Informatik-Praktikum im Grundstudium SS 04

Resümee:
Lernziele und Erfahrungen

Prof. Dr. W. Hesse / J. Berthold

Sommersemester 2004

Philipps-Universität Marburg

INHALT

Kommentar zu den Lernzielen

Arbeitsweise und Organisation

Zusammenfassung der Ergebnisse

Erfahrungen

Ziele des Praktikums

(Folie aus der Einführung)

Team- und Projektarbeit

- ➜ selbstständige Organisation
- ➜ termingerechte und koordinierte Teamarbeit
- ➜ Dokumentation und Präsentation der Ergebnisse
- ➜ Einsatz von Werkzeugen

Vertiefung der Programmierkenntnisse

- ➜ Grundlagen der Softwaretechnik
- ➜ Entwurf und Programmierung im Großen
- ➜ Auswahl und Benutzung verfügbarer Bibliotheken

Lernziele im Detail

Organisation:

- Gruppen organisieren sich selbstständig
- Eine Gruppenleitung übernimmt die Koordination

im Wesentlichen erreicht – je nach Gruppe

LERNZIELE IM DETAIL

Organisation:

- Gruppen organisieren sich selbstständig
- Eine Gruppenleitung übernimmt die Koordination

im Wesentlichen erreicht – je nach Gruppe

Teamarbeit:

- Komponenten in Teams erstellt
- selbstständige Arbeitsteilung in Kleingruppen
- termingerechtes Arbeiten

teils nicht erreicht – “Team”

Einsatz von Werkzeugen:

- Erfahrung mit Werkzeugen wie CVS und ant
- festgelegte Kommunikationsmedien und -standards

erreicht

Einsatz von Werkzeugen:

- Erfahrung mit Werkzeugen wie CVS und ant
- festgelegte Kommunikationsmedien und -standards

erreicht

Dokumentation und Präsentation

- alle Phasen nach gängigen Standards dokumentiert
- strukturierte und verständliche Präsentation

nur teilweise erreicht

professionelles Vorgehen, Softwaretechnik

- Einsatz von geeigneten Bibliotheken
- strukturiertes Vorgehen
- Einsatz von UML

nur teilweise erreicht

professionelles Vorgehen, Softwaretechnik

- Einsatz von geeigneten Bibliotheken
- strukturiertes Vorgehen
- Einsatz von UML

nur teilweise erreicht

Ergebnis:

- stabile Basisversion
- Realisierung von sinnvollen Erweiterungen
- wartungsfreundliche, erweiterbare Software

sei dem Urteil der Teilnehmer überlassen

Ziele des Praktikums

Team- und Projektarbeit

- ➜ selbstständige Organisation $\surd$
- ➜ termingerechte und koordinierte Teamarbeit ($\surd$)
- ➜ Dokumentation und Präsentation der Ergebnisse ($\emptyset$)
- ➜ Einsatz von Werkzeugen $\surd$

ZIELE DES PRAKTIKUMS

Team- und Projektarbeit

- ➜ selbstständige Organisation $\surd$
- ➜ termingerechte und koordinierte Teamarbeit ($\surd$)
- ➜ Dokumentation und Präsentation der Ergebnisse ($\emptyset$)
- ➜ Einsatz von Werkzeugen $\surd$

Vertiefung der Programmierkenntnisse

- ➜ Grundlagen der Softwaretechnik $\emptyset$
- ➜ Entwurf und Programmierung im Großen ($\surd$)
- ➜ Auswahl und Benutzung verfügbarer Bibliotheken $\surd$

ZIELE DES PRAKTIKUMS

Team- und Projektarbeit

- ➜ selbstständige Organisation √
- ➜ termingerechte und koordinierte Teamarbeit (√)
- ➜ Dokumentation und Präsentation der Ergebnisse (∅)
- ➜ Einsatz von Werkzeugen √

Vertiefung der Programmierkenntnisse

- ➜ Grundlagen der Softwaretechnik ∅
- ➜ Entwurf und Programmierung im Großen (√)
- ➜ Auswahl und Benutzung verfügbarer Bibliotheken √

Erfahrung: Probleme beim Programmieren im Großen

SOFTWARETECHNISCHE ASPEKTE

- Probleme der Anforderungsanalyse
- Fehlende Gesamtperspektive
- Unstrukturierte Vorgehensweise
- UML in Präsentationen

ANFORDERUNGSANALYSE

- ➜ ungenaue Anforderungen teils (kritiklos) übernommen
- ➜ teils zu ambitionierte Planung
- ➜ Erweiterte Version nicht strukturiert geplant
  (viele gute Ideen waren vorhanden, ihre Prioritäten nicht)
- ➜ erst während der Implementierung zeigen sich wichtige zusätzliche Anforderungen

VORGEHENSWEISE IN ENTWURF UND IMPLEMENTIERUNG

Schnittstellen zwischen Komponenten:

- ➜ im Entwurf vage geblieben, nur “klare Fälle” beschrieben
- ➜ während der Implementierung nach Bedarf komplettiert
- ➜ im Ergebnis: teils undokumentiert

VORGEHENSWEISE IN ENTWURF UND IMPLEMENTIERUNG

Schnittstellen zwischen Komponenten:

- ➜ im Entwurf vage geblieben, nur “klare Fälle” beschrieben
- ➜ während der Implementierung nach Bedarf komplettiert
- ➜ im Ergebnis: teils undokumentiert

Gesamtperspektive:

- ➜ Gruppen zu sehr auf ihre Komponente konzentriert
- ➜ Zuständigkeiten während der Implementierung verhandelt (statt sie im Entwurf festzulegen)

VORGEHENSWEISE IN ENTWURF UND IMPLEMENTIERUNG

Schnittstellen zwischen Komponenten:

- ➜ im Entwurf vage geblieben, nur “klare Fälle” beschrieben
- ➜ während der Implementierung nach Bedarf komplettiert
- ➜ im Ergebnis: teils undokumentiert

Gesamtperspektive:

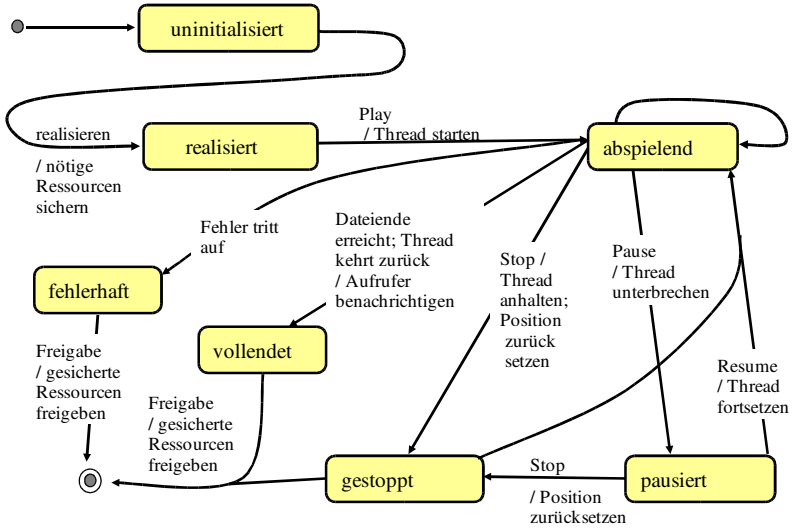
- ➜ Gruppen zu sehr auf ihre Komponente konzentriert
- ➜ Zuständigkeiten während der Implementierung verhandelt (statt sie im Entwurf festzulegen)

Gesamtperspektive ohne festgelegte Schnittstellen:

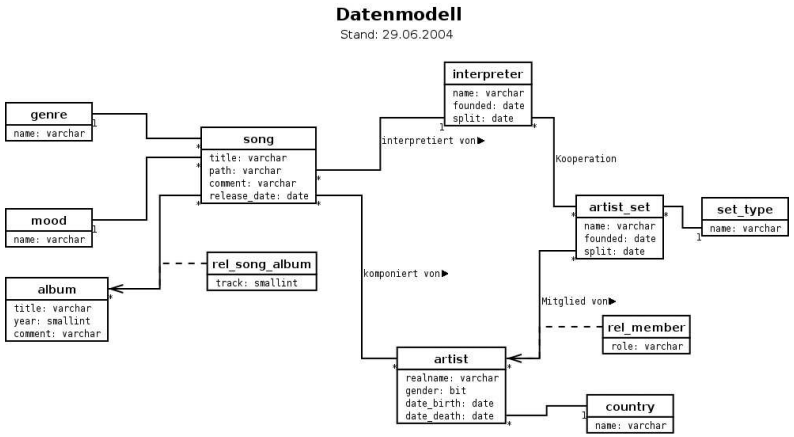
Das nennt man “code and fix” . . .

EINSATZ VON UML, STANDARD-TECHNIKEN

UML: zentraler Inhalt der Vorlesung

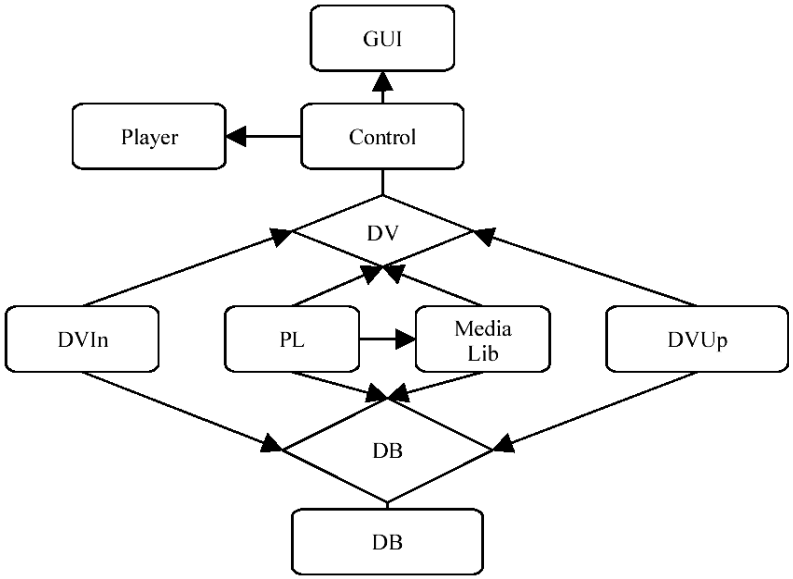


mal originalgetreu:

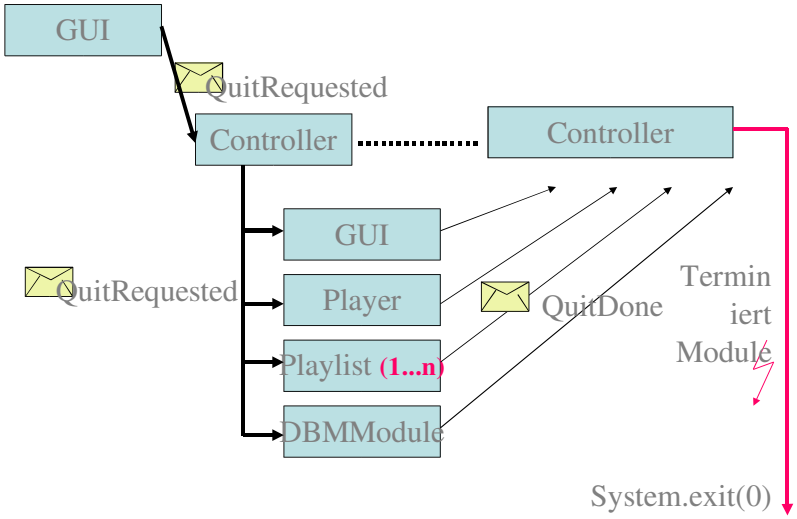


mal weniger:

Komponentendiagramm?



Sequenzdiagramm?



"BEST PRACTICES"[a] , GRUNDLAGEN

Viele Softwareprozesse unterstützen folgende "bewährte Prinzipien":

- ➜ iterative Entwicklung
- ➜ Anforderungen managen und steuern
- ➜ in Komponenten und Schichten entwerfen
- ➜ bewährte visuelle Verfahren anwenden
- ➜ Qualität ständig überprüfen
- ➜ Änderungen kontrollieren und steuern

[a] Whitepaper *Rational Unified Process: Best practices for Software Development Teams*

"BEST PRACTICES"[a] , GRUNDLAGEN

Viele Softwareprozesse unterstützen folgende "bewährte Prinzipien":

- ➜ iterative Entwicklung √
- ➜ Anforderungen managen und steuern ∅
- ➜ in Komponenten und Schichten entwerfen (√)
- ➜ bewährte visuelle Verfahren anwenden (∅)
- ➜ Qualität ständig überprüfen (∅)
- ➜ Änderungen kontrollieren und steuern ∅

Vorlesung: UML, Schichtenmodell, Spezifikationen ∅

Einige wichtige Empfehlungen wurden nicht beachtet.

[a] Whitepaper *Rational Unified Process: Best practices for Software Development Teams*

ORGANISATORISCHE ASPEKTE

(Zu?) Freie Konzeption des Praktikums

- ➜ Motivationsprobleme unvermeidlich
- ➜ kaum Steuerungsmöglichkeiten (U-Boot-Projekte)
- ➜ Risiko eines Fehlschlags in Kauf genommen

ORGANISATORISCHE ASPEKTE

(Zu?) Freie Konzeption des Praktikums

- ➜ Motivationsprobleme unvermeidlich
- ➜ kaum Steuerungsmöglichkeiten (U-Boot-Projekte)
- ➜ Risiko eines Fehlschlags in Kauf genommen

eigentliches Lernziel: Projekterfahrung $\surd$

ORGANISATORISCHE ASPEKTE

(Zu?) Freie Konzeption des Praktikums

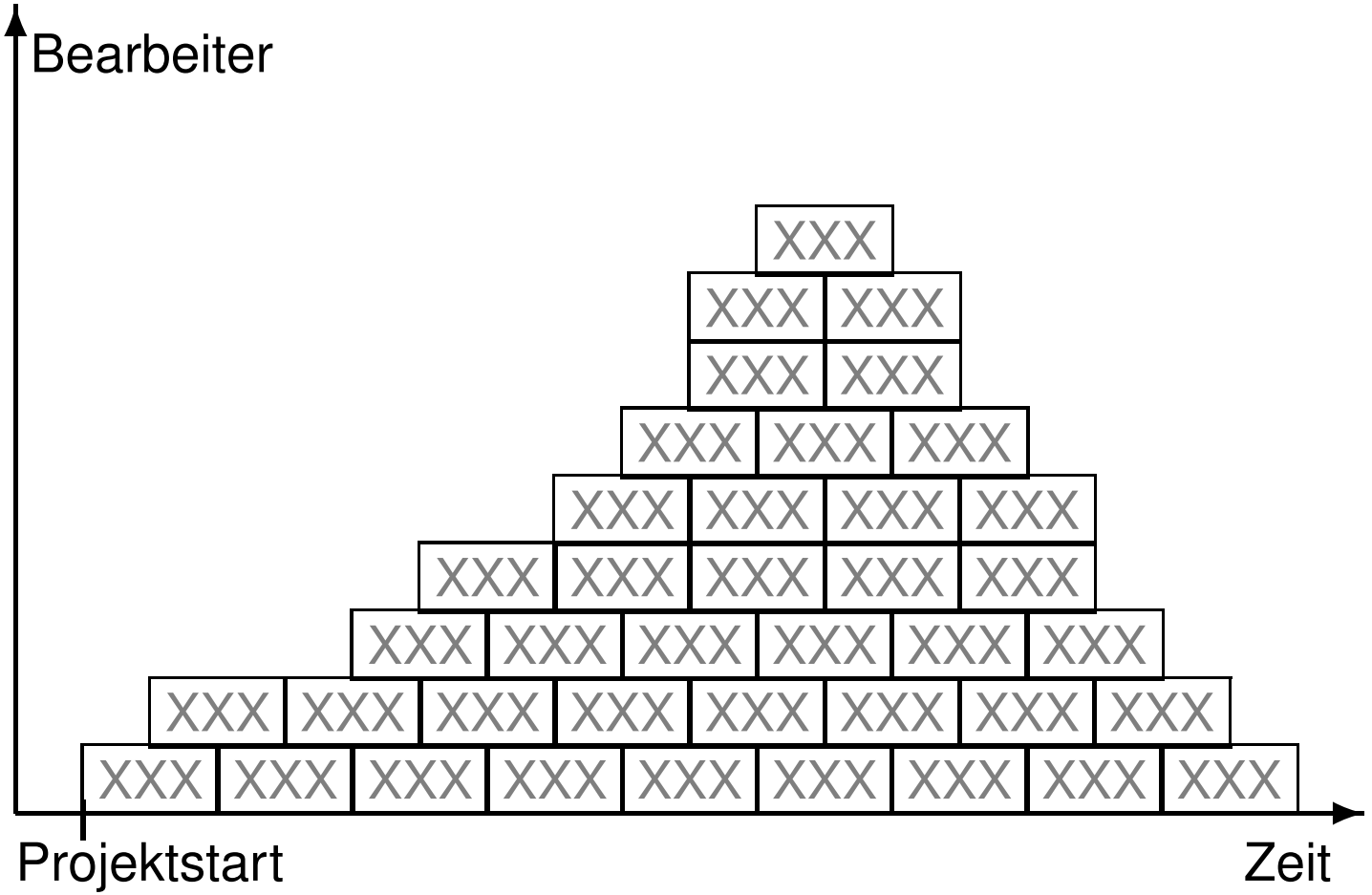
- ➜ Motivationsprobleme unvermeidlich
- ➜ kaum Steuerungsmöglichkeiten (U-Boot-Projekte)
- ➜ Risiko eines Fehlschlags in Kauf genommen

eigentliches Lernziel: Projekterfahrung √

Gruppenorganisation:

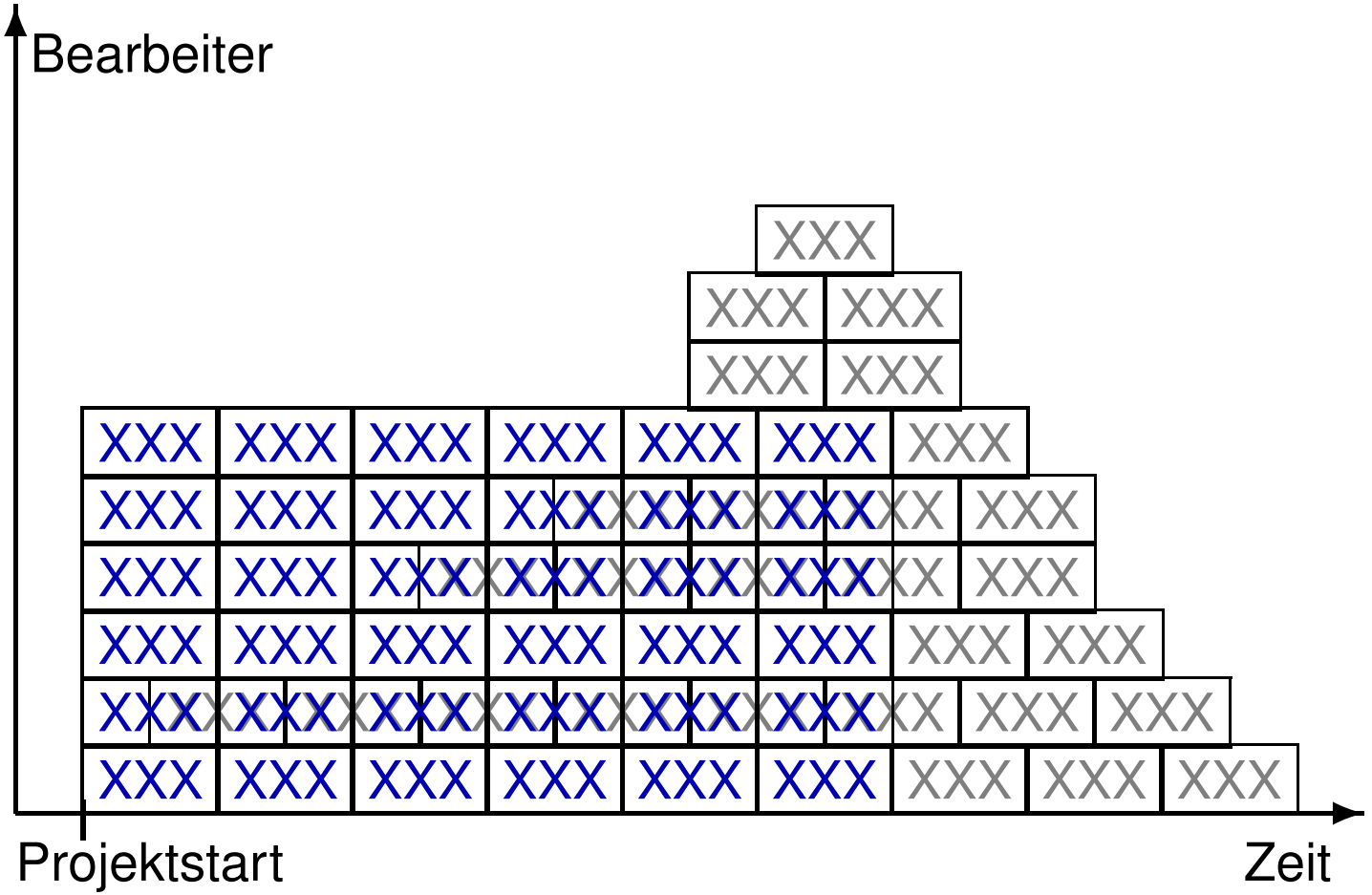
- ➜ zu große Programmiergruppen
- ➜ teils fehlende Koordination dieser Gruppen
- ➜ zu defensive Gruppenleitung (wo vorhanden)
  Kommentare zum Thema “Gruppenleitung” erwünscht

PROBLEM AUFWANDSVERTEILUNG



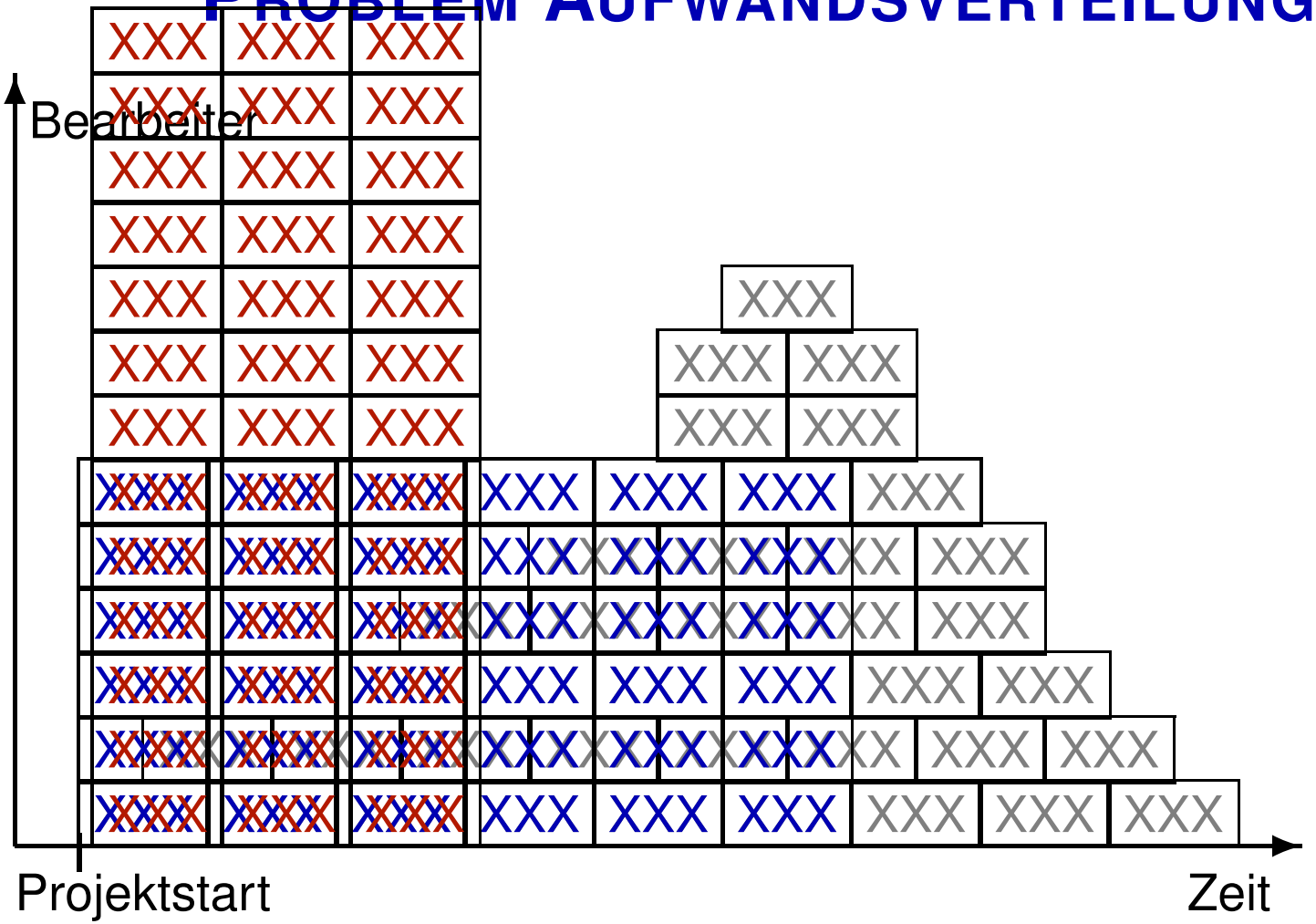
normal

PROBLEM AUFWANDSVERTEILUNG



normal Fo.-Pra.

Problem Aufwandsverteilung



normal Fo.-Pra. Unser Praktikum

ERGEBNIS

	Lines of Code (LOC)	Klassen (ohne innere)
Gruppe 1 (Ivo)	18.600 (+ GUI)	48 (+ GUI: 40)
Gruppe 2 (Aaron)	31.452	302
	davon Message-System:	
	10.262	156

Produktivität:

	Bearb.Monate	LOC / BM
Gruppe 1 (Ivo): 32	96	194
Gruppe 2 (Aaron): 30	90	349

ERGEBNIS

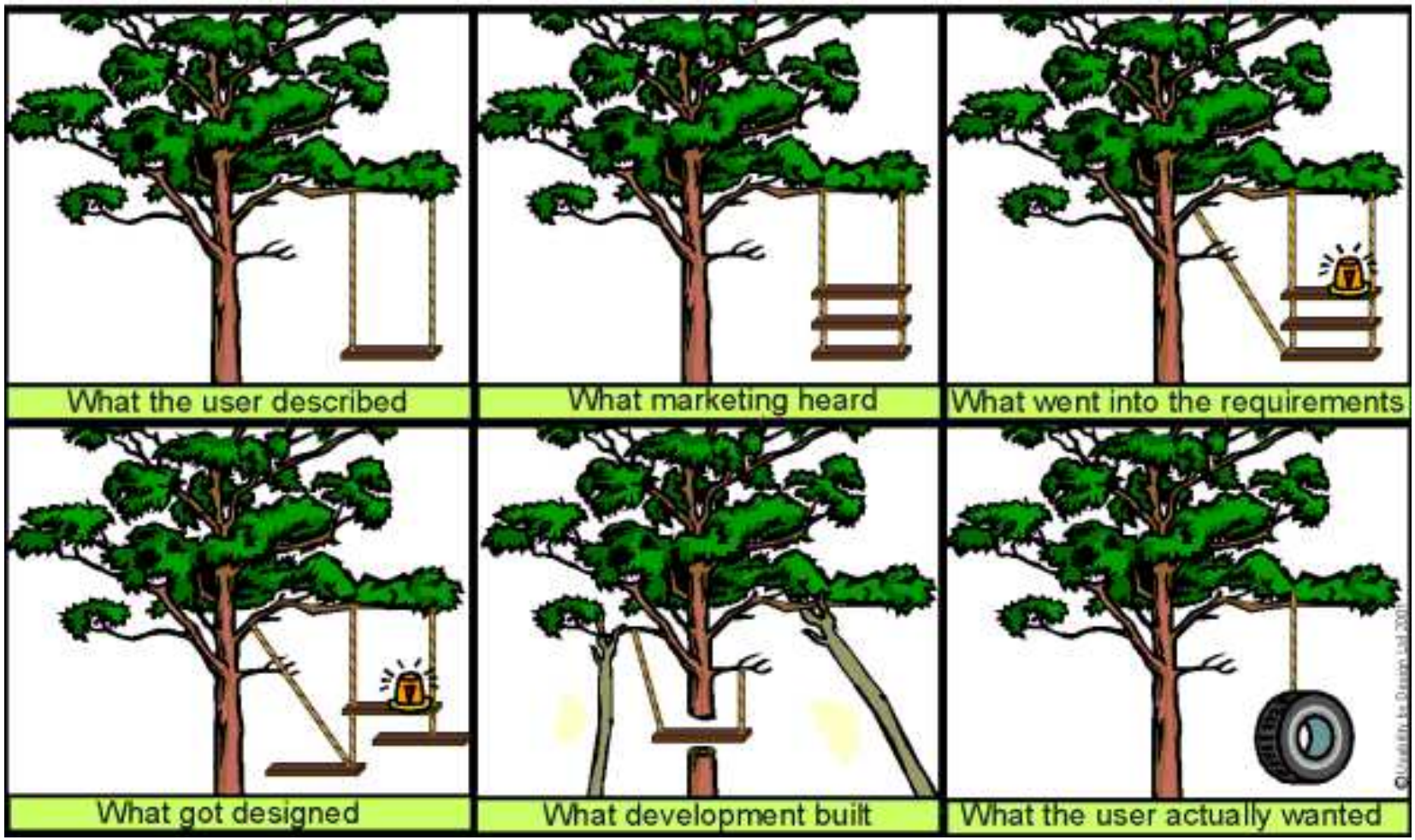
	Lines of Code (LOC)	Klassen (ohne innere)
Gruppe 1 (Ivo)	18.600 (+ GUI)	48 (+ GUI: 40)
Gruppe 2 (Aaron)	31.452	302
	davon Message-System:	
	10.262	156

Produktivität:

	Bearb.Monate	LOC / BM
Gruppe 1 (Ivo): 32	96	194
Gruppe 2 (Aaron): 30	90	349

üblich:
bis 500
(aber oft
weniger!)

ERGEBNIS


What the user described
What marketing heard
What went into the requirements
What got designed
What development built
What the user actually wanted

ERGEBNIS

⊕ termingerechte Lieferung!

- benutzbare Basis-Software (mit einigen Fehlern)
- Gute Ideen, z.T. etwas inkonsequent umgesetzt

  Beispiel: Sprachunterstützung des Skin-Systems lückenhaft
- Pläne für erweiterte Version teils nicht verfolgt

  z.B. Behandlung von austauschbaren Datenträgern
- Schwächen der Analyse:

  Problembereiche werden zu spät erkannt

  Beispiel: Erkennen von Duplikaten

EMPFEHLUNGEN — ZUSAMMENFASSUNG

Ziele einteilen, bewusst Prioritäten setzen

Kleine, aber zielbewußte Schritte machen

Das Gesamtprodukt im Auge behalten

Auf “frühe Phasen” (Analyse und Entwurf) achten

Nach bewährten Standards arbeiten