

Java Exercise – Linked List

Ausgabe 08.2011

Copyright © 2011, thomas.iten@iten-engineering.ch
Alle Rechte vorbehalten.
Reproduktion (auch auszugsweise) ist nur mit schriftlicher Bewilligung des Verfassers gestattet.

Inhalt

| Kapitel | Inhalt | Seite |
|---------|------------------------------|-------|
| 1 | Übersicht | 3 |
| 2 | add (String data) | 7 |
| 3 | add (int index, String data) | 11 |
| 4 | remove (int index) | 16 |

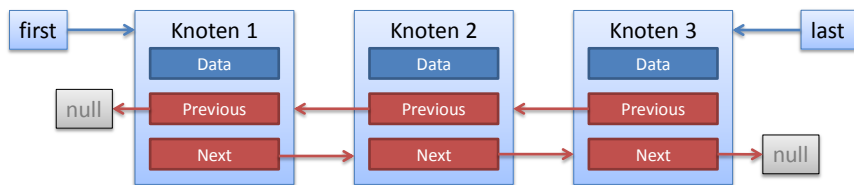
Kapitel 1

Übersicht

Linked List

- ▶ Eine **LinkedList** verwaltet Ihre Daten in einer **doppelt verketteten Liste**
- ▶ Beim Einfügen oder Löschen von Elementen sind damit nicht alle Elemente betroffen, sondern nur die **Knoten vor** und **nach** dem betroffenen Element
- ▶ Bei grossen Datenmengen mit häufigen Einfüge- und Löschooperationen sind Linked List performanter gegenüber Listen die Ihre Daten in Array Strukturen verwalten

Linked List



- ▶ Jeder Knoten enthält drei Referenzen
 - eine Referenz auf das Objekt mit den Daten
 - Eine Referenz auf den vorherigen Knoten
 - Eine Referenz auf den nächsten Knoten
- ▶ Die Knoten am Anfang und Ende enthalten jeweils nur eine gültige Referenz, die andere ist null.

Interface Methoden

- ▶ Hinzufügen von Daten ans Ende der Liste
- ▶ Einfügen von Daten an eine bestimmte Position in der Liste
- ▶ Lesen der Daten von einer spezifizierten Position
- ▶ Abfrage ob die Liste leer ist
- ▶ Löschen der Daten von einer spezifizierten Position
- ▶ Abfrage der aktuellen Listengröße

Kapitel 2

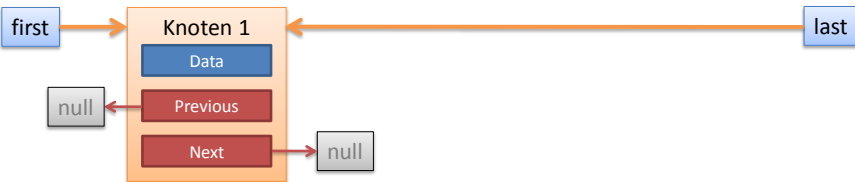
add (String data)

add the 1st element

► vorher




► nachher

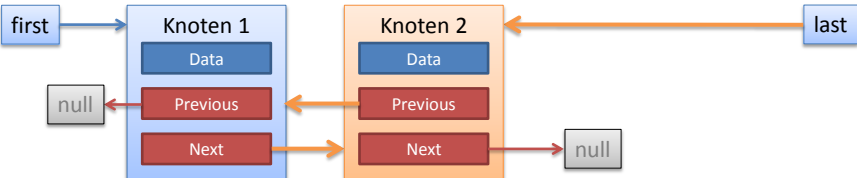


add the 2nd element

▶ vorher



▶ nachher



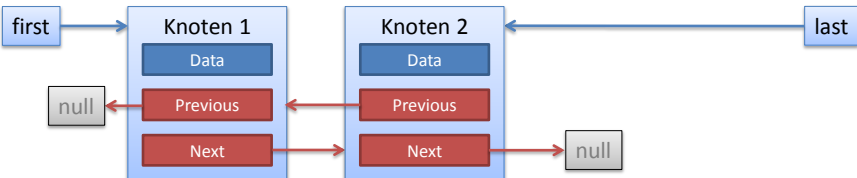
Copyright © iten-engineering.ch

Java Einführung

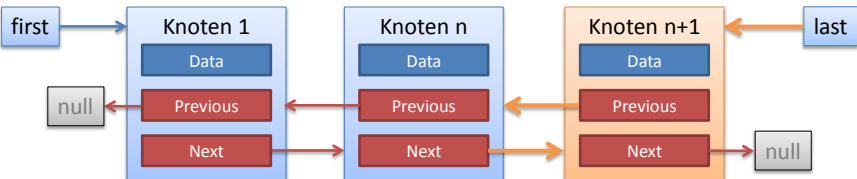
9

add the next element

▶ vorher



▶ nachher



Copyright © iten-engineering.ch

Java Einführung

10

5

Kapitel 3

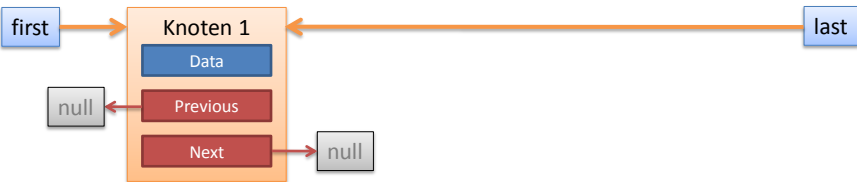
add (int index, String data)

add element at first postion to empty list

► vorher

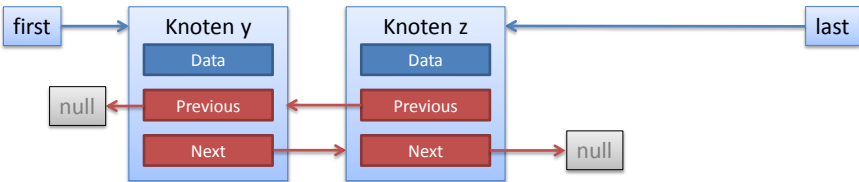


► nachher

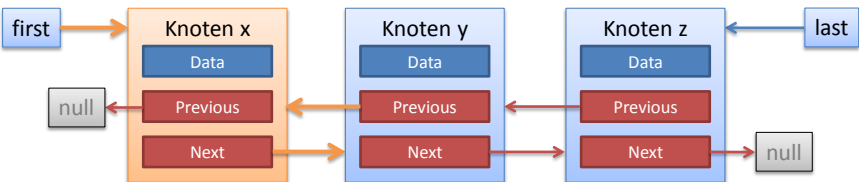


add element at first postion to non empty list

► vorher

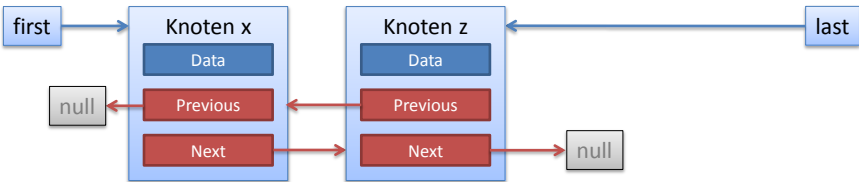


► nachher

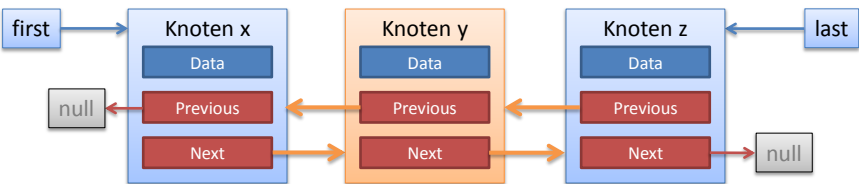


add element at arbitrary postion

► vorher

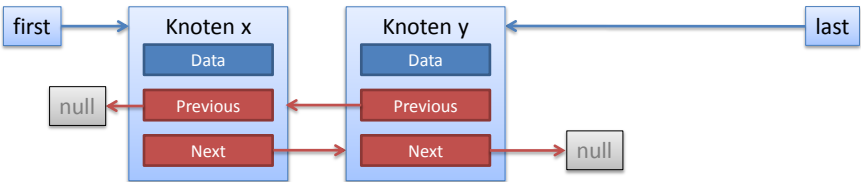


► nachher

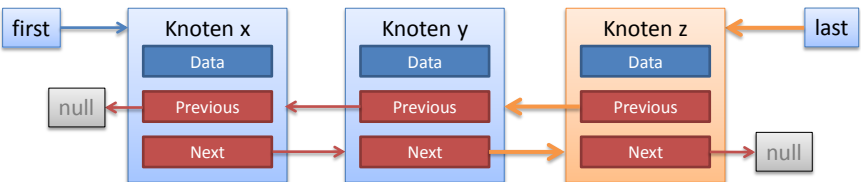


add element at last postion

► vorher



► nachher

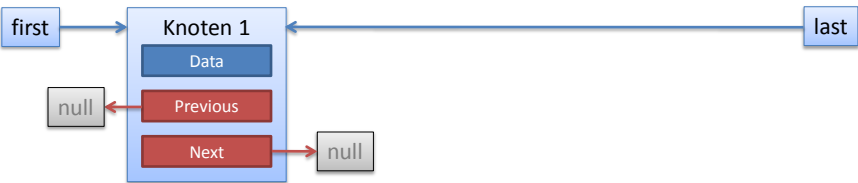


Kapitel 4

remove (int index)

remove first element of list with one entry

► vorher

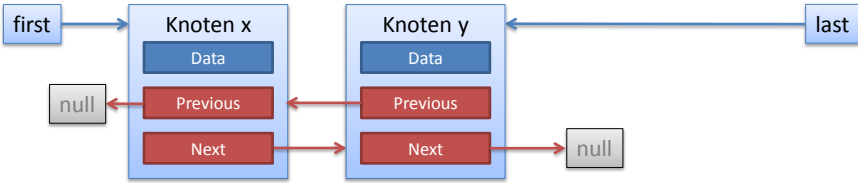


► nachher

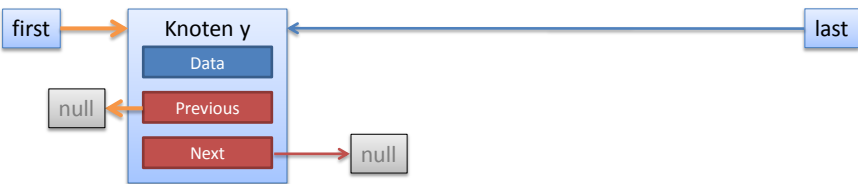


remove first element of list with two or more entries

► vorher

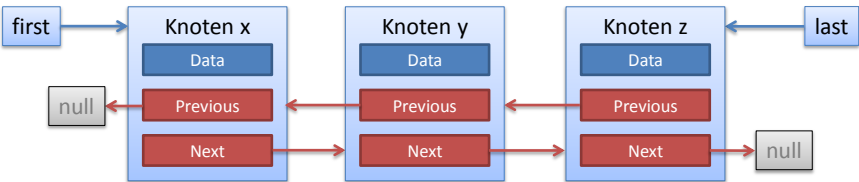


► nachher

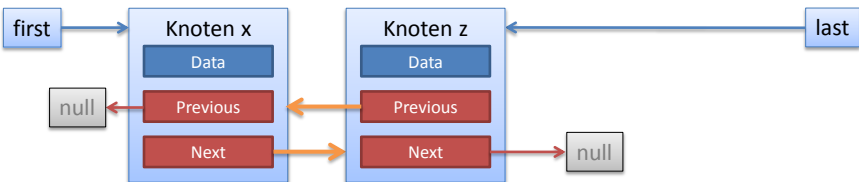


remove element at arbitrary position

► vorher

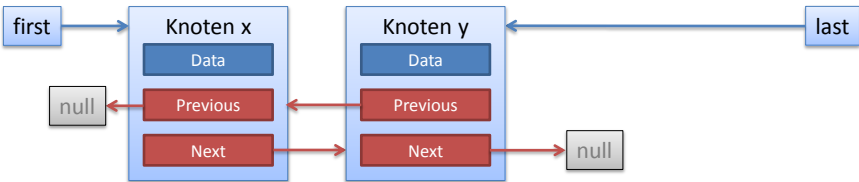


► nachher



remove last element

► vorher



► nachher

