FAU ‣ inf6 ‣ SpezKap ‣ SS2006

Übersicht/Zusammenfassung

Yves Goergen <yves@unclassified.de> 30.07.2006

Teil 1: Wissensbanken (Prof. Dr. Klaus Meyer-Wegener)

Teil 2: Datenströme (Dipl.-Inf. Michael Daum)

Datenströme und Datenstromverwaltungssysteme Datei: 03 DSMS.pdf
- Was sind Datenströme? 3 ☐
  - Klassifikation von Datenströmen 4 ☐ *[un]strukturiert, anwendungsspezifisch/-neutral, XML-/relationale Datenströme*
  - Formale Definition 5 ☐ *Tupel besitzen durch den Zeitstempel eine Ordnung*
  - Anwendungsgebiete 6 ☐ *Sensordatenverarbeitung (Wetter...), (Internet-)Verkehrsüberwachung, Newsticker, ...*
- DSMS vs. DBMS 7 ☐☐☐☐☐ *DSMS: Persistente Abfragen auf Tupelsequenz; Echtzeitverarbeitung; speichert nicht sondern verarbeitet*
  + hohe Geschwindigkeit, + geringe Produktgröße, – Daten nicht persistent, – mögl. ungenaue Ergebnisse, – Konsistenz unsicher
- Deskriptive Anfragen auf Datenstromquellen 12 ☐☐ *Continous Query Language (CQL): Anlehnung an SQL*
  - Verarbeitung der Daten in Fenstern 14 ☐☐☐ *Beschränkung nach Zeit oder Tupelanzahl*
  - Anfrageergebnis: Relation → Strom 17 ☐ *Insert-/Delete-/Relational-Stream*
  - CQL-Beispiel 18 ☐
- Architektur von DSMS 19 ☐☐
  Input Monitor: Reguliert Eingangsrate
  Data Storage: Working/Summary/Static Storage (Fenster; Zwischenergebnisse; Metadaten)
  Query Repository: Speicher für langlebige Anfragen
  Query Processor: Bearbeitung der Anfragen
  Output Buffer: Zwischenspeicher für Ergebnisse
- Anfrageverarbeitung 21 ☐☐☐
- Sensornetzwerke 24 ☐☐
- Verwandte Themengebiete 26 ☐☐☐ *Embedded software, Kommunikation, Software Engineering, Scheduling, Ereignisgetriebene Architekturen*

Teil 3: Interaction and Usability of Information Systems (Prof. Dr. Janusz Sobecki)

User Modeling File: 04 User Modeling.pdf
Contains knowledge about the individual preferences which determine a user's behavious within the system
- User-centered design 3 ☐☐ *Focus on users, involve them throughout the project lifecycle; measurement of usage*
- User model content 5
  - User data 6 *Information about personal characteristics of the user*
    - Demographic data 7 ☐ *Objective facts: record data, geographics, characteristics, psychographic data...*
    - User knowledge 8 ☐ *Explanatory details to be displayed depending on the user's expertise*
    - User skills and preferences 9 ☐ *User's "knowing how"*
    - User interests and preferences 10 ☐ *Important for recommender systems*
    - User goals and plans 11 ☐ *Limit displayed information to what is relevant to the user's current goals*
  - Usage data 12 ☐ *Data about using computer systems, related to a user's (interactive) behaviour*
    - Observable usage 13 ☐ *Selective actions, temporal viewing behaviour, ratings, purchases, other [dis]confirmatory actions*
    - Usage regularities 14 ☐ *Require further processing: usage frequency, situation–action correlations, action sequences*
  - Environment data 15 ☐ *Usage is influenced by the wide variety of hardware/software/locale characteristics*
    - Software environment 16 ☐ *Browser version and platform, availability of plugins, Java/JavaScript...*
    - Hardware environment 17 ☐ *Bandwidth, processing speed, display devices, input devices*
    - Locale 18 ☐ *Consider local characteristics, adapt to user's environment*
- User profile representation 19 ☐☐☐☐ *Purchase list, web navigation history, [weighted] feature vector*
  - User profile initialisation 23 ☐ *Empty, filled in by a questionnaire*
- Recommendation methods 24 ☐
  - Demographic filtering 25 ☐ *Stereotype reasoning based on demographic features; + simple to implement, – too general*
  - Content-based filtering 26 ☐ *Consider previously evaluated descriptions; – rare explicit evaluations*
  - Collaborative filtering 27 ☐☐ *Consider feedback from other similar users; + subjective data, – requires fair number of similar users*
  - Hybrid method 29 ☐ *Combine above methods to overcome their specific problems*

Interaction Styles File: 05 Interaction Styles.pdf
- Key modal IS 3 *The interface is operated by means of function keys or alphanumeric keyboard*
  - Menu interaction 4 ☐☐☐ *Pop-up menu, pull-down menu, pie menu; menu elements sorting*
    + good for novices (with shortcuts also for experts), + exploration, – information overload, – slow for experts, – not suitable for small display
  - Question and answer 8 ☐ *Questions depend on previous answers; + good for restricting input*
  - Function-key interface 9 ☐ *Using push-buttons, numeric keypads...*
  - Voice interaction 11 ☐ *Mainly used by phone, prerecorded messages; – poor usability (no visual feedback)*
- Direct manipulation IS 12 *The interface may display many objects in one time and the user selects and operates them using special devices*

- Graphical direct manipulation 13 □□ *Visibility of the object of interest; rapid, reversible, incremental actions*
  + visually present task concepts, + easy to leran, + exploration, + recognition, – difficult programming, – not suitable for small display
- Forms fill-in 15 □□ *+ simplifies data entry, + guides the user, – not suitable for small display, – formalisation*
- Linguistic IS
- Command-line interaction 19 □□ *+ flexible, + supports creating macros, – hard to learn and remember, – high error rates*
- Text-based natural language 21 □ *– user must learn understood language subset, – easier to point at something than to describe it*
- Interaction Style selection 22 □ *Combine styles to overcome problems, graphical direct manipulation and menus are popular today*
- Virtual Reality IS 23 *"Not formal" and "not real", yet "true" and "existing"; Intensity, Interactivity, Immersion, Illustrativeness, Intuition*

Usability of the Interactive System File: 06 Usability of the Interactive System.pdf

- What is usability 2 *Usability is good design, ISO 9241*
- 5 "E"s of usability 5 □□
- Effective *Completeness, Accuracy*
- Efficient *Speed, Effort*
- Engaging *Pleasant, Satisfying*
- Error tolerant *Error prevention, Error recovery*
- Easy to learn *Predictability, Consistency*
- Usability assurance methods 7 *Depend on the stage of the project*
- Planning 8 □ *Preliminary meeting, context of use analysis, market analysis*
- Analysis 9 □
- Contextual Inquiry 10 □ *Usually a long-term study*
- Interviews and Focus Groups 11 □ *Users voice their opinions and experiences regarding the product*
- Card Sorting 12 □ *Let users group statements into clusters, statistical analysis*
- Design 13 □
- Heuristic evaluation 14 □□□ *Usability experts evaluate each element of a UI*
- Nielsen ten heuristics *Visibility of system status, Match between system and real world, User control and freedom, Consistency and standards, Error prevention, Recognition rather than recall, Flexibility and efficiency of use, Aesthetic and minimalist design, Help users recognise, diagnose and recover from errors, Help and documentation*
- Wizard of Oz 17 □ *Evaluate unimplemented technology by simulating it by a human*
- Implementation
- Style guides 19 □ *Used to provide a consistent look and feel, can reduce development time*
- Test and measure
- Performance testing 21 □□ *Evaluation of a working system under realistic conditions; measure success rate, task time, user satisfaction*
- Eye tracking 23
- Post release
- User surveys 29 □□ *Often analysed statistically*

Teil 4: Fuzzy Databases and Propagation Theory (Prof. Dr. Dariusz Król)

Query language File: 07 Query Language.pdf

- What is a query language? 3 □ *User friendly, Ad-hoc interactive language, very-high-level programming construct*
- Properties of query languages 4 □ *Data-independent, declarative, natural, efficient, universal*
- A query language in a database environment 5 □ *Integrity constraints, access restrictions*
- Query optimisation 6 □ *Required; Methods: rewriting, indices, caching results*
- OQL vs. SQL3 7 □ *OQL optimised for object retrieval; SQL3 based on relational model, enhanced by OO features*
- Is Java an alternative to query languages? 8 □ *Java is too low-level and not suitable for relational databases*
- Requirements to object query languages 9 □ *Conceptual simplicity, universality, compositionality, high potential for optimisation*
- Coupling a QL with a PL 10 □ *Loose coupling ("embedding"), Tight coupling ("seamless integration")*
- What is impedance mismatch? 11 □ *Incompatibility between PL and QL to be embedded, concerns: syntax, type system, semantics (declarative QL, procedural PL), abstraction levels, name spaces and scoping rules, null values, iteration mechanisms, persistence*
- Object-orientedness in databases 12 □ *No agreements, today's state of the art is premature*
- Complex objects, classes 13 □□
- Encapsulation, interfaces, inheritance 15 □□
- Methods and messages 17 □
- Types, links, class extents, collections 18 □□□□
- The OODBMS ideals 22 □ *Orthogonal persistence (same types), object relativism, total internal identification*
- Syntax, semantics and pragmatics of languages 23 □□□□

8 Seitenangabe
□ Eine Seite