

Flexible Codemodifikation im Elucidative Programming

Martin Heinzerling

TU Dresden, Fakultät Informatik

3. September 2009

Inhaltsverzeichnis

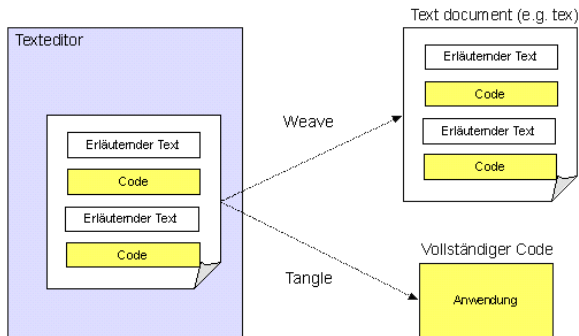
Einleitung

IST-Zustand

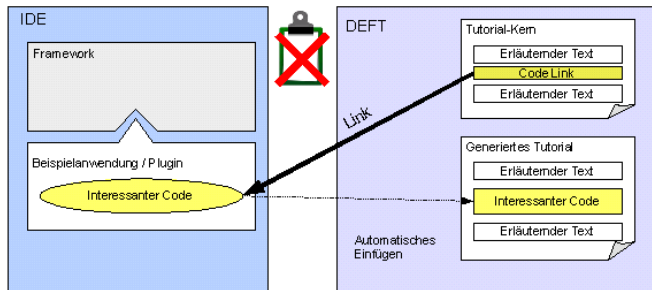
Erweiterungen

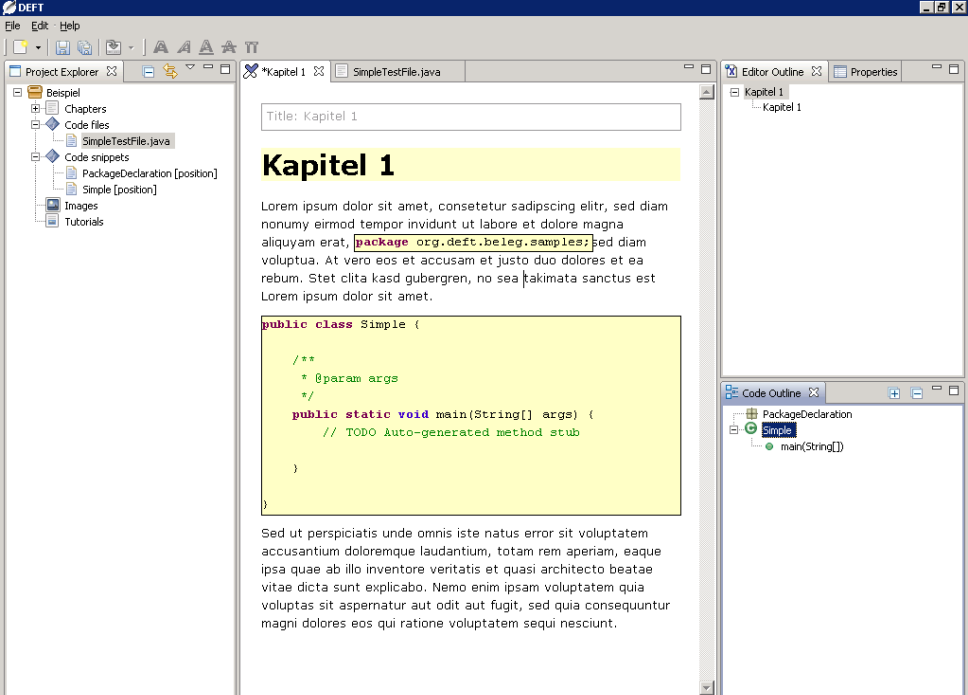
Zusammenfassung

Literate Programming



Elucidative Programming





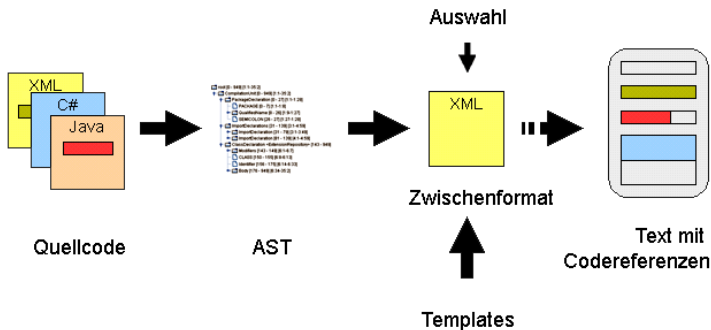
Kapitel 1

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, **package org.deft.beleg.samples;** sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

```
public class Simple {  
  
    /**  
  
     * @param args  
  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
  
    }  
}
```

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

```
package org.deft.beleg.samples;  
  
public class Simple {  
  
    /**  
  
     * @param args  
  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
  
    }  
}
```



Zeilennummerierung

```
3 public class Simple {  
4  
5     /**  
6      * @param args  
7      */  
8     public static void main(String[] args) {  
9         // TODO Auto-generated method stub  
10  
11     }  
12  
13 }
```


Ausblenden von Codeabschnitten

```
public class Simple { ... }
```

```
public class Simple {  
  
    public static void main(String[] args) {  
  
    }  
  
}
```

```
public class Simple {  
  
    public static void main(String[] )  
  
}
```


Hervorhebungen

```
public class Simple {  
  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
    }  
}
```

Anfügen und Ersetzen

```
public class Simple {  
  
    public static void main(String[] args) {  
        System.out.println("A");  
        System.out.println("B");  
        System.out.println("C");  
  
        notImportant();  
    }  
}
```



```
public class Simple {  
    //New comment  
    public static void main(String[] args) {  
        someOutput();  
  
        ...  
    }  
}
```

Schutz von invaliden Abschnitten

```
public class Simple { ... }
```

```
public class Simple {  
  
    public static void main(String[] )  
  
}
```

Entfernen von unerwünschten Leerzeichen

```
public class Simple {  
  
    public void main(String[] args)  
}
```

```
public class Simple {  
  
    public static void main(String[] )  
}
```

Zusammenziehen von Zeilen

```
public static void main(String[] args) {  
    ...  
  
    t=a;  
    a=b;  
    b=t;  
}
```

```
public static void main(String[] args) {  
    ...  
  
    t=a; a=b; b=t;  
}
```

Zusammenfassung

- ▶ Codeformatierung in beliebiger Auszeichnungssprache (z.B. CSS)
- ▶ Einfügen von Token/ Freitext/ Markern vor/ nach beliebigen Token bzw. Ersetzung von Token
- ▶ Markierung von invaliden Code für weitere Verarbeitung
- ▶ Automatisches entfernen unnötiger Leerzeichen
- ▶ Einfügen/ Entfernen von Leerzeichen/ Leerzeilen/ Zeilenumbrüchen
- ▶ Umwandeln von Tokenlisten in Kommentare

Vielen Dank!