Liskov substitution principle

Presentation by Eckhard Grodtke

Barabara Liskov
American computer scientist
made pioneering contributions to programming
languages and distributed computing. Her notable work includes the development of the Liskov substitution principle which describes the fundamental nature of data abstraction, and is used in type theory (see subtyping) and in object-oriented programming. Her work was recognized with the 2008 Turing Award, the highest distinction in computer science

Subtype Requirement: Let $\phi(x)$ be a property provable about objects x of type T. Then $\phi(y)$ should be true for objects y of type S where S is a subtype of T.

Liskov's notion of a behavioural subtype defines a notion of substitutability for objects

if S is a subtype of T, then objects of type T in a program may be replaced with objects of type S without altering any of the desirable properties of that program

Method Signature Requirements

Contravariance of method parameter types in the subtype.(minimum the same than Super)

Covariance of method return types in the subtype.(not more general than Super)

Behavioural Conditions

Preconditions cannot be strengthened in the subtype.

Postconditions cannot be weakened in the subtype.

Invariants must be preserved in the subtype.

Causes of Liskov violation

Feature reduction in Subtypes.(can be fixed by change of typehirarchie)

Overwriting Methods with extended functionality. (can be fixed by change of typehirarchie)

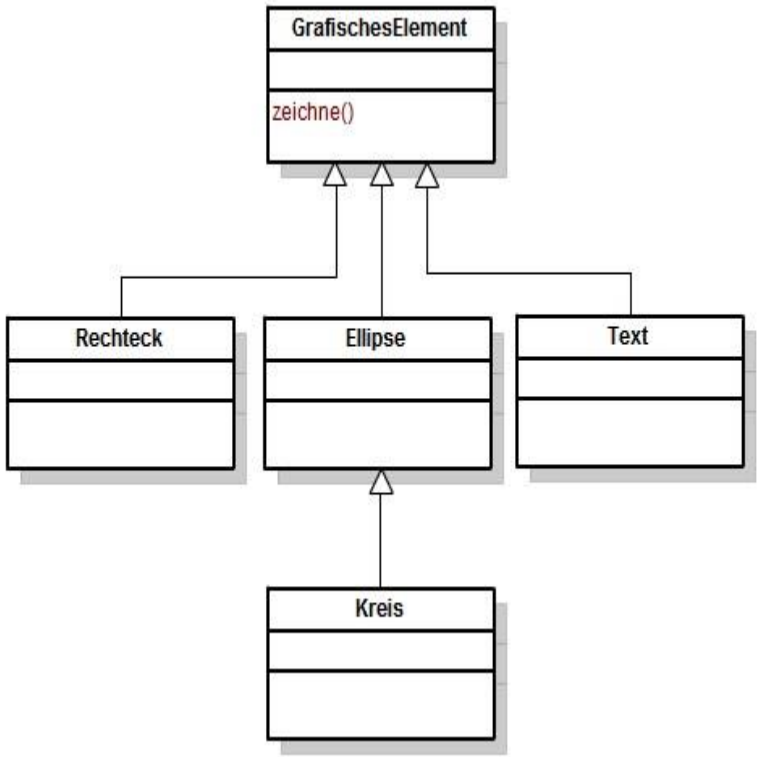
Multiple inheritance.(not possible in Java)

Circle–ellipse problem

Possible Solutions
changing the model

using a different language (or an existing or custom-written extension of some existing language)

using a different paradigm



Question

If we violate Liskov do we have to refacture the type hirarchie?

Question

Is overwriting “equals” and “hash” conform with Liskov?

Summary

Liskov Substitution Principle is a ingenious simple informal way to fix object oriented problems in design phase.

Its not a dogmatic ruleset that reduces richness of expression.

Thank you

Contact: eckhard@grodtke.com

Referenzen

*Liskov, B. (May 1988). "Keynote address — data abstraction and hierarchy". ACM SIGPLAN Notices. **23** (5): 17–34. doi:10.1145/62139.62141.* A keynote address in which Liskov first formulated the principle.
Meyer, B. (1988). Object-oriented Software Construction. Prentice Hall. ISBN 0-13-629031-0.
Wikipedia