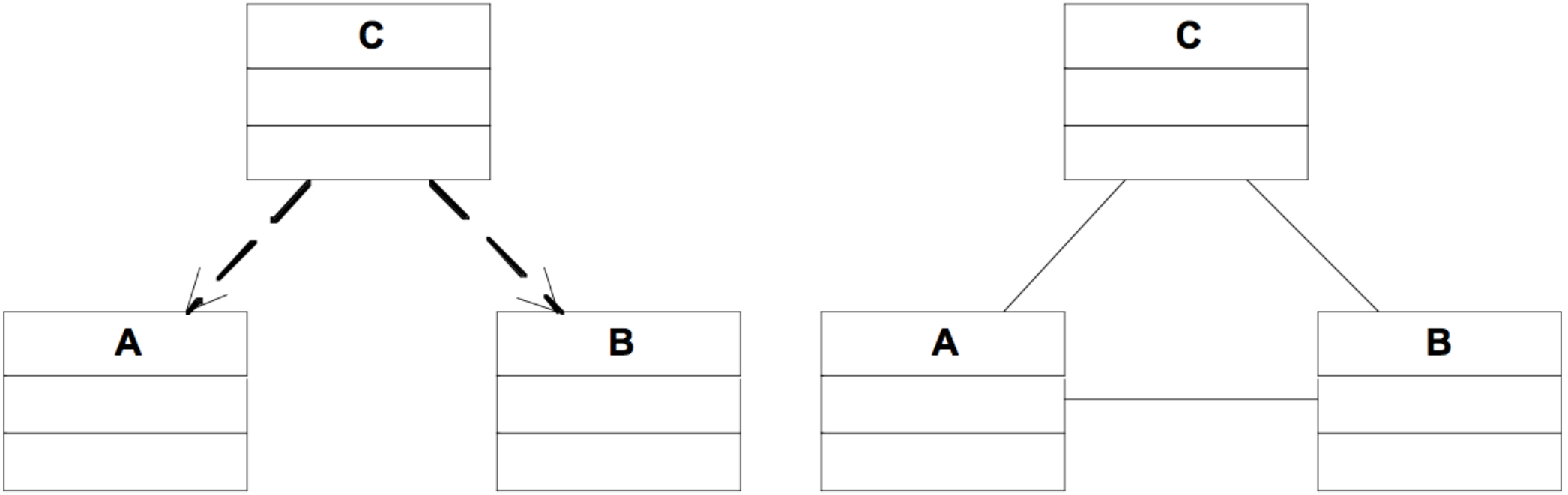
Impact analysis process



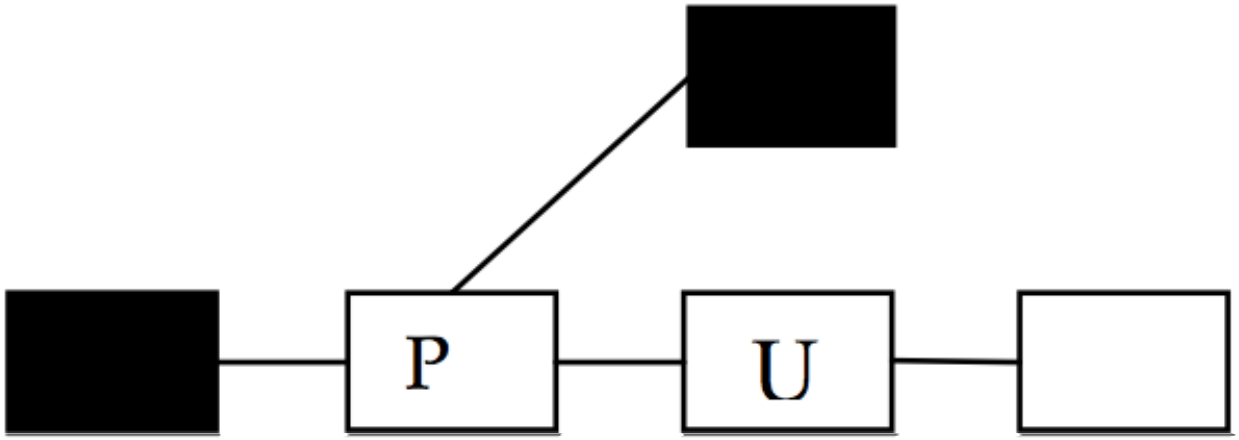
Impact analysis example



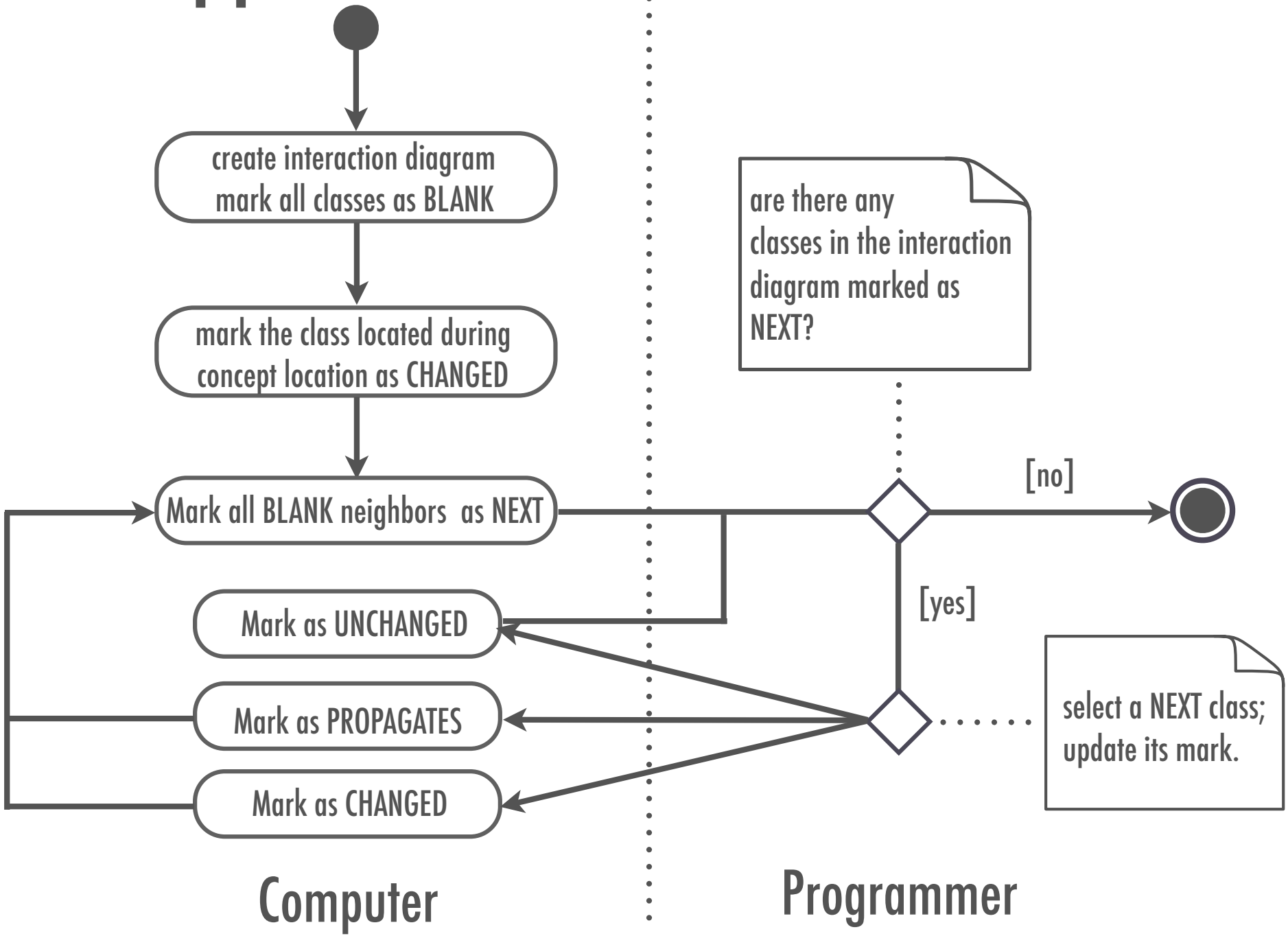
Classes propagating changes



Impact analysis example, with Propagating classes



Tool support for IA



Further tool support permit the ranking of the “NEXT” classes

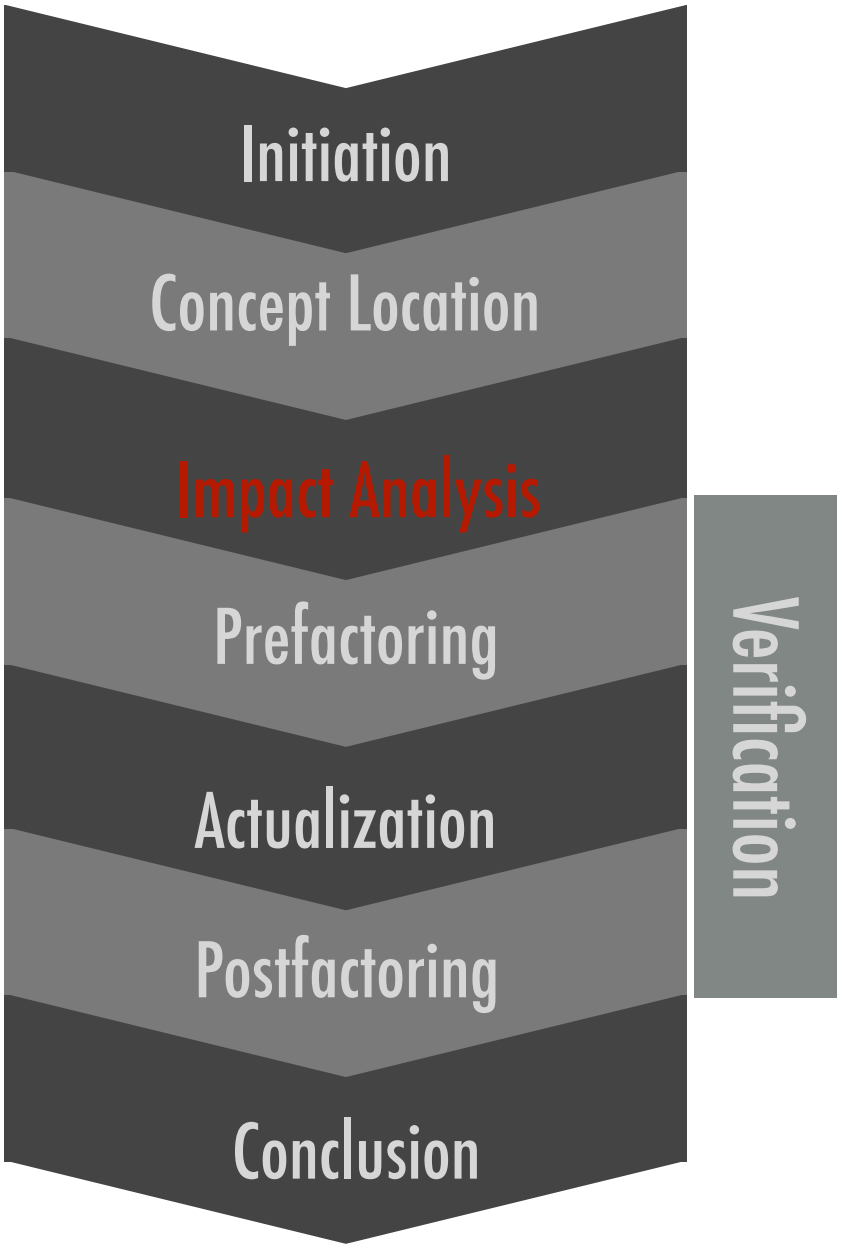
Based on number of references, textual similarity, etc ...

Finer-grained impact analysis is also possible
(methods, instance variables ...)

In Practice ...

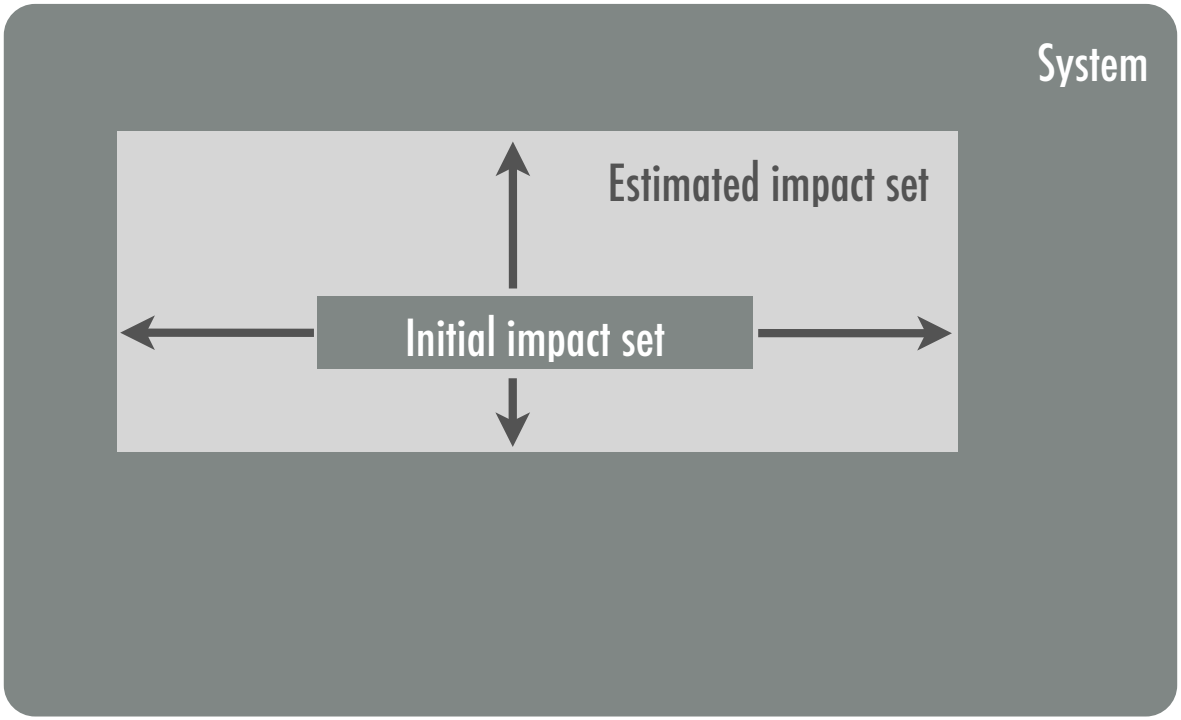
Conclusions

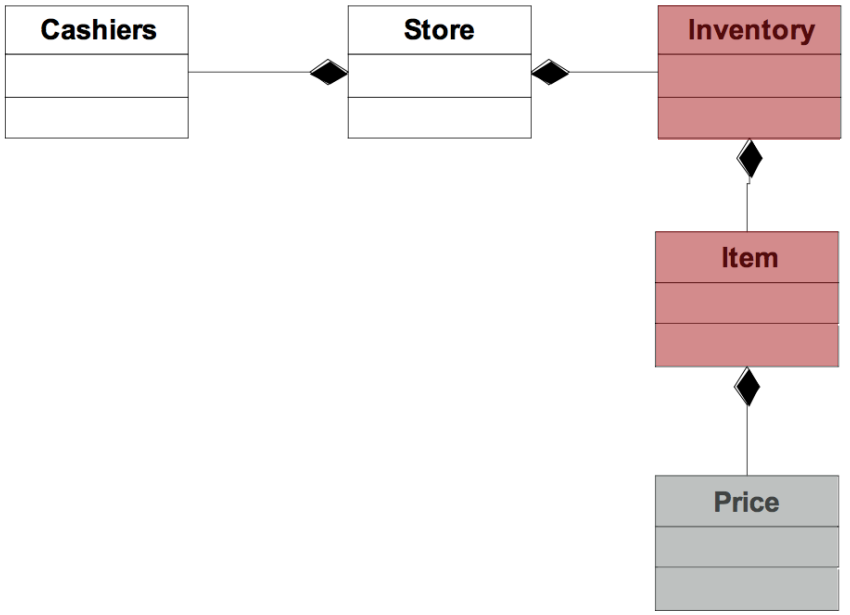
Impact analysis estimates actual changes



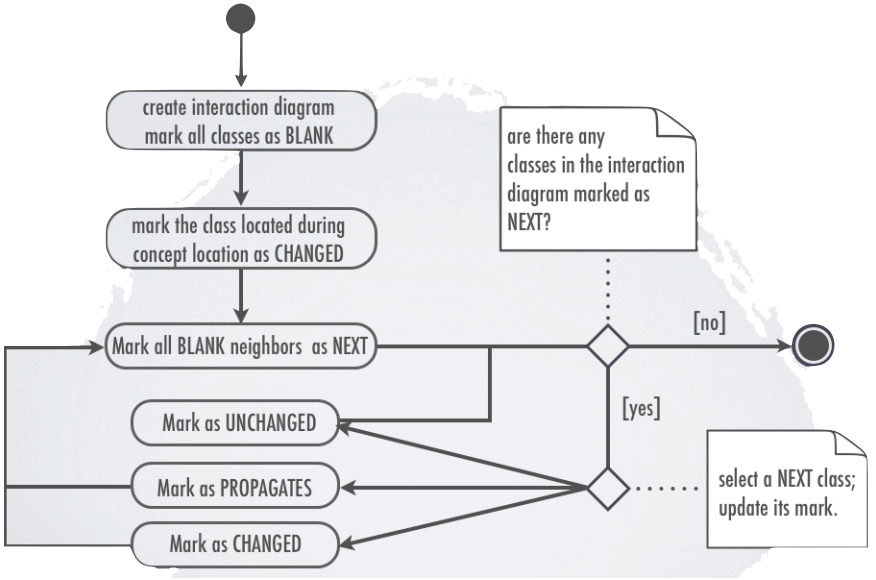
If the change is not localized, the programmer determines which other modules are affected by the change.

Impact analysis estimates the actual extent of the change





We can follow a process to perform impact analysis



Problems | Javadoc | Declaration | JRipples Hierarchical View | Search

Class	Mark	Matches	Full Name
BufferUpdate	Next		org.gjt.sp.jedit.msg.BufferUpdate
EditBus	Next		org.gjt.sp.jedit.EditBus
HyperSearchResults	Impacted	94	org.gjt.sp.jedit.search.HyperSearchResults
HyperSearchRequest		29	org.gjt.sp.jedit.search.HyperSearchRequest
SearchBar		26	org.gjt.sp.jedit.search.SearchBar
SearchDialog		25	org.gjt.sp.jedit.search.SearchDialog
HyperSearchOperationNode		21	org.gjt.sp.jedit.search.HyperSearchOperationNode
SearchAndReplace		16	org.gjt.sp.jedit.search.SearchAndReplace