



Techniken der Projektentwicklung

Dokumentation

Matthias Rolf

3. Februar 2006

Überblick

- 1 Verständliches Schreiben
 - Wie drücke ich mich einfach aber präzise aus?
- 2 Technische Dokumentation
 - Schreiben und Gestalten von Benutzerhandbüchern
- 3 Code-Dokumentation
 - Java-Code dokumentieren mit JavaDoc



Verständliches Schreiben



Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...

Quelle: Wikipedia

Unverständliche Texte

- (leider) sehr häufig in Wissenschaft und Technik
- Gründe vielseitig:
 - falsches Verständnis von Wissenschaftlichkeit
 - Unvermögen (gutes Schreiben wird kaum gelehrt)
 - fehlendes inhaltliches Verständnis (unpräzise Ausdrucksweise)
 - Selbstdarstellung durch Fachwörter und komplizierte Satzstrukturen
- Fragen:
 - Was macht Texte unverständlich?
 - Wie machen wir es besser?

Hamburger Verständlichkeitskonzeption

- Verständlichkeit hat vier Dimensionen:
 - 1 Einfachheit
 - 2 Gliederung und Ordnung
 - 3 Kürze und Prägnanz
 - 4 Zusätzliche Stimulanz
- Einfachheit und Gliederung am wichtigsten

Einfachheit

- komplizierte Texte immer schwerer zu verstehen
- verschiedene Betrachtungsebenen:
 - 1 Inhaltliche Ebene:
komplizierte Sachverhalte schwerer als simple (meist vorgegeben)
 - 2 Satzbau:
lange Sätze und Schachtelsätze schwer
 - 3 Wortwahl:
ungebräuchliche (Fach-)Wörter schwerer als gebräuchliche

Gliederung und Ordnung

- Gliederung wird immer zuerst entworfen
- Zusammengehörige Informationen zusammen darlegen
- Auf einander aufbauende Informationen in passender Reihenfolge
- Überschriften wichtig
- Einleitungen und Zusammenfassungen erleichtern Verständnis

Kürze und Prägnanz

Prägnanz die; - : Schärfe, Genauigkeit, Knappheit des Ausdrucks.

- lange Texte prinzipiell schwieriger zu verstehen
- prägnant formulierte Aussagen besser als diffuse Gedankengänge
- Problem: kann im Widerspruch zu Einfachheit stehen
 - prägnante Aussagen erfordern vielleicht Fachwörter
 - evtl. zu hohe Informationsdichte
- Gewisse Redundanz zum Verständnis nötig

→ Tatsachen auf den Punkt bringen

Zusätzliche Stimulanz

- Stimulanz/Anregung durch:
 - passende Beispiele
 - Analogien
 - Metaphern
- Verständnis und Interesse können gesteigert werden
- motivierende Textpassagen
- Problem: kann im Widerspruch zu Kürze stehen

→ richtiges Maß finden

Hamburger Verständlichkeitskonzeption: Fazit

Vorteile

- präzisiert den Begriff „Verständlichkeit“
- erlaubt Beurteilung von Texten
- hilft beim Eingrenzen von Problemstellen

Nachteile

- Verständlichkeit als reines Textmerkmal
→ Vorwissen des Lesers auch wichtig
- liefert keine konkreten Verbesserungsvorschläge

Füllwörter

- überflüssige Wörter weglassen
- Beispiele:
 - *Rein* rechnerisch soll dieses Investitionsprogramm ...
 - Das Gleiche gilt *auch* für ...
 - Informationen, die *in Gestalt von* Dokumenten vorliegen
 - *aufgeführte* Gründe
- häufige Füllwörter: auch, aber, rein, nun, jetzt, an sich, außerdem, bloß, halt, quasi
- Text wird einfacher und präziser
- gelungener Text
→ nichts kann mehr weggelassen werden

Pronomen

Wer seinen Hund liebt, muss nicht auch seine Flöhe lieben.

- Pronomen (seine, dessen, dieser, jene, . . .) ersetzen Substantive
 - führen leicht zu Rätseln
 - Beispiel:
 - ① Am Anfang war das Wort, und das Wort war bei Gott, und Gott war das Wort. (Johannes-Evangelium)
 - ② Am Anfang war das Wort. Es befand sich bei Gott, und letzterer war identisch mit ersterem.
- Substantive/Namen ruhig wiederholen — bei Verben variieren

Fremdwörter

- Fremdwörter müssen treffend sein
- kein Ersatz für präzise Aussagen
- Beispiel:

alternative Basisaktivitäten

- beide Wörter unpräzise
- gibt es traditionelle Basisaktivitäten?
- „Aktivitäten“ selten treffend

→ entweder passendes Fremdwort, oder gar keins

Nominalisierungen

- Bildung von Substantiven aus anderen Wortarten
- schwerer zu verstehen wegen höherer Informationsdichte
- Beispiel:
 - 1 Die *Eliminierung* von Nominalisierungen in Texten von geringer *Verständlichkeit* führt zu einer *Erhöhung* derselben.
 - 2 Nominalisierungen zu ersetzen macht Texte verständlicher.

→ Nominalisierungen durch passende Verben/Adjektive ersetzen

Satzbau

- Satzbau kann Verständlichkeit fördern:
 - ① Hauptaussagen in den Hauptsatz
 - ② Begründungen u.ä. in Nebensätze
 - ③ Nebensätze an Hauptsatz anschließen
- Beispiel:
 - *Die Voraussetzung* dafür, dass man ein Array von sortierbaren Werten mit HeapSort sortieren kann, *ist*, dass dieses einen binären Heap repräsentiert.
 - *Man kann ein Array von sortierbaren Werten mit HeapSort sortieren*, wenn es einen binären Heap repräsentiert.
- Bei langen Sätzen / vielen Nebensätzen schwierig

Einen Punkt setzen

- lange Schachtelsätze aufspalten:
 - 1 Welche (Teil-)Aussagen gibt es?
 - 2 Für jede Aussage einen Satz
- Beispiel:
 - Man kann ein Array von sortierbaren Werten mit HeapSort sortieren, wenn es einen binären Heap repräsentiert.
 - Ein Array von sortierbaren Werten kann man mit HeapSort sortieren. Dazu muss das Array einen binären Heap repräsentieren.

Zusammengesetzte Verben

- zusammengesetzte Verben laden zu Schachtelsätzen ein
- Beispiel:
 - Ich *schlage*, sofern ihr nichts dagegen habt, eine kurze Pause *vor*.
- solche Verben also . . .
 - zusammen lassen, oder
 - ganz ersetzen
- Beispiele für Ersetzungen:
 - informieren statt mitteilen
 - beteiligen statt teilnehmen
 - können statt möglich sein

Funktionsverben

- Verben, die nicht ohne Substantiv auskommen
- Beispiel:
 - Erst stirbt der Wald, dann stirbt der Mensch.
 - Erst *kommt* der Wald *zu Tode*, dann *scheidet* der Mensch *aus dem Leben*.
- solche Verben vermeiden:
 - ermöglichen statt Gelegenheit bieten
 - können statt in der Lage sein
 - bezweifeln statt Zweifel hegen
 - prüfen statt einer Prüfung unterziehen

Aktiv statt Passiv

- bei Verben Aktiv statt Passiv verwenden
- der Passiv . . .
 - läd zu Schachtelsätzen ein
 - macht Texte unpräzise (Handelnder unklar)
- Passiv nur wenn Handelnder tatsächlich unbekannt oder irrelevant



Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...

Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...

„Formal“ reicht aus, evtl. ganz streichen



*Rein formal wird beim EM-Algorithmus **angenommen**, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Aussage im Hauptsatz nicht zentral

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der **beobachteten** stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Von was sonst?



*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe **auf folgende Art und Weise** zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Füllphrase



*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der **eingehenden** Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Gibt es hier ausgehende?



*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme **de-ren** Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Besser: „ihren“

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als **Endergebnis**. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Missverständlich

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. **Das bedeutet**, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Hauptsatz!



*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. **Bei der Approximation** der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Besser: „Beim Approximieren“ / „Beim Annähern“

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte **durch den EM-Algorithmus** ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Durch was sonst?

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist **dies** normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Was?

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies **normalerweise** aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Unpräzise

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise **aber** nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Füllwort



*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus **ist dies normalerweise aber nicht mehr der Fall**. Die Wahrscheinlichkeitsdichte eines Zielwertes lässt sich bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen darstellen als: ...*

Hauptsatz!

*Rein formal wird beim EM-Algorithmus angenommen, dass die Werte der beobachteten stochastischen Größe auf folgende Art und Weise zustandekommen: Wähle zuerst eine der eingehenden Zufallsvariablen aus und übernehme deren Wert als Endergebnis. Das bedeutet, dass genau ein Gewicht den Wert eins annimmt und alle anderen null sind. Bei der Approximation der Gewichte durch den EM-Algorithmus ist dies normalerweise aber nicht mehr der Fall. Die Wahrscheinlichkeitsdichte eines Zielwertes **lässt sich** bei Normalverteilungsannahme und konstanter Varianz der einzelnen Zufallsvariablen **darstellen als**: ...*

Prädikat nicht zusammen



Technische Dokumentation



Was bisher war:

- Dokumentieren von Anforderungen (Use Cases)
- Dokumentieren von Softwaremodellen mit UML-Diagrammen
- Nutzbarkeit von Software (Usability)

Heute:

- Schreiben von Benutzerhandbüchern (jetzt)
- Dokumentieren von Quellcode (nächster Abschnitt)

Benutzerhandbücher

- Anleitung zum Benutzen einer Software
- auch bei guter Usability sehr wichtig
- großer Teil der Funktionalität oft unbekannt: bei Office-Paketen häufig nur 2% bekannt
- komplexe Funktionen selten selbsterklärend
- Handbücher oft unverständlich/unbrauchbar

→ Wie machen wir es besser?

Struktur

- Frage: wie gliedere ich Informationen im Handbuch?
- abhängig von der Komplexität des Produktes:
 - bei Produkt mit einer Funktion nicht schwer
 - Zusammengehörige Gruppen von Komponenten/Funktionen
- abhängig von der Erwartung des Benutzers:
 - Was interessiert den Benutzer als erstes, zweites ... ?
 - Was muss der Benutzer als erstes, zweites ... wissen?
- abhängig von der Kompetenz des Benutzers
→ besser zu niedrig als zu hoch schätzen

1. Versuch: Produkt-orientierter Ansatz

Beispiel: ein einfacher Taschenrechner

- ① Einschalttaste
- ② Grundrechnungstasten
 - ① Additionstaste
 - ② Subtraktionstaste
 - ③ Multiplikationstaste
 - ④ Divisionstaste
- ③ Funktionstasten
 - ① Sinustaste
 - ② Cosinustaste
 - ③ Exponentialtaste
- ④ Stromversorgung

1. Versuch: Produkt-orientierter Ansatz

- Komponenten des Produktes bestimmen Struktur
- geeignet für Stücklisten, Kataloge
- Produktelemente in den Überschriften
- beschreibender Textstil
- nur geeignet für Benutzer, die sich mit dem Produkt ohnehin auskennen

2. Versuch: Aufgaben- oder Funktions-orientierter Ansatz

Beispiel: ein einfacher Taschenrechner

- ① Rechner einschalten
- ② Grundrechnungen ausführen
 - ① Zahlen addieren
 - ② Zahlen subtrahieren
 - ③ Zahlen multiplizieren
 - ④ Zahlen dividieren
- ③ Winkelfunktionen ausführen
 - ① Sinus berechnen
 - ② Cosinus berechnen
- ④ Batterien einlegen/wechseln

2. Versuch: Aufgaben- oder Funktions-orientierter Ansatz

- Funktionen des Produktes bestimmen Struktur
- Reihenfolge: was *will* der Benutzer zuerst machen?
- Funktionen des Produktes in den Überschriften
- beschreibender oder anweisender Textstil
- geeignet für Benutzer, die den Leistungsumfang des Produktes kennen
- erlaubt schnellen Einstieg

3. Versuch: chronologischer oder Benutzer-orientierter Ansatz

Beispiel: ein einfacher Taschenrechner

- 1 Batterien einlegen/wechseln
- 2 Rechner einschalten
- 3 Grundrechnungen ausführen
 - 1 Zahlen addieren
 - 2 Zahlen subtrahieren
 - 3 Zahlen multiplizieren
 - 4 Zahlen dividieren
- 4 Winkelfunktionen ausführen
 - 1 Sinus berechnen
 - 2 Cosinus berechnen

3. Versuch: chronologischer oder Benutzer-orientierter Ansatz

- mögliche Handlungen bestimmen Struktur
- Reihenfolge: was *muss* der Benutzer zuerst wissen/machen?
- Funktionen des Produktes in den Überschriften
- anweisender Textstil
- gut geeignet für Anfänger
- erlaubt schnellen Einstieg

Handbuch als Nachschlagewerk

- Handbuch wird auch nach häufiger Benutzung gebraucht
- Nachschlagen kann aktiv unterstützt werden:
 - Inhaltsverzeichnis
 - Glossar
 - Stichwortverzeichnis
 - Verzeichnisse für Abkürzungen, Abbildungen, Tabellen, Literatur, ...

Kapitel „Zu dieser Anleitung“

- Anleitung zur Anleitung
- Welchen Sinn hat kursive/fette Schrift?
- besondere Hervorhebung von ...
 - Merksätzen
 - Beispielen
 - Detailbeschreibungen
 - ...
- Erklärung von Schreibweisen und Symbolen

Orientierungshilfen

- Seitenzahlen selbstverständlich
- Kopf- und Fußzeilen nutzen!
→ Überschriften der (Unter-)Kapitel
- Register zum schnellen Nachschlagen
- Einleitungen erläutern Struktur
- Zusammenfassungen dienen zum Querlesen



Code-Dokumentation



Was macht diese Funktion?

```
public static double xyz(double arg0, int arg1){  
    double ret = 1.0, h = 1.0;  
    int f = 1;  
    for(int i=1; i<=arg1; i++){  
        f *= i;  
        h *= arg0;  
        ret += h / f;  
    }  
    return ret;  
}
```

```
/**
 * Nähert die Exponentialfunktion an der Stelle
 * x an. Dazu dient die Taylor-Entwicklung
 * der Exponentialfunktion vom Grad deg
 * um die Stelle 0.
 * @param x Stelle der Exponentialfunktion
 * @param deg Grad des Taylorpolynoms
 * @return Angenäherter Wert
 */
public static double approxExp(double x, int deg) {
    // späterer Rückgabewert, initial  $x^0/0! = 1$ 
    double exp = 1.0;
    // speichert Potenzen von x (initial  $x^0 = 1$ )
    double pow = 1.0;
    // speichert Fakultäten (initial  $0! = 1$ )
    int fak = 1;

    /* Summe von 1 bis deg über  $x^i/i!$  */
    for(int i=1; i<=deg; i++){
        fak *= i;
        pow *= x;
        exp += pow / fak;
    }

    return exp;
}
```

Arten von Kommentaren

- Drei Typen von Kommentaren in Java
- Zeilenkommentare: `// Kommentar`
 - gelten bis zum Zeilenende
- Blockkommentare: `/* Kommentar */`
 - können nicht geschachtelt werden!
- Dokumentationskommentare: `/** Kommentar */`
 - werden von JavaDoc ausgewertet

Taktische Kommentare

- Taktische Kommentare beschreiben einzelne Programmzeilen oder Blöcke von Programmzeilen
- fördern Verständlichkeit der *Implementierung*
- nur das kommentieren, was nicht offensichtlich ist:
`i++; // i wird um 1 erhöht`
- so kommentieren, dass der Code noch nach Jahren verständlich ist
- bei Codeänderungen Kommentare anpassen
- bei gefundenen Fehlern dokumentieren, warum eine Änderung geschehen ist

Kommentieren einzelner Zeilen

- Was macht ein Statement? Vor allem bei verschachtelten Aufrufen
- Bei Variablen:
 - Funktion der Variable
 - warum ein bestimmter Initialwert?

```
// späterer Rückgabewert, initial  $x^0/0! = 1$   
double exp = 1.0;  
// speichert Potenzen von x (initial  $x^0 = 1$ )  
double pow = 1.0;  
// speichert Fakultäten (initial  $0! = 1$ )  
int fak = 1;
```

Kommentieren von Codeblöcken

- Funktion von Schleifen
- Gliederung von Quelltext
- // ab hier geschieht ...

```
//Create and set up the window.  
JFrame frame = new JFrame("HelloWorldSwing");  
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
  
//Add the ubiquitous "Hello World" label.  
JLabel label = new JLabel("Hello World");  
frame.getContentPane().add(label);  
  
//Display the window.  
frame.pack();  
frame.setVisible(true);
```

Strategische Kommentare mit JavaDoc

- Strategische Kommentare beschreiben den Zweck und/oder das Verhalten von Klassen, Methoden und Attributen
- fördern Verständlichkeit der *Modellierung*
- stehen vor der Klasse, dem Attribut, der Methode
- *JavaDoc* erstellt aus strategischen Kommentaren HTML-Dateien
- wertet nur Dokumentationskommentare aus

```
/**  
 * Returns a BigInteger whose value is (this + val).  
 */  
public BigInteger add(BigInteger val)  
    throws ArithmeticException {
```



Kommentar im Code

- erst Kommentar zu Zweck und Verhalten
- HTML-Code verwendbar
- dann spezielle Tags möglich

```
/**  
 * Nähert die Exponentialfunktion an der Stelle  
 * x an. Dazu dient die Taylor-Entwicklung  
 * der Exponentialfunktion vom Grad deg  
 * um die Stelle 0.  
 * @param x Stelle der Exponentialfunktion  
 * @param deg Grad des Taylorpolynoms  
 * @return Angenäherter Wert  
 */
```

Ergebnis als HTML

- JavaDoc wertet Kommentar und Tags aus
- Ergebnis: HTML-Dokument

approxExp

```
public static double approxExp(double x,  
                                int deg)
```

Nähert die Exponentialfunktion an der Stelle x an. Dazu dient die Taylor-Entwicklung der Exponentialfunktion vom Grad deg um die Stelle 0.

Parameters:

x - Stelle der Exponentialfunktion

deg - Grad des Taylorpolynoms

Returns:

Angenäherter Wert

JavaDoc-Tags

Tags für Klassen

@author <author> Erzeuger/Programmierer der Klasse

@see <klass> Verweist auf Dokumentation einer anderen Klasse

@version <version> Versionsnummer der Klasse

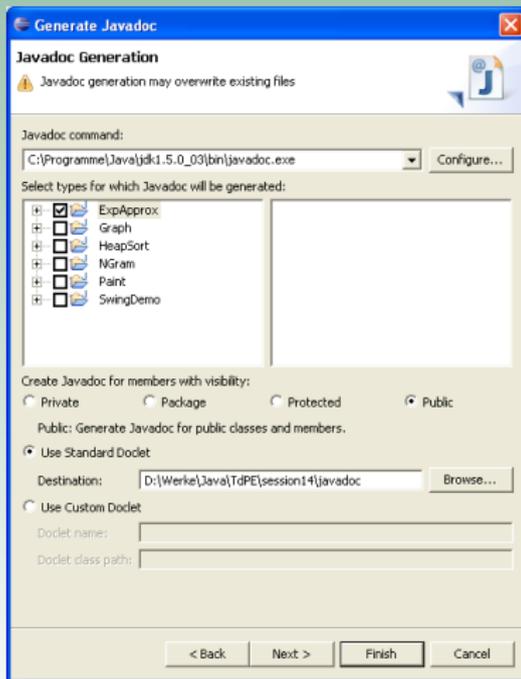
Tags für Methoden

@param <name> <description> Beschreibung eines Parameters

@return <description> Beschreibung des Rückgabewertes

@throws <name> <description> Welche Exception wird wann
geworfen?

JavaDoc in Eclipse



- 1 Eclipse-Menu:
File → *Export...* →
JavaDoc
- 2 Projekt(e) auswählen
- 3 bestimmen, ob auch
protected und private
Methoden ausgewertet
werden
- 4 Zielort angeben
- 5 *Finish*



Danke für die Aufmerksamkeit



Bis nächstes Semester!