VROOM: Virtuelle Realität und Objekt-Orientierte Modellierung

Martin Gogolla

29. April 1999

Zusammenfassung

Inhaltliches Hauptziel des Projektes ist die Erarbeitung eines Visualisierungssystems, das Diagramme aus dem Software-Entwurf (z.B. Objekt- und Sequenzdiagramme) animiert und es gestattet, eine Szene aus verschiedenen Blickwinkeln zu betrachten und zu analysieren.

1 Thema

Virtuelle Realität und Objekt-Orientierte Modellierung (VROOM)

2 Veranstalter des Projekts

Martin Gogolla

3 Erläuterung des Themas

3.1 Projektkontext

Das geplante Projekt wird sich innerhalb der folgenden drei Kontexte bewegen.

Kontext Objektorientierter Entwurf: Innerhalb der letzen Jahre hat sich die Sprache Unified Modeling Language (UML) [BJR97, BJR99b, BJR99a] zum Entwurf und zur Dokumentation von (objektorientierter) Software als federführend herauskristallisiert. Die Sprache gestattet es, verschiedene Diagrammarten (u.a. Klassendiagramme, Objektdiagramme und Sequenzdiagramme) für unterschiedliche Aufgaben in der Software-Entwicklung zu verwenden.

Klassische Diagrammarten in der Informatik sind beispielsweise Automaten [SM92], Petri-Netze [Rei85], Entity-Relationship-Diagramme [Che76, Gog94], Daten- und Kontrollfluss-Diagramme [GJM91] und Syntaxdiagramme [GJM91]. Beispiele für allgemeine Ansätze, die die genannten speziellen Diagrammarten einbeziehen, sind die Object Modeling Technique OMT [RBP+91], die Methode zur objektorientierten Analyse und Design OOA nach [CY90], der Ansatz nach Booch [Boo91], der "Use-Case-Driven"-Ansatz [JCJÖ92], und insbesondere die Unified Modeling Language UML, die versucht, die zuvor genannten Notationen in eine einheitliche Form zu bringen.

Aber: "The UML notation is intended to be drawn on 2-dimensional surfaces. ... In the near future, true 3-dimensional layout and navigation may be possible ...; however, it is not currently practical." [BJR97]

Kontext Software-Engineering-over-the-Web: Aktuell gibt es starke Tendenzen in der Software-Technik zur verteilten Anwendungsentwicklung. Dies bedeutet, dass die Kommunikation zwischen Auftraggeber und Entwickler und auch die Kommunikation innerhalb des Entwicklerteams über das Web unterstützt werden soll.

Diese Entwicklung wird begleitet von Bestrebungen, Groupware-Ansätze für die Software-Entwicklung stärker zu unterstützen. Das Aufkommen von Standards wie XML oder RDF, die den Austausch von komplexen Objekten über das Web ermöglichen, begünstigt diese Tendenzen.

Kontext Visualisierung: Die Informatik hat in ihrer kurzen Geschichte in vielen Arbeitsgebieten für die Darstellung von komplexen statischen Sachverhalten auf Graphen (Diagramme, Bilder) zurückgegriffen, und die Darstellung von komplexen Sachverhalten die dynamische Elemente beinhalten wird neuerdings durch Animationen (bewegte Bilder) realisiert.

Moderne technische Plattformen, wie z.B. VRML [MC97] oder Java 3D, unterstützen solche Ansätze zur Visualisierung.

3.2 Projektziele

Inhaltliches Hauptziel des Projektes ist die Erarbeitung eines Visualisierungssystems, das Diagramme aus dem Software-Entwurf (z.B. Objekt- und Sequenzdiagramme) animiert und es gestattet, eine Szene aus verschiedenen Blickwinkeln zu betrachten und zu analysieren.

Entwurfsdokumente und Animationen sollen im Web kommunizierbar sein und zwar sowohl zwischen Auftraggeber und Entwickler als auch innerhalb des Entwicklerteams.

3D-Graphiken sollen zur Darstellung von komplexen Systemzuständen und Animationen zur Darstellung von Systementwicklungen eingesetzt werden. Systementwicklungen werden ausgelöst durch Zustandsänderungen wie z.B. Attributänderungen, Senden von Nachrichten oder die Einrichtung von Beziehungen zwischen Objekten.

Die Abbildungen 1 und 2 verdeutlichen beispielhaft diese Ziele: In Abbildung 1 ist ein UML-Sequenz-Diagramm wiedergeben, das das Zustandekommen eines Telefongesprächs darstellt; in der Abbildung 2 sind Schnappschüsse aus einer Animation der Nachrichten-Sequenz dargestellt; Objekte sind als Kugeln repräsentiert während in der Animation die Nachrichten sich auf einem Kommunikationskanal vom Sender zum Empfänger bewegen; die Bewegungsrichtung in der Animation ist hier durch dunkelgraue Pfeile verdeutlicht; die Schnappschüsse in der Abbildung 2 sind aus unterschiedlichen Perspektiven und zu unterschiedlichen Zeitpunkten aufgenommen.

3.3 Bedeutung des Themas

Innerhalb der Informatik ist die Bedeutung des Themas vor allen Dingen darin zu sehen, dass versucht werden soll, sich von konkreten, implementierungsnahen Beschreibungen zu lösen und sich hin zu mehr konzeptionellen, anwendungsbezogenen Beschreibungen zu wenden und diese auch geeignet darzustellen.

3.4 Einbeziehung gesellschaftlicher Aspekte

Ein gesellschaftlicher Aspekt liegt darin, Software-Spezifikationen durch Visualisierung so zu gestalten, so dass auch Nicht-Experten in der Lage sind, die Spezifikationen zu verstehen bzw. sie vielleicht sogar selbst zu formulieren.

4 Plan der vorgesehenen Lehrveranstaltungen

4.1 Wintersemester 1999/2000

Pflichtveranstaltungen			
VROOM	AV	4 SWS	Gogolla, N.N.
Objektorientierte Software-Entwicklung	K	4 SWS	Gogolla
Empfohlene Veranstaltungen			
Seminar Sichere Systeme	S	2 SWS	Gogolla et al.
LV im Bereich "Graph-Ersetzung"	K	4 SWS	Kreowski, Drewes

4.2 Sommersemester 2000

Pflichtveranstaltungen			
VROOM	AV	4 SWS	Gogolla, N.N.
Empfohlene Veranstaltungen			
LV im Bereich "Virtuelle Realität/VRML"	K	4 SWS	N.N.
Seminar Sichere Systeme	S	2 SWS	Gogolla et al.

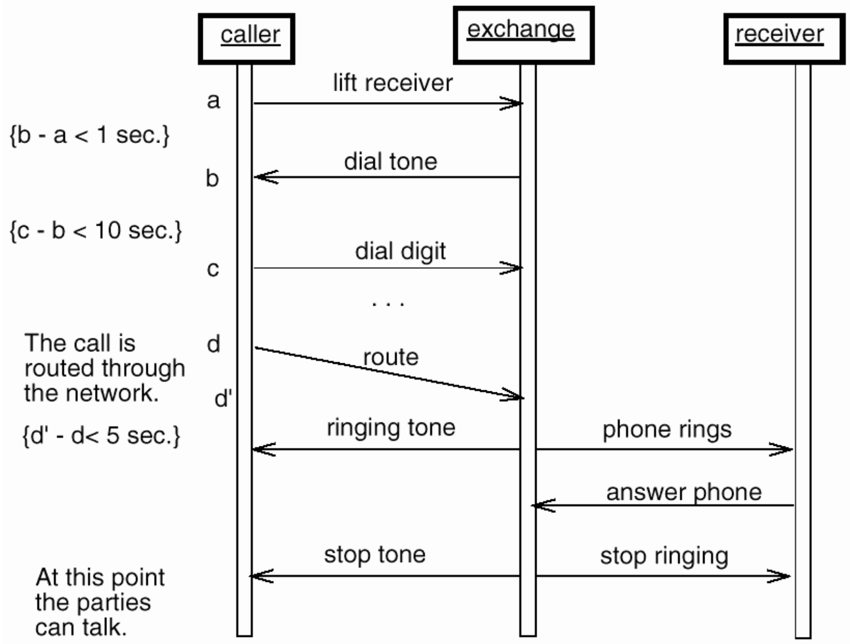

Abbildung 1: UML Sequenz-Diagramm

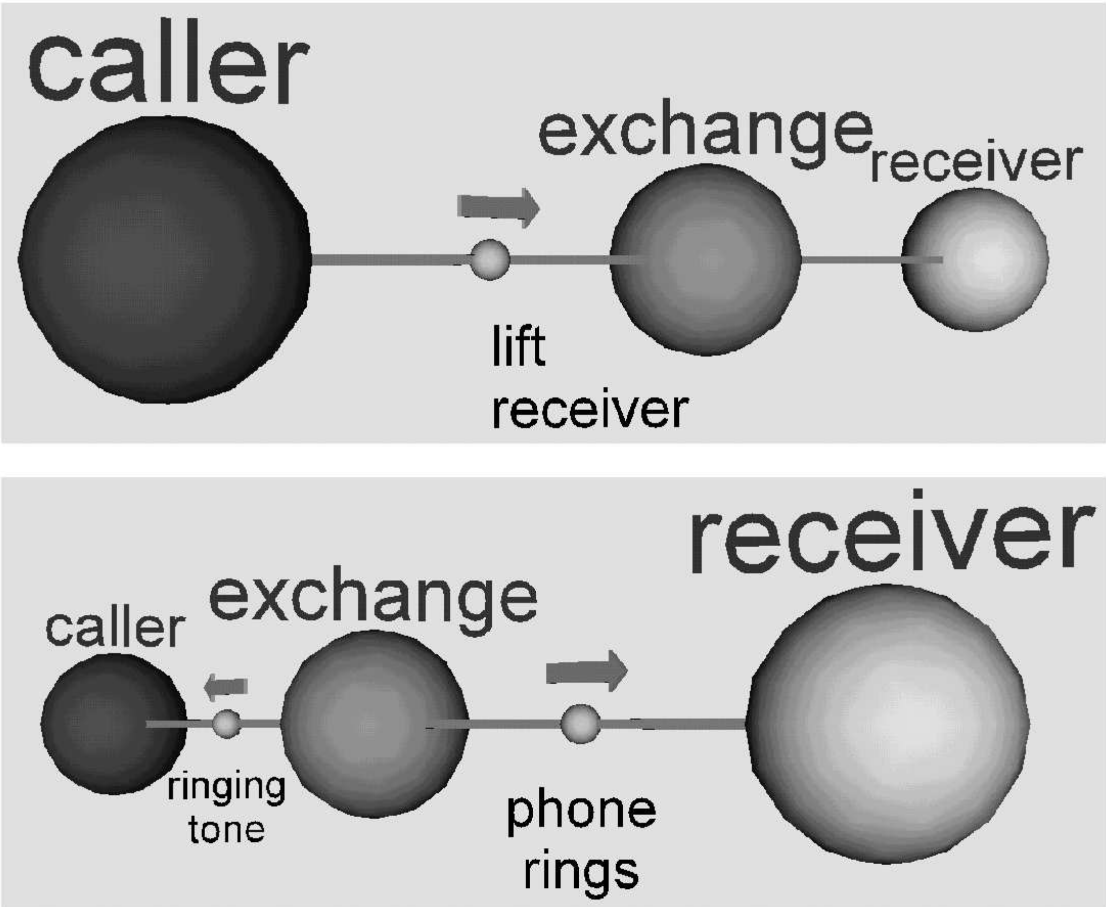

Abbildung 2: Ausschnitte aus einer Animation zum Sequenz-Diagramm

4.3 Wintersemester 2000/2001

Pflichtveranstaltungen			
VROOM	AV	4 SWS	Gogolla, N.N.
Empfohlene Veranstaltungen			
Seminar aus dem Bereich “Datenbanksysteme”	S	2 SWS	Gogolla
LV im Bereich “Graphische DV”	K	4 SWS	Nake

4.4 Sommersemester 2001

Pflichtveranstaltungen			
VROOM	AV	4 SWS	Gogolla, N.N.
Empfohlene Veranstaltungen			
Seminar aus dem Bereich “Software-Technik”	S	2 SWS	Gogolla

5 Prüfungsgebiete

Software-Technik (78), Datenbanksysteme (73)

6 Angaben zur Realisierbarkeit des Projekts

6.1 Personal

Martin Gogolla, N.N. (Oliver Radfelder, Mark Richters; abhängig von Vertragsverlängerungen)

6.2 Besondere finanzielle Mittel

./.

6.3 Hardware und Software

Nutzung des FB3 Netzes und der Hard- und Software der AG DB; Neuanschaffungen durch Mittel der AG DB realisierbar

6.4 Forschungssemester während der Projektlaufzeit

Momentan Forschungssemester für Martin Gogolla im 4. Projektsemester geplant; wird im Falle der Projektrealisierung gegebenenfalls verschoben

6.5 Projektraum

Wird vom FB3 zur Verfügung gestellt

6.6 Optimale Zahl von Teilnehmenden

12

Literatur

[BJR97] G. Booch, I. Jacobson, and J. Rumbaugh. *UML Summary (Version 1.0)*. Rational Corporation, Santa Clara, 1997. http://www.rational.com.

[BJR99a] Grady Booch, Ivar Jacobson, and Jim Rumbaugh. *The Unified Modeling Language Reference Manual*. Addison-Wesley, Reading, 1999.

[BJR99b] Grady Booch, Ivar Jacobson, and Jim Rumbaugh. *The Unified Modeling Language User Guide*. Addison-Wesley, Reading, 1999.

[Boo91] G. Booch. *Object-oriented Design with Application*. Benjamin-Cummings, Redwood City, 1991.

[CAB+94] D. Coleman, P. Arnold, S. Bodroff, C. Dollin, H. Gilchrist, F. Hayes, and P. Jeremaes. *Object-Oriented Development: The Fusion Method*. Prentice Hall, Englewood Cliffs, 1994.

[Che76] P.P. Chen. The Entity-Relationship Model – Towards a Unified View of Data. *ACM Trans. on Database Systems*, 1(1):9–36, 1976.

[CY90] P. Coad and E. Yourdon. *Object-Oriented Analysis*. Yourdon Press, Prentice Hall, 1990.

[GJM91] C. Ghezzi, M. Jazayeri, and D. Mandrioli. *Fundamentals of Software Engineering*. Prentice-Hall, Englewood Cliffs (NJ), 1991.

[Gog94] M. Gogolla. *An Extended Entity-Relationship Model — Fundamentals and Pragmatics*. Springer, Berlin, LNCS 767, 1994.

[JCJÖ92] I. Jacobson, M. Christerson, P. Johnsson, and G. Övergaard. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Prentice-Hall, 1992.

[MC97] Chris Marrin and Bruce Campbell. *VRML 2*. Sams Net, Indiana, 1997.

[RBP+91] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. *Object-Oriented Modeling and Design*. Prentice-Hall, Englewood Cliffs (NJ), 1991.

[Rei85] W. Reisig. *Petri Nets: An Introduction*. EATCS Monographs on Theoretical Computer Science. Springer, Berlin, 1985.

[SM92] S. Shlaer and S.J. Mellor. *Object Life Cycles: Modeling the World in States*. Yourdon Press computing series, Prentice-Hall, Englewood Cliffs (NJ), 1992.