
Architektur und Entwicklung von Augmented Reality Anwendungen; AR Frameworks

Ludwig-Maximilians-Universität München
HS Virtual- and Augmented Reality
Branimir Mirtchev

Gliederung

- Einführung
 - Grundlagen und Ansätze
- Beschreibungssprachen
 - ASUR und ASUR++
 - UMLi
- Frameworks allgemein
- AMIRE
- DART
- ARVIKA
- DWARF
- Studierstube

Einführung

- Augmented Reality ist ein neues Forschungsgebiet und befindet sich noch in der Entwicklungsphase
- Die Entwicklung von AR Anwendungen ist eine komplexe und schwierige Aufgabe, die ohne Wiederverwendung von Code viel zu aufwendig wäre
- Aufgrund der schnellen Entwicklung der Technik ist die Notwendigkeit nach Hilfsmittel, Software und Standards sehr groß
- Die meisten Projekte sind aber noch in der Entwicklungsphase, sind experimentell und unvollständig und es fehlt an der nötigen Dokumentation

Grundlagen

- Viele Augmented Reality Anwendungen basieren auf zwei low-level Bibliotheken
 - OpenGL wird zur *3-D-Darstellung von Objekten* benutzt
 - ARToolkit wird zur Markererkennung benutzt und ist gut geeignet für die schnelle Prototypenbildung von AR Anwendungen
- Es gibt Projekte, die diese und andere ähnliche Software benutzen, integrieren und erweitern, um Gesamtlösungen für AR anzubieten
- Viele Projekte bauen ihre eigene AR Algorithmen und Techniken – es gibt keine etablierten Standards und Wiederverwendung ist nicht das Hauptziel

Entwicklungsansätze

- Man kann die Projekte, die sich mit AR beschäftigen, in 2 Typen unterteilen
 - Entwürfe, die ein spezielles Problem lösen und damit wichtige allgemeine Erfahrungen mit AR sammeln
 - Entwürfe, die Modelle aufstellen, wie AR Anwendungen allgemein entwickelt werden und dabei Bibliotheken, Tools, Frameworks und Entwicklungsumgebungen zur Verfügung stellen, die in andere Projekte eingesetzt werden könnten

Beschreibungssprachen

Trotz der wachsenden Bedeutung von AR Technologien gibt es keine ausreichende Werkzeugunterstützung für das Design von AR Systeme. ASUR und UMLi sind zwei Notationen, die benutzt werden können, um designspezifische Eigenschaften von Augmented Reality Anwendungen zu erfassen.

ASUR und ASUR++

■ Allgemein

- Wurden an der Universität Glasgow entwickelt
- ASUR ist eine graphische Notation, die für die Beschreibung, Charakterisierung und Analyse von "mixed" Umgebungen benutzt werden kann
- ASUR ++ ist eine Weiterentwicklung von ASUR
- Beide Notationen ermöglichen die Modellkonstruktion eines AR Systems.
- Modellieren die Schlüsselobjekte einer AR Umgebung, zusammen mit Ihren physischen und logischen Beziehungen

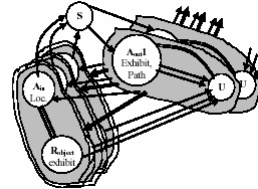
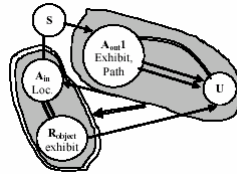
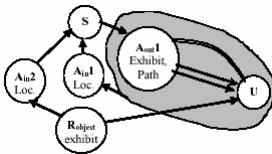
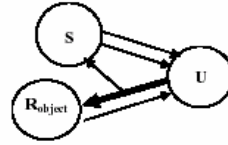
ASUR++

■ Modell

- Eine ASUR++ Beschreibung enthält folgende Entitäten, so genannte Komponenten
 - Computer System (Komponente **S**)
 - Benutzer (User) des Systems (Komponente **U**)
 - Reales Objekt (Komponente **R_{too}** und **R_{object}**)
 - Adapter für Input oder Output (Komponente **A_{in}** und **A_{out}**)
- Es bestehen Beziehungen zwischen den Komponenten:
 - Tausch von Informationen (Daten), dargestellt mit einem einfachen Pfeil (**A->B**)
 - Physische Aktivität, die eine Aktion auslöst (**A=>B**)
 - Dauerhafte physische Beziehung, dargestellt durch doppelte Linie (**A=B**)

ASUR++

- Beispiel: Beschreibung einer AR Anwendung für ein Museum mit Exponaten



21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

9

UMLi

- Das Projekt UMLi wird entwickelt in der [Information Management Group](#) der Universität Manchester seit 1998
- Notation für das Design und Implementierung von grafischen Benutzeroberflächen
- Benutzt eine Kombination von UML and MB-UIDE (Model-based user interface development environments) Techniken
- Bietet eine Integration zwischen User Schnittstellen und unterliegende Anwendungen
- ARGOi ist die UMLi Version von Argo/UML

21.06.2004

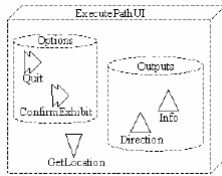
Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

10

UMLi Modell

■ UMLi constructors

- Freie Container
- Container
- Inputters
- Displayers
- Editoren
- ActionInvokers



Frameworks

- Definition nach Pomberger / Blaschek
 - Menge von zusammengehörigen Klassen, die einen abstrakten Entwurf für eine Problemfamilie darstellen
- Direkt aus dem Englischen übersetzt
 - Rahmen, Gerüst, Bezugssystem, System usw.
- Klassifikation:
 - Architektur-getrieben
 - Daten-getrieben
 - Zwei-Schichten-Architektur

AMIRE authoring mixed reality

- Das Hauptziel und Idee von dem AMIRE Projekt ist ein Softwaresystem zu definieren und implementieren, das Experten das leichte Design und Implementierung von Mixed Reality Anwendungen ohne detailliertes Wissen über die grundlegenden Technologien von MR ermöglicht
- Das AMIRE -Projekt ist ein technologisches Projekt, gefördert von dem EU IST (Information Society Technologies) Programm.

AMIRE – authoring mixed reality

- Zielsetzungen sind die Entwicklung von:
 - „best practice“ - Beispiele
 - Spezifikation von Schnittstellen
 - MR Gems
 - MR Komponenten
 - MR Frameworks
 - MR Autorenmetaphern
 - Allgemeine Richtlinien, Empfehlungen und Verfahren fürs Design
 - Autorenwerkzeuge
- Zwei Demonstrationsprogramme werden implementiert für
 - Ö raffinerie
 - Museum

MR Gems

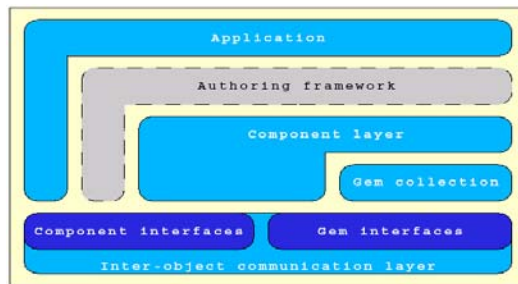
- Sammlung von Techniken und Algorithmen für Programmierer
- Freigeben und Wiederverwendung von Ideen und Tools
- Innovative Lösungen von Programmierproblemen in MR Anwendungen
- z.B. object recognition, tracking, camera control
- Existierende Bibliotheken werden erweitert, um ihre Anwendung in den AMIRE Komponenten zu unterstützen

MR Components

- Gut konstruierte Schnittstellen
 - Benutzen „generic uniform component interface“
 - Bestehend aus geometrischem Modell
 - Domainspezifisch
 - Mehrfach nutzbar und anpassbar
- Ermöglichen auch objektinterne Kommunikation
- Werden in eine Komponentenbibliothek gespeichert
- Kosteneffektive Entwicklung von Anwendungen

MR Framework

- Klebstoff zwischen Gems und Komponenten
- Bietet eine High Level API und ein Interface für die Komponenten
- MR Runtime Framework
- Authoring Framework

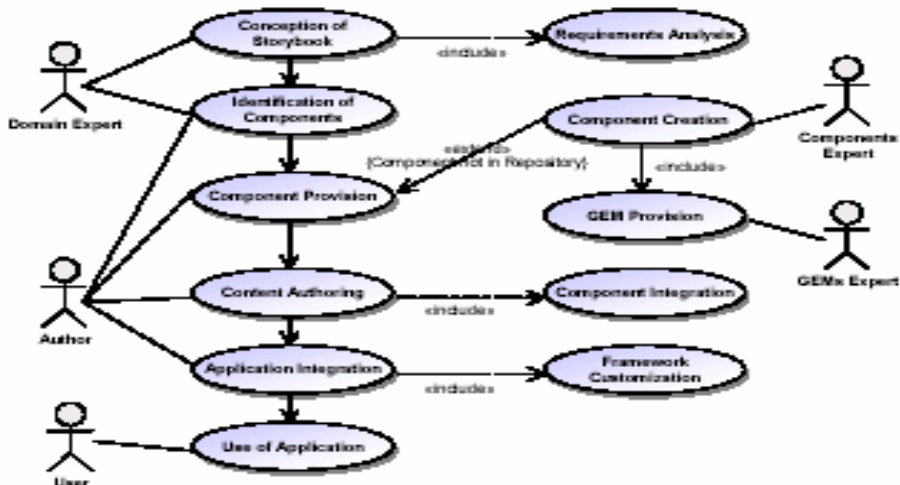


21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

17

AMIRE production process



21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

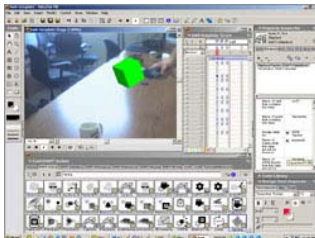
18

DART

- Das Designers Augmented Reality Toolkit besteht aus Werkzeugen, die das schnelle Design und Implementierung von AR Anwendungen ermöglichen
- Erleichtert das komplette Design und den Entwicklungsprozess von dem Anfangskonzept bis zum fertigen Produkt
- Wurde entwickelt als Sammlung von Erweiterungen von Macromedia Director
- Ermöglicht iterative Entwicklung durch Testen in der Entwicklungsphase

DART

- Kann in jedem Domain benutzt werden
- Ist geeignet auch für Leute, die keine Erfahrung mit AR haben
- Die Entwicklung geschieht in der mächtigen Umgebung Macromedia Director



ARVIKA

- Augmented Reality für Entwicklung, Produktion und Service
- Gefördert von Ministerium für Bildung und Forschung
- Ziel ist es, durch Einsatz von AR Technologien bestimmte Arbeitsprozesse in Entwicklung, Produktion und Service für komplexe technische Produkte und Anlagen zu unterstützen.

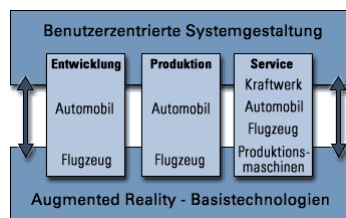
21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

21

ARVIKA

- Projektstruktur
 - Zwei horizontale Themenschwerpunkte
 - Benutzerdefinierte Systemgestaltung
 - Basistechnologien
 - Drei vertikale Themenschwerpunkte
 - Entwicklung
 - Produktion
 - Service



21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

22

ARVIKA

- Anwendungsfelder
 - Automobil- und Flugzeugbau
 - Maschinen- und Anlagenbau
 - Stärkung mittelständischer Unternehmen durch verbesserte Diagnose- und Wartungskompetenz



21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

23

DWARF

- Was ist DWARF: Distributed Wearable Augmented Reality Framework
- Entwickelt am Lehrstuhl für Angewandte Software Entwicklung der TUM
- Die Idee ist eine Lösung zu finden, wo die schnelle Prototypenbildung mit mehreren Entwicklern an verschiedenen Computer möglich ist
- CORBA basiert

21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

24

DWARF

■ Architektur

- Basiert auf zusammenarbeitende verteilte Services
- Diese werden von Servicemanagers kontrolliert
- Jedes Service hat eine XML Beschreibung

■ Implementierung

- Die stationären Computer sind in einem 100 megabit LAN
- Dwarf Services sind realisiert durch separate Prozesse und Threads
- Middleware ist grundsätzlich CORBA

DWARF

■ Vorteile

- Plattform- und Programmiersprachenunabhängigkeit
- Rapide Prototypenbildung
- Benutzung von vorhandener Hardware
- Integration von Hard- und Software anderer Forschungsgruppen
- Remote Überwachung des Systems
 - Tool, genannt DWARF Interactive Visualisation Environment

Studierstube

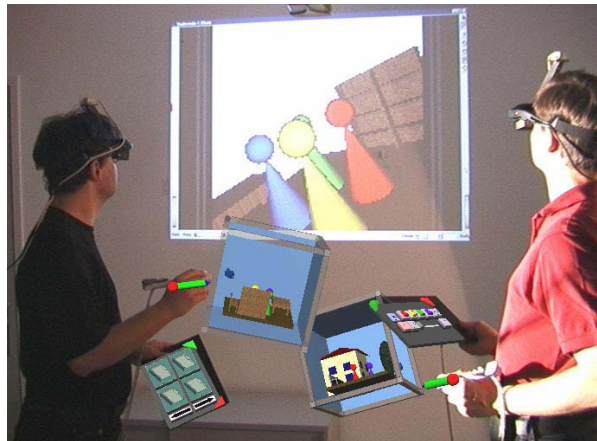
- Umgebung für die Entwicklung von kollaborative AR Anwendungen
- Entwickelt an der TU Wien
- Einfache und leichte Entwicklung von AR Anwendungen
- Das Konzept umfasst
 - Spezifische Hardwareeinstellungen
 - Framework, genannt Studierstube Application Programmer's Interface (StdAPI)
- StdAPI ist realisiert als ein Set von c++ Klassen, aufgebaut auf Open Inventor
- Unterstützt eine große Auswahl von Hardware
 - Input: trackers, artoolkit, desktop hardware
 - Output: HDMS, virtual table, projection setups, stereo usw.

21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

27

Studierstube



21.06.2004

Branimir Mirtchev- Architektur und
Entwicklung von AR-Anwendungen

28

Quellen

- ASUR und ASUR++
 - http://lihs.irit.fr/dubois/Research/Publi/DuboisNG_MobileHCI02.pdf
 - http://www.ksl.stanford.edu/people/pp/papers/Dubois_DSVIS_2002.pdf
- UMLi
 - <http://www.cs.man.ac.uk/img/umli/>
- Frameworks
 - http://www.medien.informatik.uni-muenchen.de/fileadmin/mimuc/mt_ws0304/material/mt8e.pdf
- AMIRE
 - <http://www.amire.net/>
- DART
 - <http://www.gvu.gatech.edu/dart/>
- ARVIKA
 - <http://www.arvika.de/>
- DWARF
 - <http://www.bruegge.in.tum.de/DWARF/WebHome>
 - <http://www.augmentedreality.de>
- Studierstube
 - <http://www.studierstube.org>