

# Encapsulating and Exploiting Change with Changeboxes

Marcus Denker

Tudor Gîrba

Adrian Lienhard

Lukas Renggli

Oscar Nierstrasz

Pascal Zumkehr

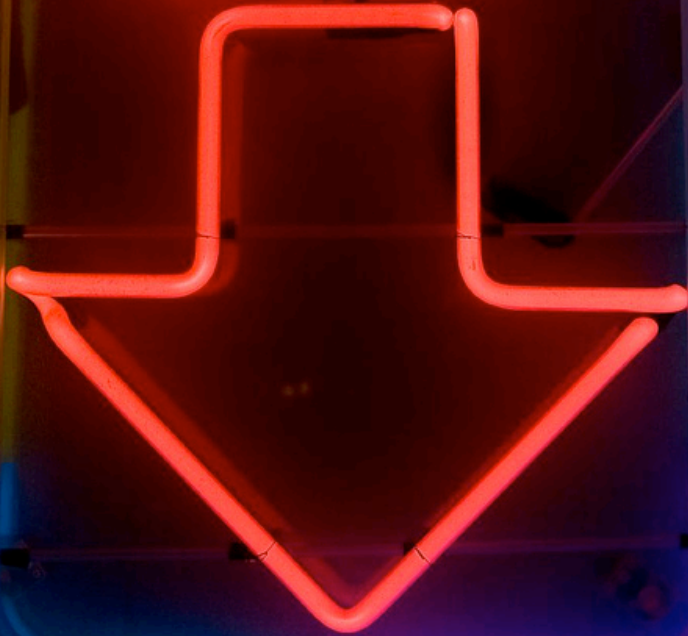
The logo of the University of Bern, featuring a stylized lowercase 'u' with a superscript 'b'.

---

*b*

**UNIVERSITÄT  
BERN**

CHANGE



Marcus Denker  
Tudor Gîrba  
Adrian Lienhard  
Lukas Renggli  
Oscar Nierstrasz  
Pascal Zumkehr



Change happens.



# Extreme Programming *Explained*

EMBRACE CHANGE

**KENT BECK**  
WITH **CYNTHIA ANDRES**  
Foreword by Erich Gamma

*Second Edition*



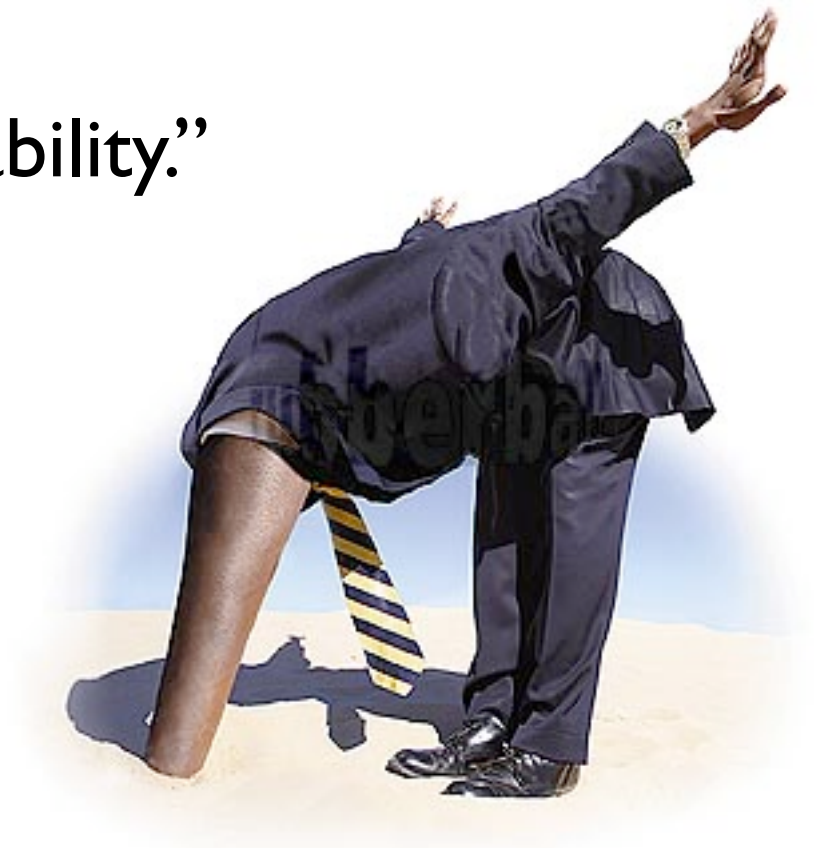
Extreme  
Programming  
*Explained*

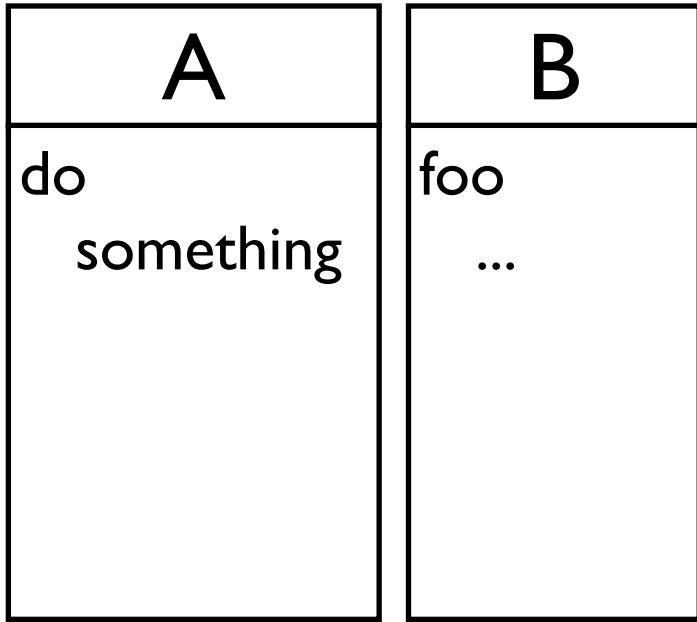
**EMBRACE CHANGE**

**KENT BECK**  
WITH **CYNTHIA ANDRES**  
Foreword by Erich Gamma

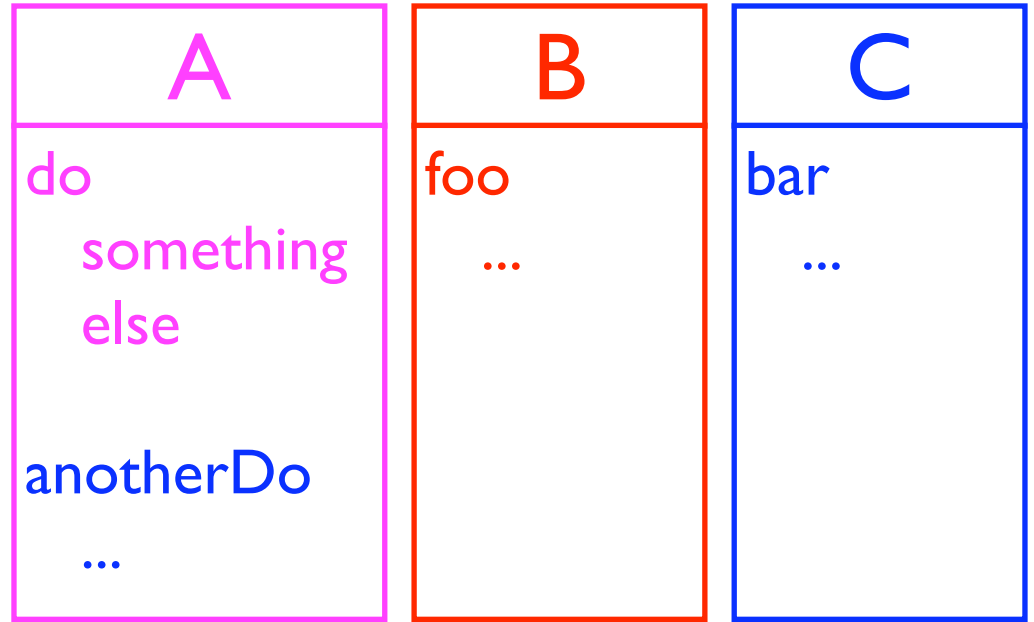
*Second Edition*

“Depend in the direction of stability.”





Version i



Change

Remove

Add

Version i+1



“Commit often.”



Update

Update

Change

Update

Change

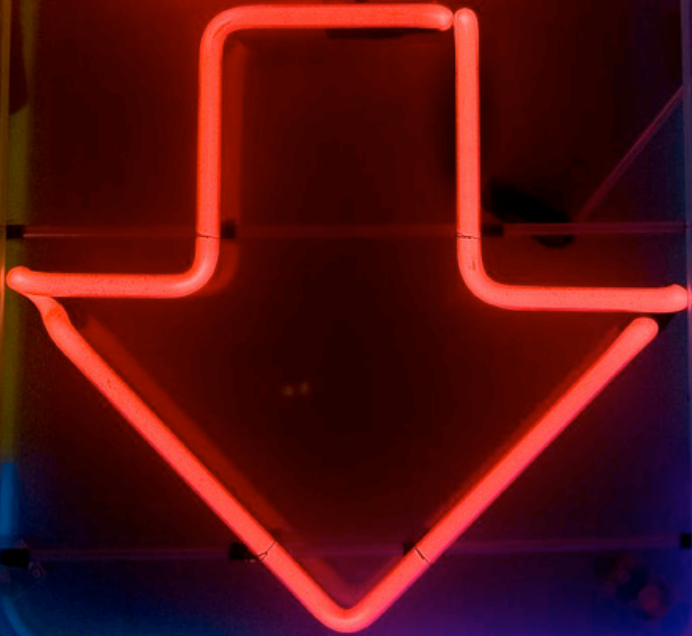
Commit

Update

?

Commit

CHANGE

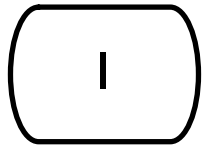




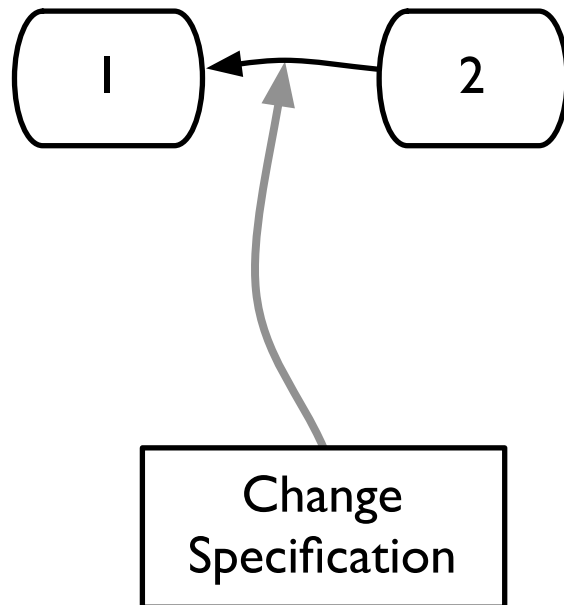
We put the change in a box



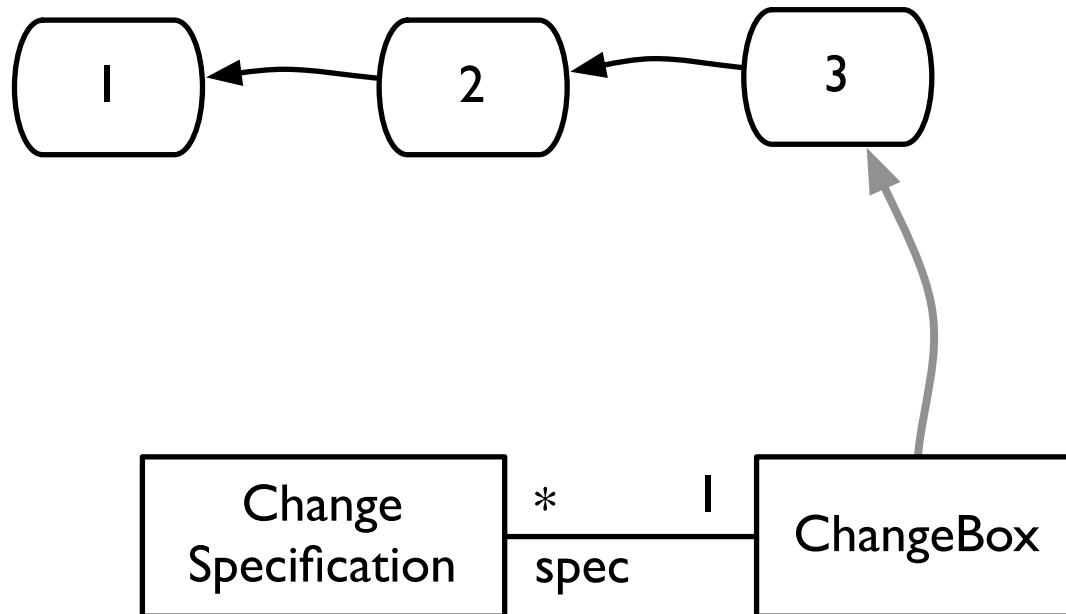


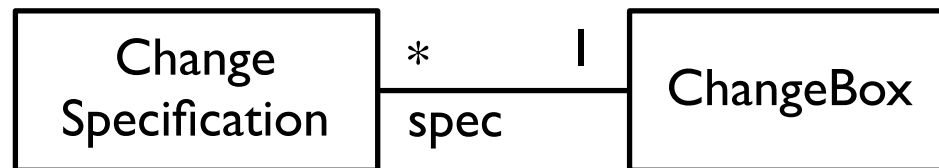
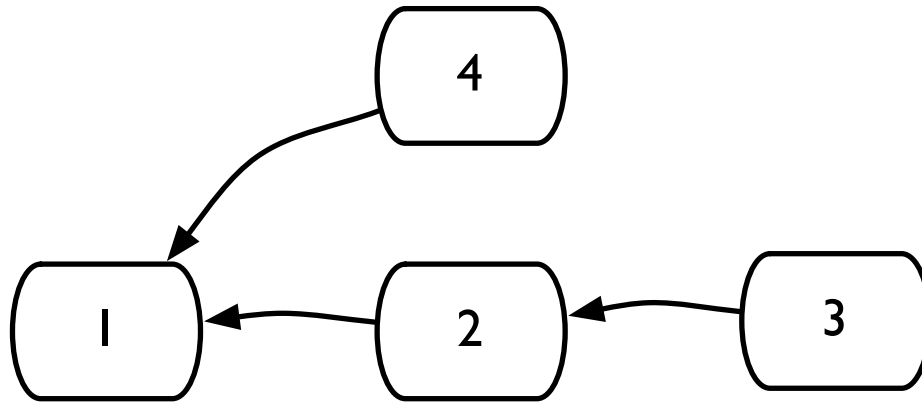


Changeboxes encapsulate changes.

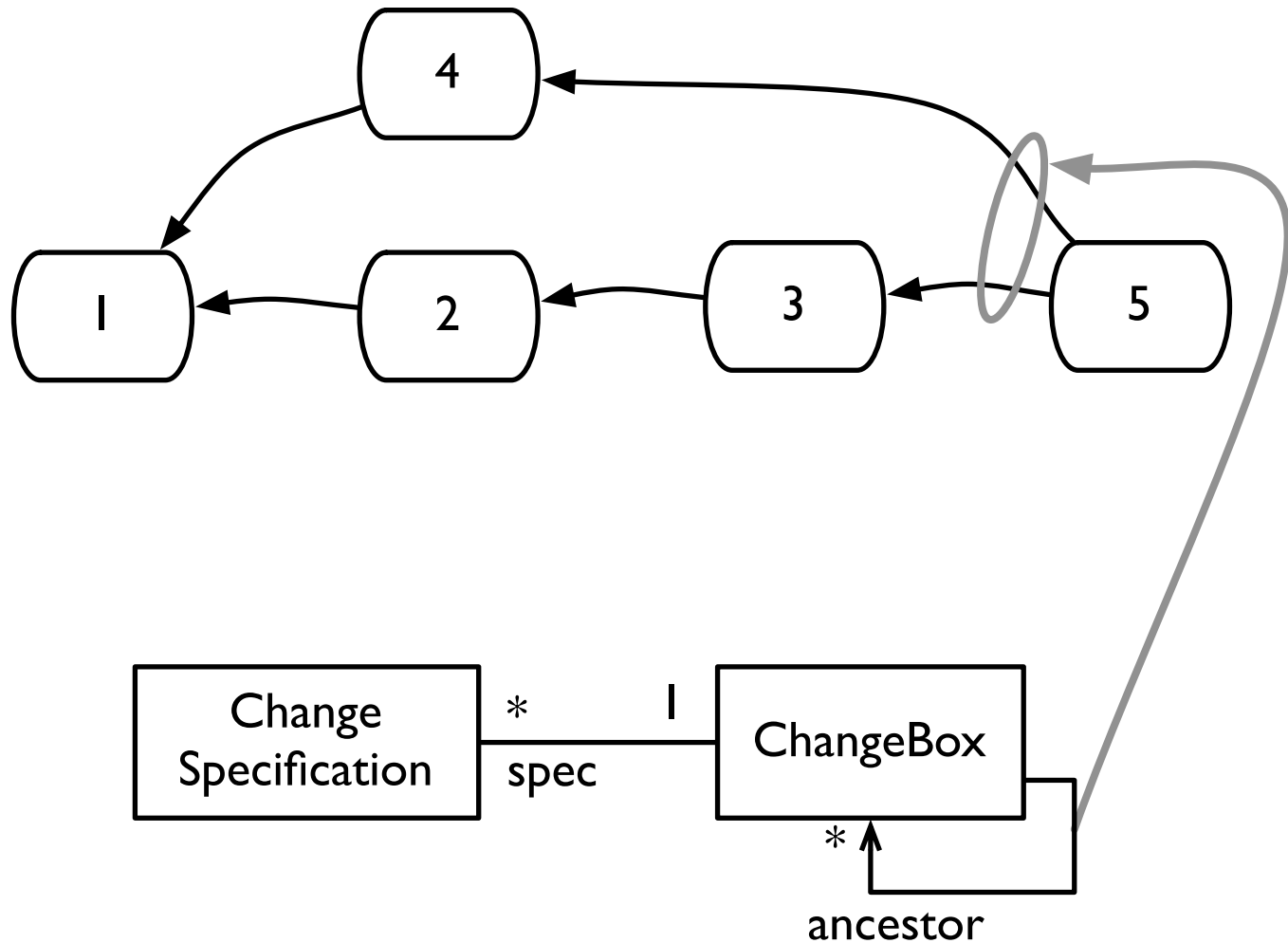


Changeboxes are immutable.

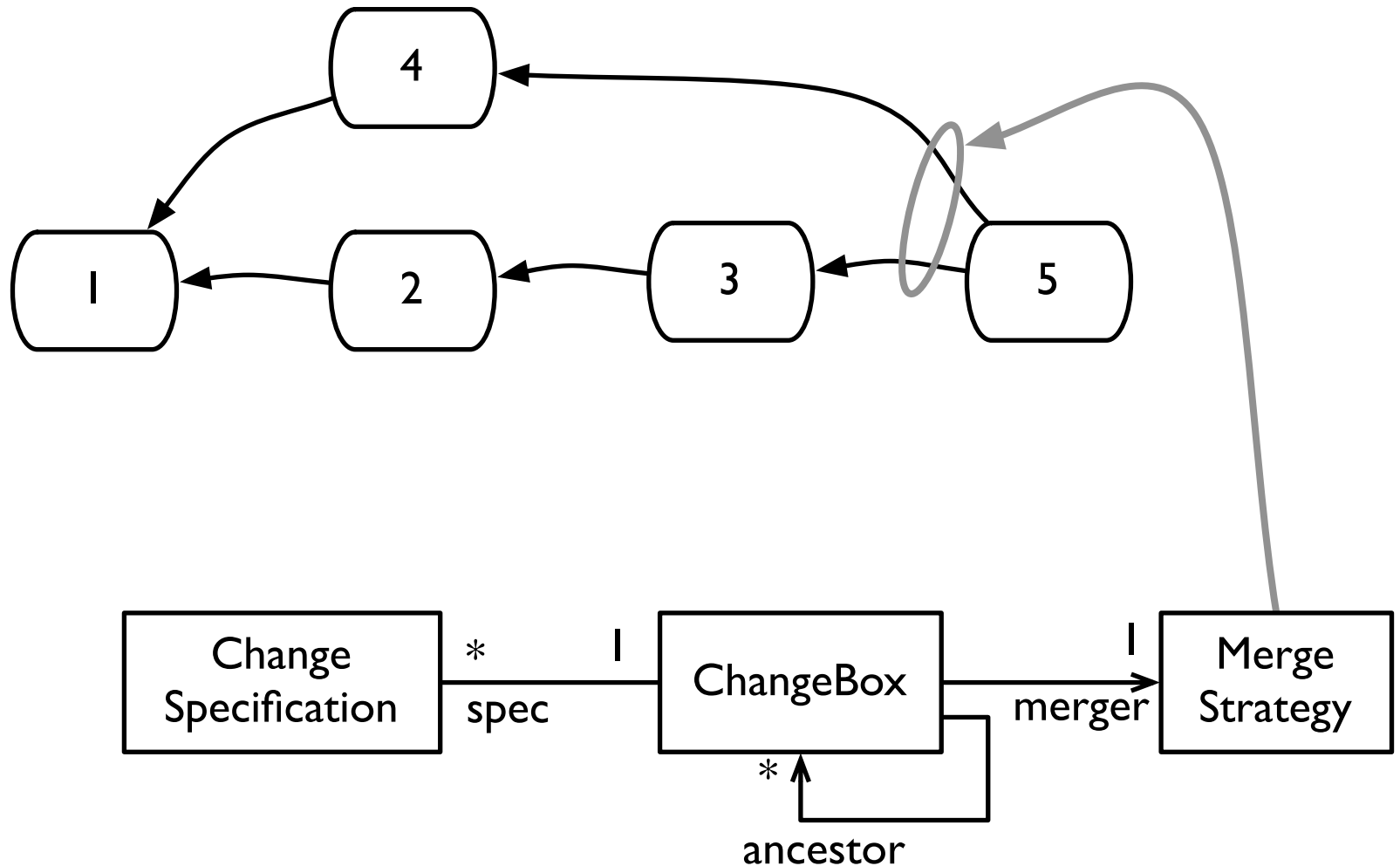




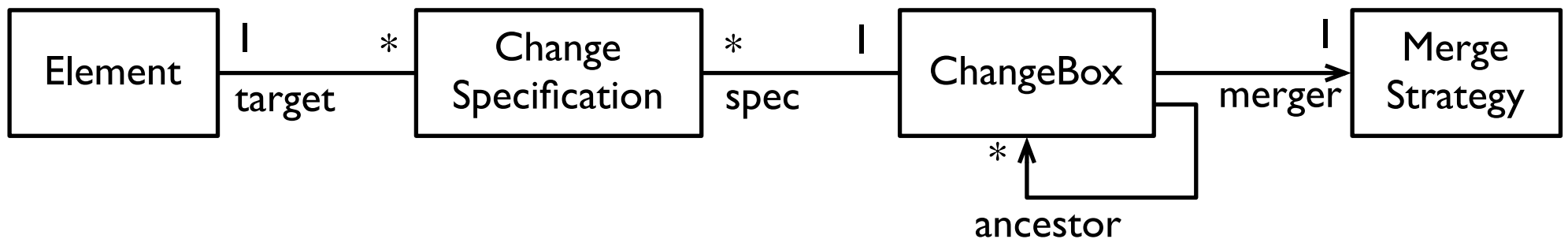
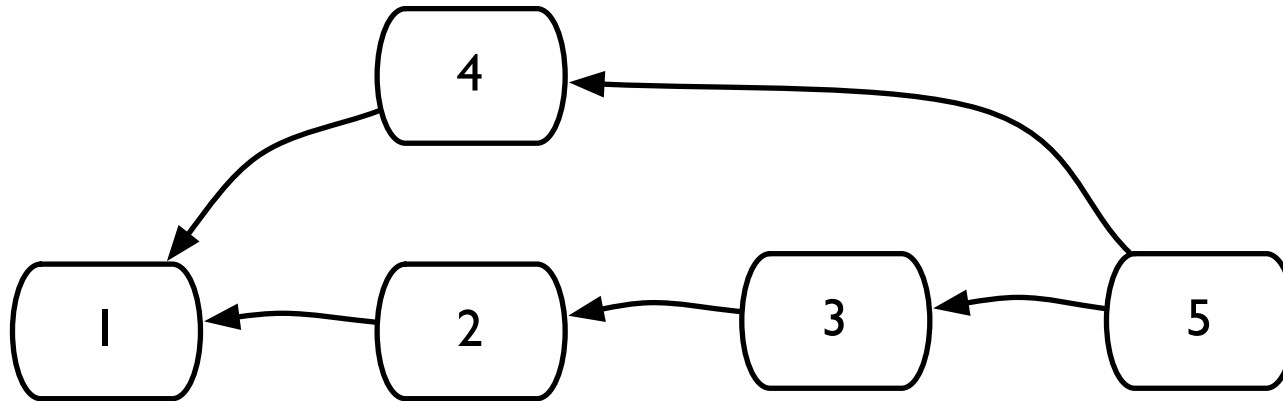
Changeboxes are applied on other Changeboxes.

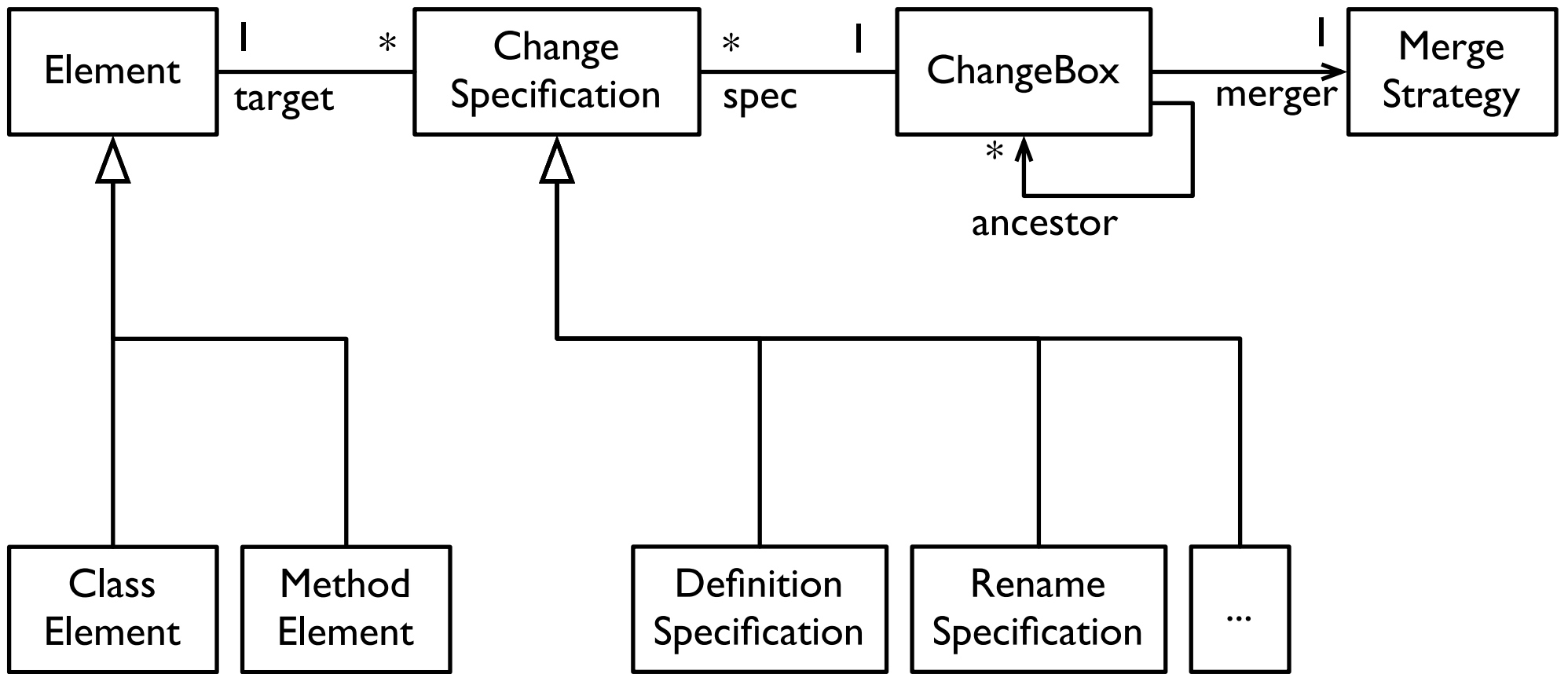


Changeboxes are applied on other Changeboxes.



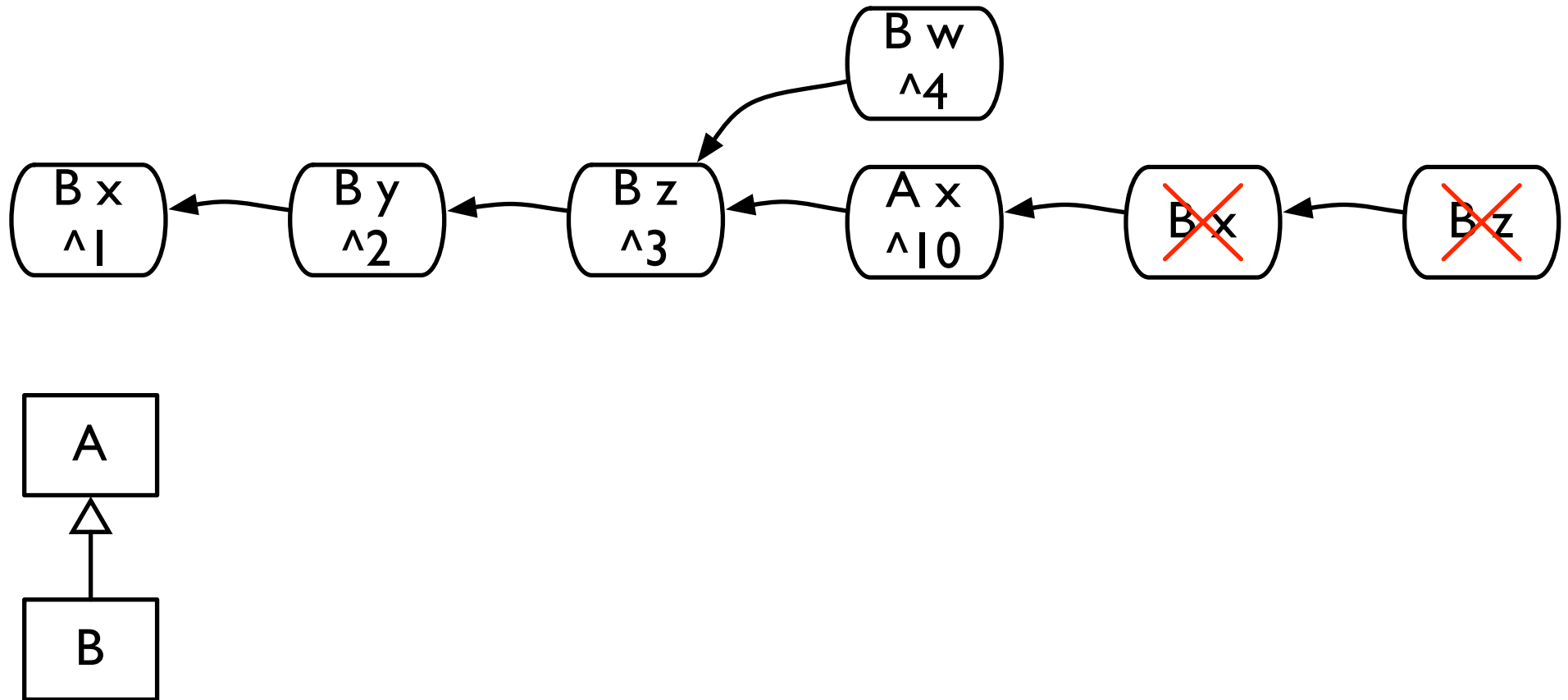
Changeboxes define the scope for dynamic execution.



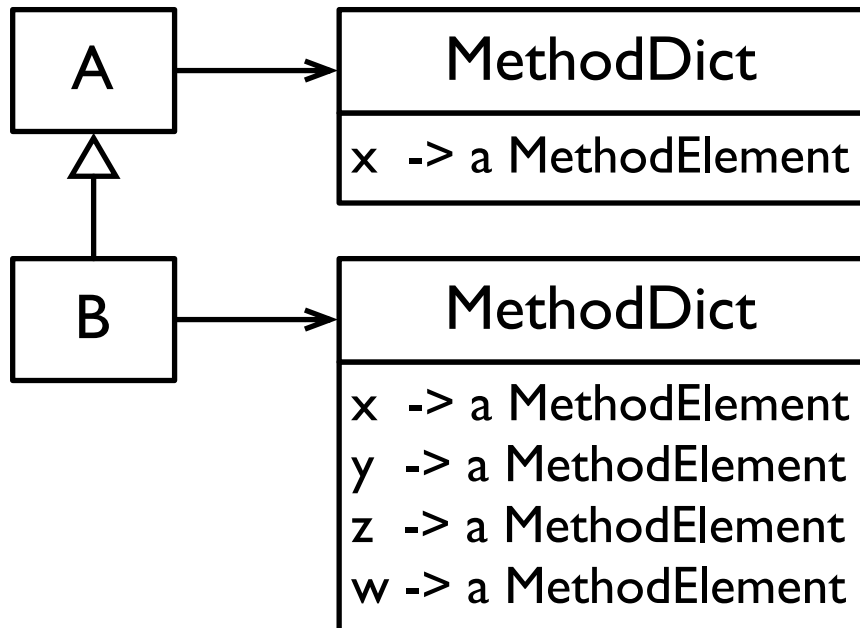
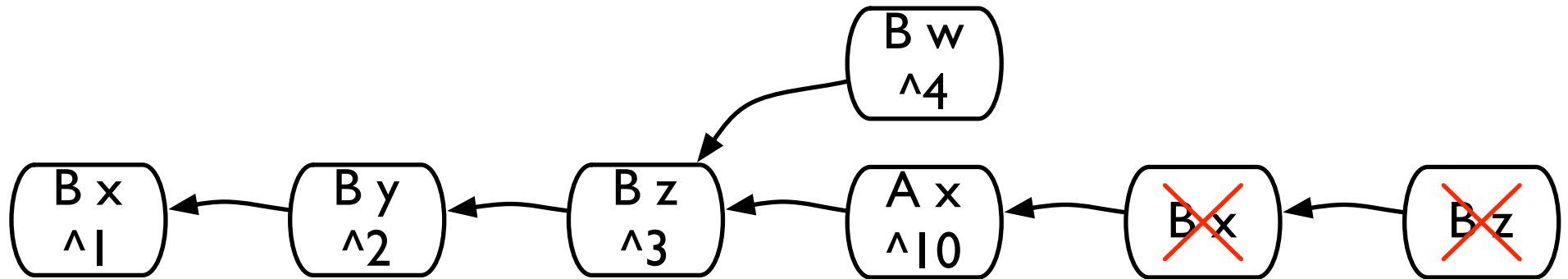




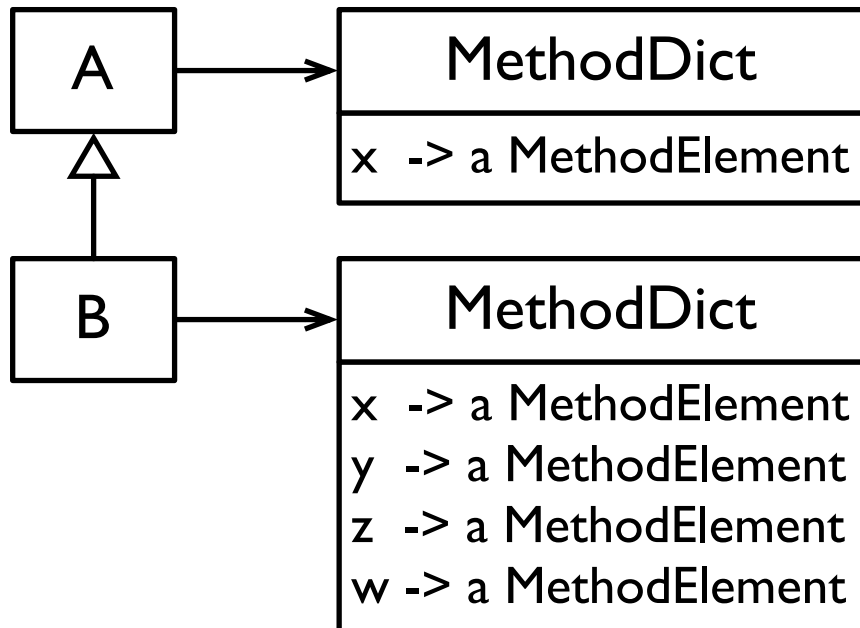
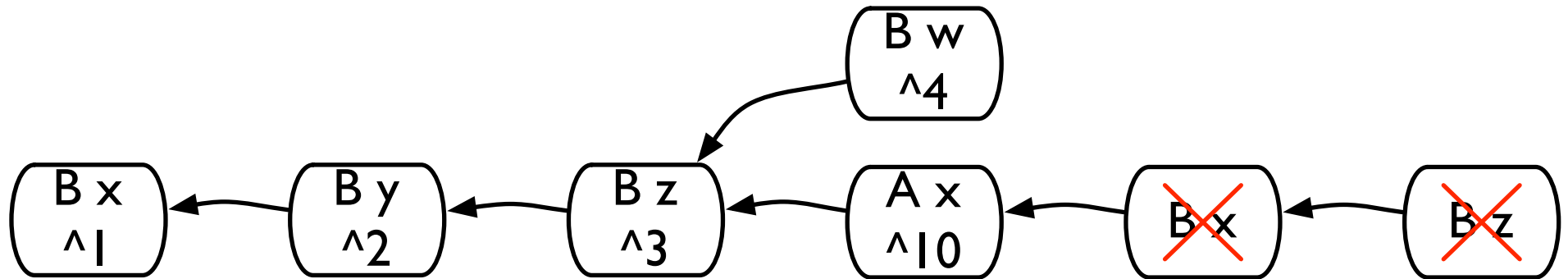
Method lookup is similar to Classboxes.



Method lookup is similar to Classboxes.



Method lookup is similar to Classboxes.



B new x ?

Classes are looked up in a similar way.

Merge is supported via a MergeStrategy.

Changeboxes encapsulate changes.

Changeboxes are immutable.

Changeboxes are applied on other Changeboxes.

Changeboxes define a scope for dynamic execution.

# Encapsulate and exploit change in Changeboxes

