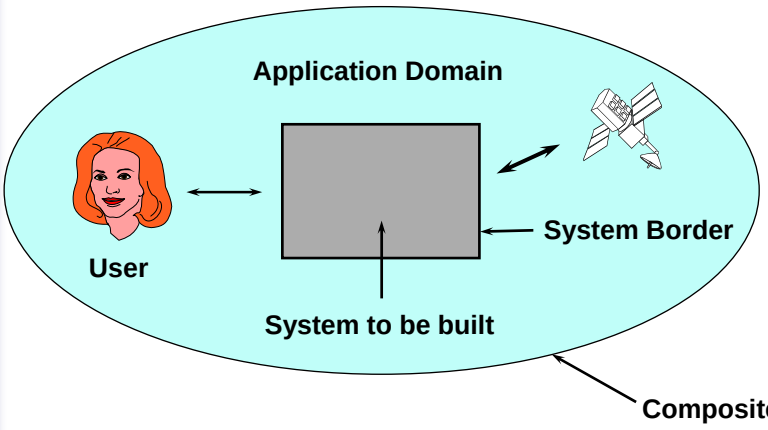
Requirements Meet Interaction Design

Institut für Computertechnik
ICT
Institute of Computer Technology

Hermann Kaindl
Vienna University of Technology, ICT
Austria
kaindl@ict.tuwien.ac.at

System overview



Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

What are requirements?

- User wishes / needs 
- *IEEE Standard:*
  "A condition or capacity needed by a user to solve a problem or achieve an objective."
- "The <*system*> shall be able to ..."
  - system to be built
  - composite system
- *Example:* "The ATM shall accept a cash card."
- Requirements modeling

What are requirements? – In practice

- User requirements documents
- Software/system requirements documents
- Mostly descriptions in natural language
- Representation often unstructured
- Ad hoc process
- Communication problem
- Requirements and use cases?

Traditional UI development

- Based on toolkits employing widgets
- Widgets grouped according to their graphical appearance
- Highly-specialized designers and programmers needed
- Lots of UI code
- Error-prone, low maintainability
- Expensive

Interaction design

- Design of interactions between human and computer
- Relation to requirements engineering
- Relation to task analysis
- No commitment to specific user interface

Scenarios – Stories and narratives

- For representation of
  - cultural heritage
  - explanations of events
  - everyday knowledge
- Human understanding in terms of specific situations
- Human verbal interactions by exchanging stories

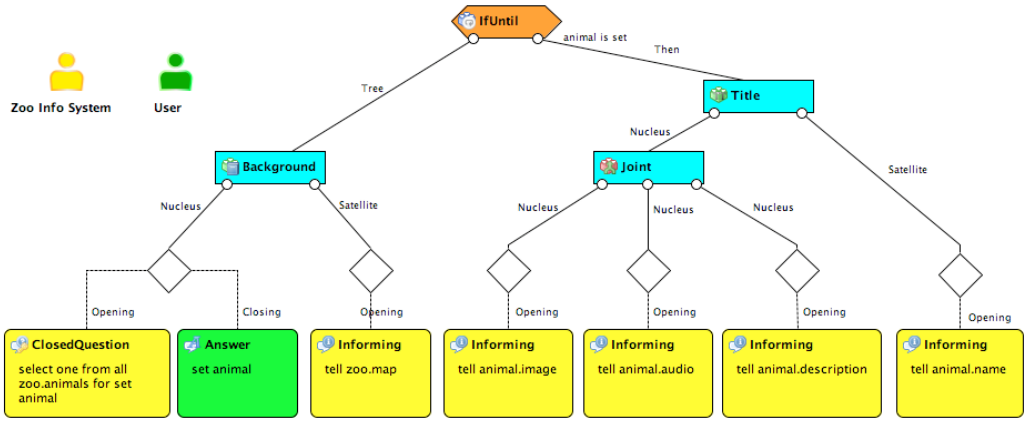
Scripts

- Schank and Abelson
- Script: structure that describes appropriate sequences of events in a particular context
- Handles well-known everyday situations
- Predetermined and stereotyped sequence of actions

Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

Discourse Example



Discourse Model

Discourse "atoms" and "molecules"

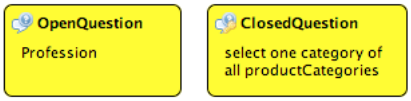
- Metaphorical view
  - Communicative acts as atoms
  - Adjacency pairs as molecules
- Communicative acts instead of RST text portions
  - Interaction instead of text
- Two dimensions
  - Tree with discourse relations (monologue)
  - Adjacency pair (dialogue)
- Integration of RST and procedural constructs with Conversation Analysis

Speech acts

- John R. Searle
- Theory from philosophy of language
- Human speech also used to do something with intention — to act
- "Speaking a language is performing speech acts, act such as making statements, giving commands, asking questions and so on"
- Speech acts: basic units of language communication
- Communicative acts: abstraction from speech

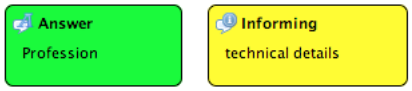
Communicative Acts – Open & Closed Question

- Open Questions enable asking for a particular type of information, respectively, an instance of a domain class.
- Closed Questions restrict the possible answer to a list of provided domain instances to choose from.



Communicative Acts – Informing & Answer

- Both are used to convey information.
- Answer communicative acts are always directly related to questions, whereas Informing is uttered standalone or together with acknowledgment.



Communicative Acts – Request

Used to request the communication partner to act. Thus, the propositional content of a request is always an action that has to be carried out. The action can be defined either for the given application, or it can be the request to utter a particular communicative act.

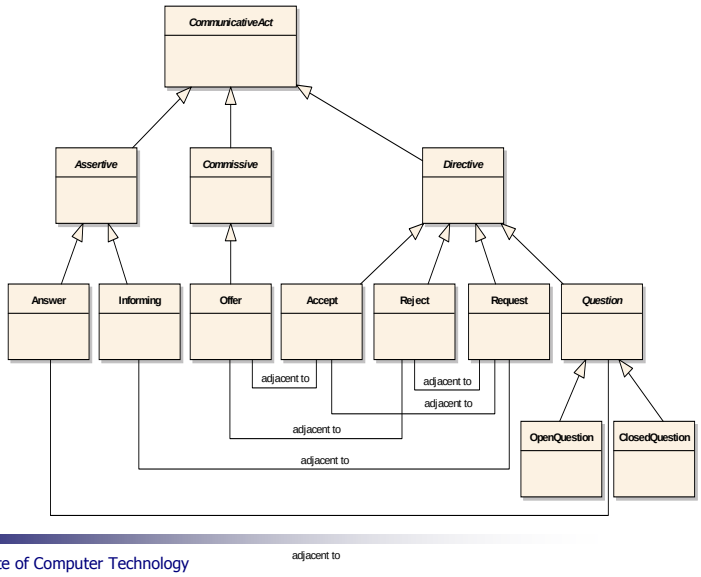
Communicative Acts – Offer

Offers to carry out an action or to add information to the shared knowledge.

Communicative Acts – Accept & Reject

Accept and Reject provide for accepting or rejecting a particular request or offer.

Communicative Acts Taxonomy

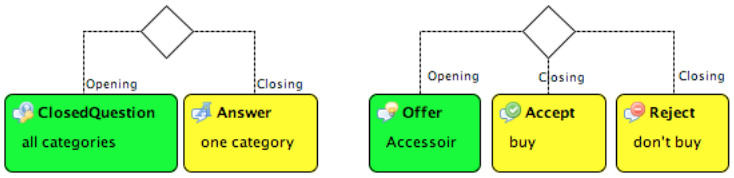


Conversation Analysis

- Harvey Sacks; Luff, Gilbert and Frohlich
- Theory from sociology
- Focus on sequences of naturally-occurring talk "turns"
- To detect patterns that are specific to human oral communication
- Adjacency pair: e.g., a question should have a related answer
- Inserted sequence: subordinate interactions

Adjacency Pair

- Relates an initial communicative act with one subsequent communicative act or two alternative subsequent communicative acts.
- Typical adjacency pairs of communicative acts are:
  - ClosedQuestion–Answer, OpenQuestion–Answer
  - Offer–Accept, Offer–Reject
  - Request–Informing, Request–Accept, Request–Reject



Rhetorical Structure Theory (RST)

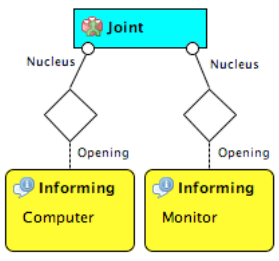
- Mann and Thompson
- Linguistic theory
- Internal relationships among text portions and associated constraints and effects
- Relationships in a text are organized in a tree structure
- Rhetorical relations associated with non-leaf nodes, and text portions with leaf nodes

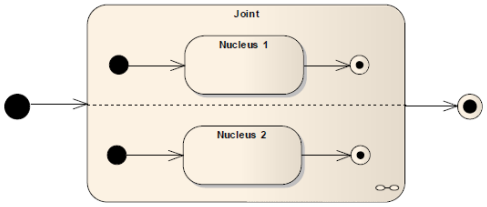
RST relations (in our approach)

- Nucleus: the main part of the communication
- Satellite: the helper part
- Communicative acts instead of text portions

RST relation – Joint

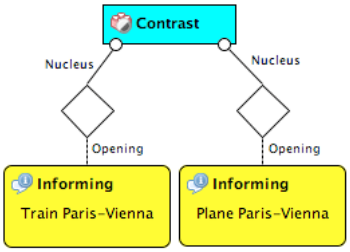
Relates independent subtrees with communicative acts (of the same kind). It does not imply any order. So, it is also possible to issue both nuclei concurrently (e.g., on a GUI).





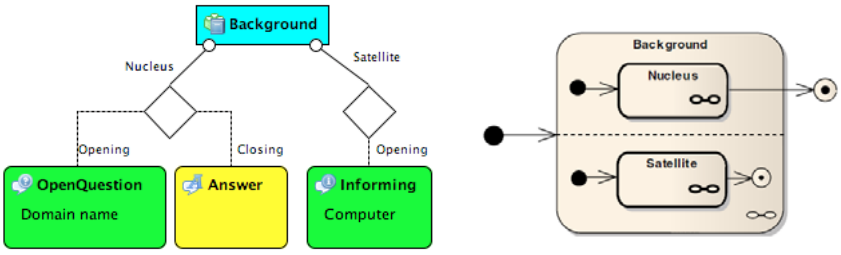
RST relation – Contrast

Relates similar subtrees and compares them with respect to differences.



RST relation – Background

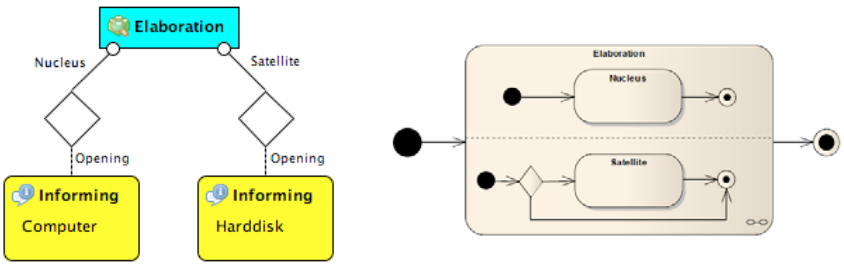
- General information of any sort that is likely to help understand the nucleus.
- Thus, satellite of the background relation shall only contain Informing communicative acts.



RST relation – Elaboration

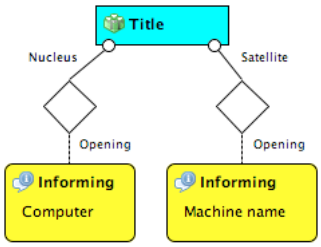
- Satellite contains additional detail about some element of subject matter which is presented in the nucleus, in one or more of the ways listed below (nucleus :: satellite):
  - set :: member
  - abstraction :: instance
  - whole :: part
  - process :: step
  - object :: attribute
  - generalization :: specific
- The communicative acts can also be questions, for example, if one communicative partner wants to figure out additional details about the subject matter.

RST relation – Elaboration (cont.)



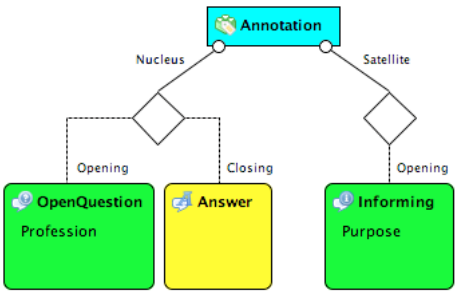
RST relation – Title

Specialization of Elaboration, restricting the additional detail of some element of subject matter to a short description, either title or caption.



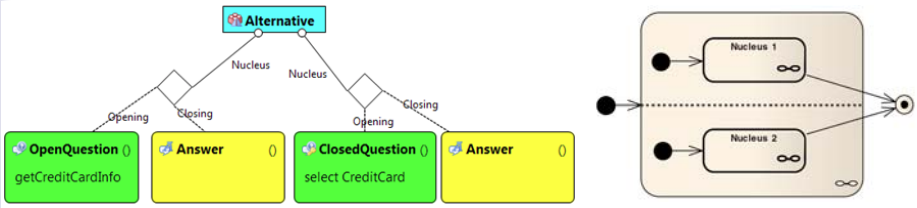
RST relation – Annotation

Another specialization of Elaboration, restricting the additional detail of some element of subject matter to meta information.

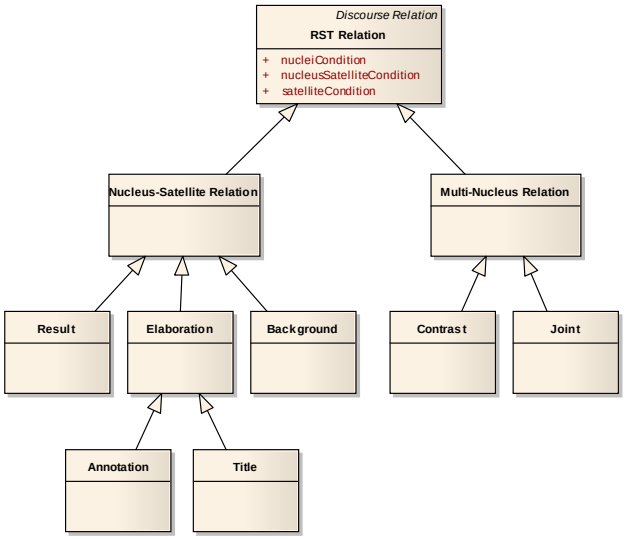


RST relation – Alternative

- Represents that the discourse can be performed in either branch.
- Both branches can be started to present alternatives, but only one branch can be finished.

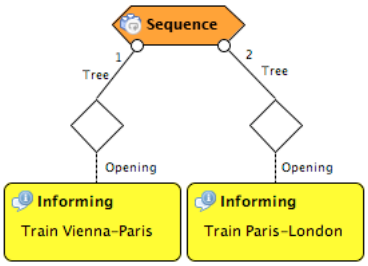


Taxonomy of RST relations



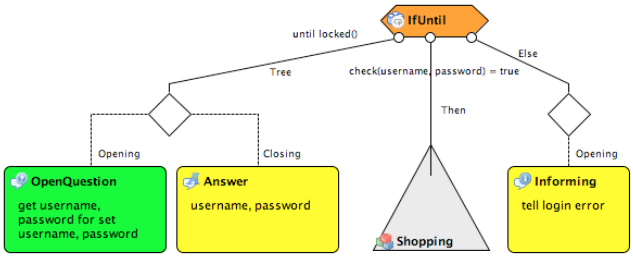
Procedural construct – Sequence

Defined order of uttering the communicative acts or subtrees.

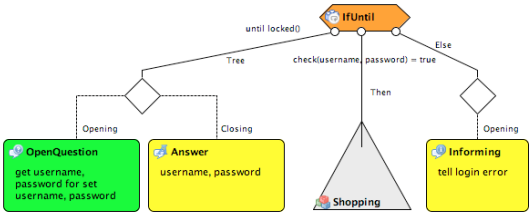


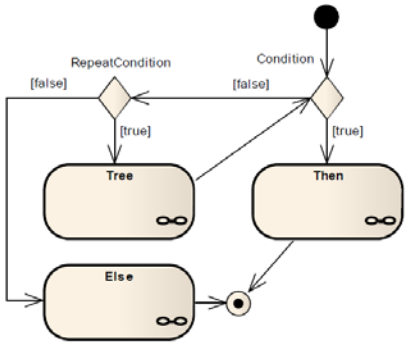
Procedural construct – IfUntil

- If-statement combined with a conditional loop
- Utterance of the <Then> subtree depends on successful execution of the related Condition.
- Repetition of the <Tree> branch until Condition becomes fulfilled, while RepeatCondition is fulfilled



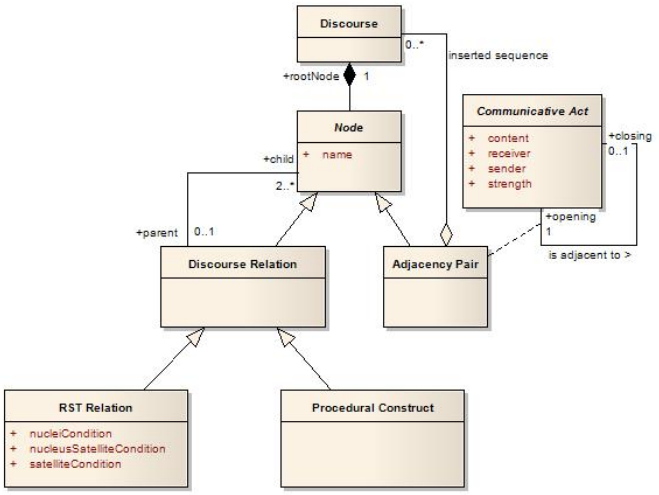
Procedural construct – IfUntil (cont.)





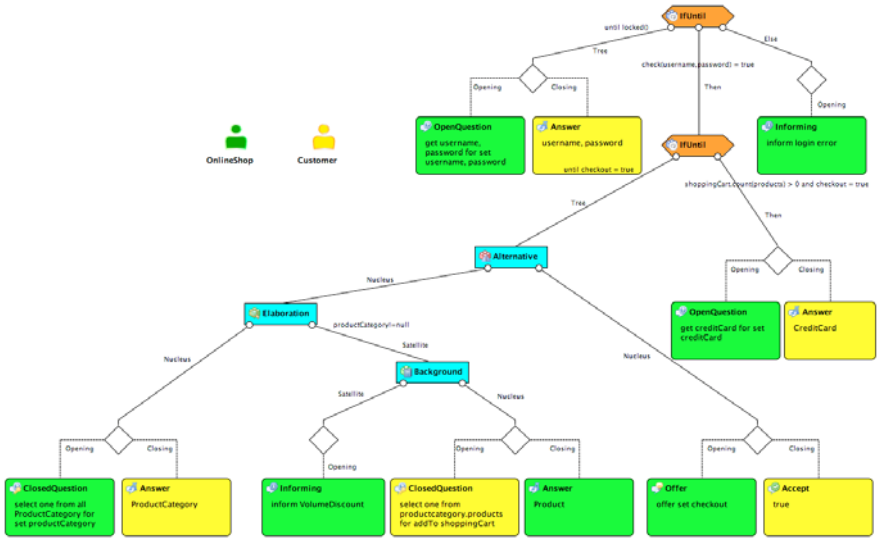
Institute of Computer Technology

Conceptual Discourse Metamodel



Institute of Computer Technology

Example model



Ontologies

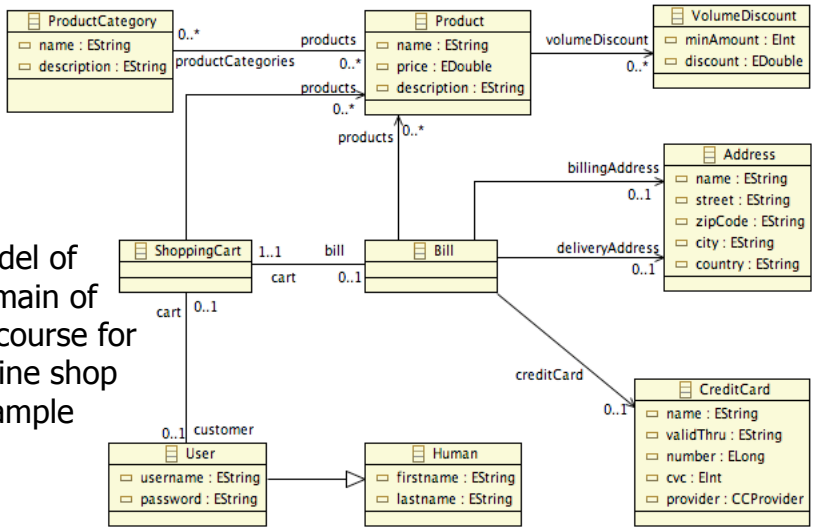
- Tom Gruber
- Actually, the old Greeks
- Domain models
- Conceptualizations of a domain
- Often using taxonomies and object-based ideas
- Ontology languages based on knowledge-representation theories
- E.g., OWL based on description logic

Domain representation

- Speech act usually talks about something in the domain of discourse
- Model of the domain
- Integration and use of ontologies

Ontologies

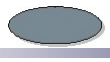
Model of domain of discourse for online shop example



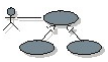
Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

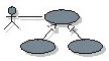
Use cases

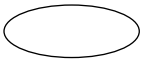


- "particular cases of how the system is to be used"
- Use-Case Report (according to Unified Process):
  1. Brief Description
  2. Flow of Events
  3. Special Requirements
  4. Pre-conditions
  5. Post-conditions
  6. Extension Points
  7. Relationships
  8. Use-Case Diagrams
  9. Other Diagrams



Use-case diagram

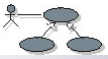
- UML graphical notation 
- Ellipse: use case

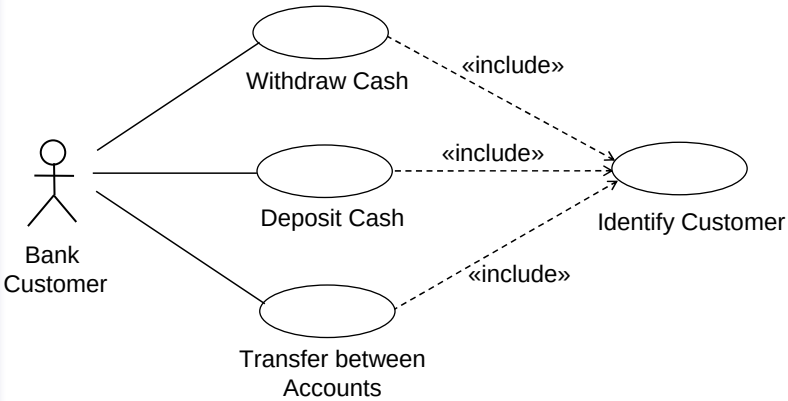


- Stick man: actor

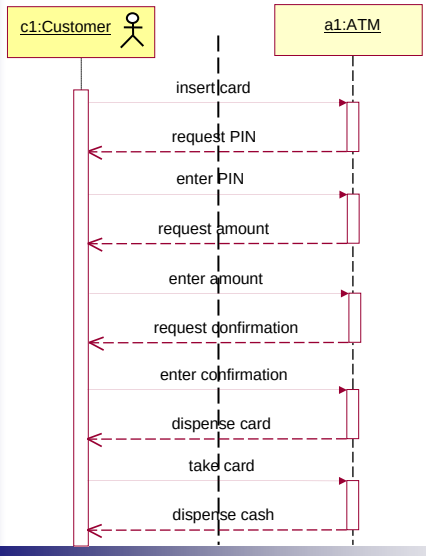
- Connecting line: association

Use-case diagram



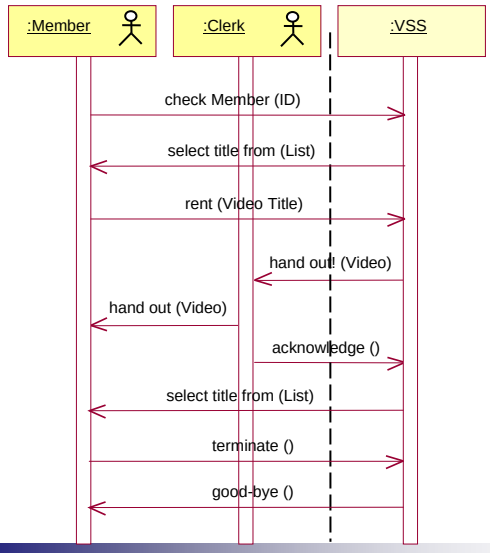


OOA model – UML sequence diagram



- Represents a scenario
- Interaction of instances
- Activation
- System border

OOA model – UML sequence diagram



Unnamed instances
Concurrent objects

Specification based on discourse model

- Scenario: focus on thread of events or actions
- Difficult to specify variations in Use-Case Report
- Discourse model: specification of class of dialogues
- Possible flows well defined and understandable
- Additional information in RST relations

Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

Exercise

- **Given at the tutorial**

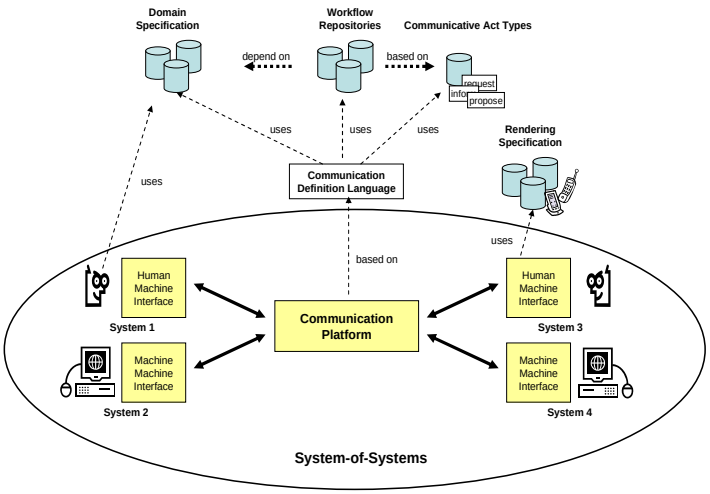
Questionnaire

Voluntarily, please fill in the subjective questionnaire at http://ontoucp.org/cms/technology/questionnaire.html

Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

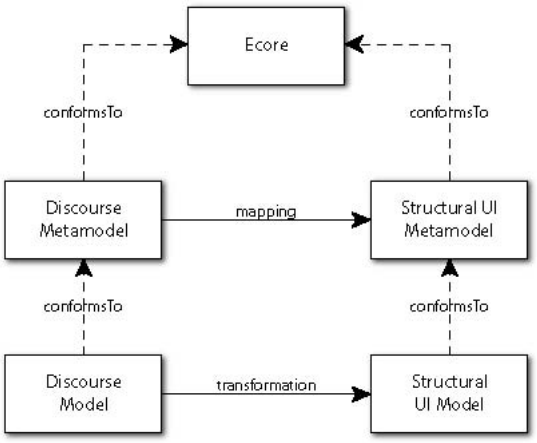
Unified Communication Platform



Automated generation of user interfaces

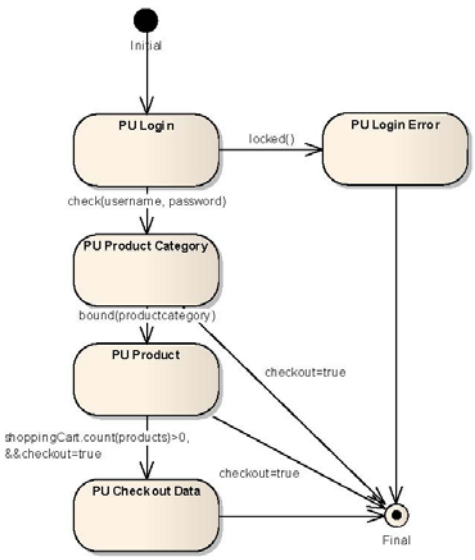
- Essential steps
  - Generation of structural UI model
  - Generation of finite state machine
  - Rendering of UI
- Even for multiple platforms

Generation of structural UI model



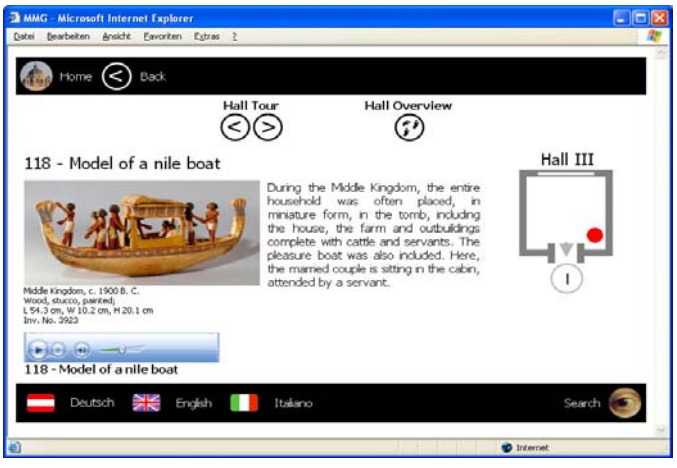
Transformation process according to MDA (model-driven architecture)

Finite State Machine



Maximum presentation unit state machine for the online shop example

Rendered User Interface



Guide in Kunsthistorisches Museum Wien

Outline

- Background
- Interaction design based on discourse modeling
- Use case specification
- Exercises
- Sketch of automated user-interface generation
- Summary and Conclusion

Summary and Conclusion

- Interaction design can be based on discourse modeling.
- These models can be also viewed as specifying classes of scenarios, i.e., use cases.
- Requirements meet interaction design to make applications both more useful and usable.

Literature

- Carroll, J. M., (editor), *Scenario-Based Design: Envisioning Work and Technology in System Development*. New York, NY: John Wiley & Sons, 1995.
- Luff, P., Gilbert, N., Frohlich, D., (eds.), *Computers and Conversation*, Academic Press, 1990.
- Mann, W.C., and Thompson, S.A. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(3): 243–281, 1988.
- Searle, J.R. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge, England, 1969.
- Schank, R. C., and Abelson, R. P., *Scripts, Plans, Goals and Understanding*. Hillsdale, NJ: Lawrence Erlbaum, 1977.

Selected work of this tutorial presenter

- Bogdan, C., Falb, J., Kaindl, H., Kavaldjian, S., Popp, R., Horacek, H., Arnautovic, E., and Szep, A., "Generating an Abstract User Interface from a Discourse Model Inspired by Human Communication". In *Proceedings of the Forty-first Annual Hawaii International Conference on System Sciences (HICSS-41)*, p. 10, Hawaii, 2008, IEEE Computer Society Press.
- Bogdan, C., Kaindl, H., Falb, J., and Popp, R., "Modeling of interaction design by end users through discourse modeling". In *Proceedings of the 2008 ACM International Conference on Intelligent User Interfaces (IUI'08)*, Gran Canaria, Spain, 2008. ACM Press, pp. 305–308.
- Falb, J., Kaindl, H., Horacek, H., Bogdan, C., Popp, R., and Arnautovic, E., "A discourse model for interaction design based on theories of human communication". In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, New York, NY, USA, 2006. ACM Press, pp. 754–759.
- Falb, J., Kavaldjian, S., Popp, R., Raneburger, D., Arnautovic, E., and Kaindl, H., "Fully Automatic User Interface Generation from Discourse Models". In *Proceedings of the 2009 ACM International Conference on Intelligent User Interfaces (IUI'09)*, ACM. Sanibel Island, Florida, USA, 2009. ACM Press. Tool demo paper.

Selected work of this tutorial presenter (cont.)

- Falb, J., Popp, R., Röck, T., Jelinek, H., Arnautovic, E., and Kaindl, H., "UI Prototyping for Multiple Devices Through Specifying Interaction Design". In *Proceedings of IFIP INTERACT 2007, LNCS 4662, Part I.* Heidelberg, Germany, 2007. Springer, pp. 136–149.
- Kaindl, H., Is Object-Oriented Requirements Engineering of Interest?, *Requirements Engineering*, 10, 2005, pp. 81–84.
- Kavaldjian, S., Bogdan, C., Falb, J., and Kaindl, H., "Transforming Discourse Models to Structural User Interface Models". In *MoDELS 2007 Workshops, LNCS 5002*. Heidelberg, Germany, 2008. Springer, pp. 77–88.
- Popp, R., Falb, J., Arnautovic, E., Kaindl, H., Kavaldjian, S., Ertl, D., Horacek, H., and Bogdan, C., "Automatic Generation of the Behavior of a User Interface from a High-level Discourse Model". In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS-42)*, p. 10, Hawaii, 2009, IEEE Computer Society Press.