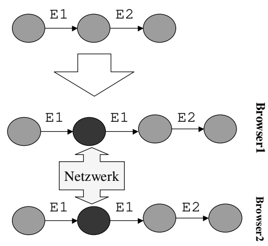
VRML Erweiterungen für verteilte Welten

- VSPLUS
- LivingWorlds **(schauen wir uns erst später an)**
  - Sony's Implementierung von LW verwendet VSPLUS

VSPLUS

- Wie macht man aus Einbenutzerwelten auf einfache Weise Mehrbenutzerwelten?
- Netnodes
  - different timestamps !



VSPLUS: Consistency Manager

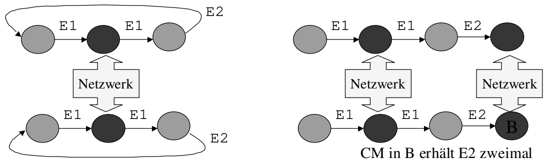
- Einer pro Netnode
  - für Avatar im Browser des Avatarbesitzers
  - für Bots im Anwendungsserver
- Konflikt = CM empfängt Event von mehreren Teilnehmern gleichzeitig
- Durchschnitt der Wert oder andere Auflösungsoperation
- Abstimmen (voting)
- oder Konfliktvermeidung durch Zugriffskontrolle über Semaphore

VSPLUS: Consistency Manager

- Late Comers: Bevor ein neue hinzugekommener Browser ein Event auslösen darf, fordert er beim CM den Wert des letzten Events an.
- **Single-User:** Netnodes lassen Events einfach durch, falls die Netzwerkverbindung unterbrochen wurde.
- **Eventströme:** Kontinuierliche Events werden gesampled, d.h. nur nach festgelegten Intervallen übertragen.

VSPLUS: Kausalität

- Aufgabe des CM ist es auch, die Kausalität, d.h. welcher Event welchen auslöst, zu erhalten.
- Problem: Event-Loops und Message-Expansion
- Lösung: Beide Fälle sind verboten! (Testtool)



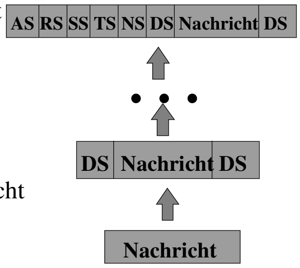
CM in B erhält E2 zweimal

Protokolle für verteilte Welten

- Protokoll: In verteilten Systemen kommunizieren Anwendungen durch Austausch von Nachrichten. Ein Protokoll legt Form, Inhalt und Abfolge soclher Nachrichten fest.
- Zuerst Grundlagen: IP,TCP,UDP,Multicast
- DIS
- DWTP
- VRTP

Schichtenmodell

- phyikalische Schicht
- Datenschicht
- Netzwerkschicht
- Transportschicht
- Sitzungsschicht
- Repräsentationsschicht
- Anwendungsschicht



Internet Protocol IP

- Protokoll der Netzwerkschicht
- 32 Bit-Adressen:
  - 11000001:01100001:11111011:00000100
  - 193.97.251.4
  - www.addison-wesley.de
- IPv6: 128 Bit-Adressen

Transmission Control Protocol TCP

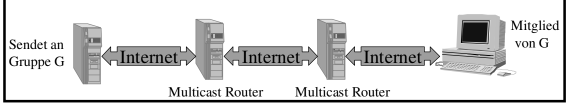
- Protokoll der Transportschicht
- Ablauf:
  - Nachricht ====> Pakete
  - Sequenznummern
  - fordert verlorene oder defekte Pakete erneut an
  - Pakete ====> Nachricht
- Ports, Dienste, Server, Superserver
- verläßlich

User Datagram Protocol UDP

- Protokoll der Transportschicht
- Nachricht gleich Paket
- Best Effort:
  - Pakete können verloren gehen oder in der falschen Reihenfolge ankommen
- schnell, aber unzuverlässig

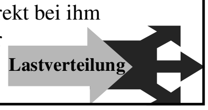
Multicast IP (MBone)

- Virtuelles Teilnetz im Internet (Info: www.mbone.com)
- nur UDP, daher unzuverlässig (derzeit sind einge zuverlässige Protokolle vorgeschlagen, aber meist anwendungsabhängig)
- Multicast Gruppen
  - Rechner können Gruppen beitreten, dann erhalten sie die Nachrichten für diese Gruppen.
- Multicast Router



Multicast verhindert Bottleneck

- Ohne Multicast
  - sende N Nachrichten an N Mitglieder
- Mit Multicast
  - sende eine Nachricht an nächsten Multicast Router
  - sende k << N Nachrichten an die k nächsten Multicast Router
  - Der letzte Router vor einem Gruppenmitglied schickt dann j<<N Nachrichten an die direkt bei ihm eingetragenen Gruppenmitglieder



Architekturen für verteilte Welten

- Client/Server: häufigste, da einfachste Architektur.
  - Server=Bottleneck (alle Kommunikation über Server)
  - Single-Point-of-Failure: was tun, wenn Server (kurzzeitig) ausfällt?
  - Problem, wenn Client Daten cached.
- Peer-to-Peer: Jeder Client hat Kopie der Daten. Der aktuelle Zustand ist auf alle Clienten verteilt.
- Hybrid: Vereinigung beider Modelle
  - Server für Startup und Persistenz
  - Peer-to-Peer für bessere Performanz

Konsistenz

- Strikt: alle Änderungen müssen an alle Clienten geschickt und Konflikte vermieden oder aufgelöst werden (--> TCP)
- Best Effort: schicke Änderungen an alle Clienten, so gut es geht. (--> UDP)
- **Reduktion des Aufwands** durch
  - Consistency Group (Teilmengen aller Clients)
  - Abschwächung (Relaxation) der Konsistenz, z.B.
    - zeitlich (zeitliche Abfolge muß nicht überall gleich sein)
    - datenbezogen (Mittelwertbildung, erlaube Abweichungen bis zu einem gewissen Grad)

Protokolle für verteilte, virtuelle Welten

Es gibt noch nicht **den** Standard, aber man kann von den vorgeschlagenen Standards viel lernen.

Distributed Interactive Simulation DIS

- Vorgänger: SIMNET 1985
  - Computersimulation eines Schlachtfelds mit Panzern, Flugzeugen, Geschossen, …
  - Berechnung der Simulation verteilt auf viele Rechner im Netz
  - interaktiv = Menschen kontrollieren einige der Objekte
  - Einsatz: Ausbildung von Panzerfahren bishin zu Kommandeuren
- DIS seit 1989, IEEE Standard seit 1993
- Nachfolger: HLA (high-level architecture, ähnlich CORBA)

Distributed Interactive Simulation DIS

- Nachrichtenformat: protocol data unit (PDU)
  - 27 verschiedene, etwa für Detonation, Feuern, Markieren eines Minenfeldes
  - Wichtigste PDU: Entity State PDU
    - Position, Ausrichtung, Geschwindigkeit, …
- stateless
- hearbeat
- peer-to-peer
  - Jede Zustandsänderung wird an alle geschickt

Header
Entity ID
…
EntityType
…
Linear Velocity
Location
Orientation
Appearance
Dead Reckoning Parameters
...

Schätzung:
100.000 Teilnehmer erfordern 375 Mbit/s
(siehe Dissertaton von M. Macedonia, 1995)

nicht skalierbar

Distributed Worlds Transfer and Communication Protocol DWTP

- Entwickelt von Wolfgang Broll, GMD
- Anwendungsschichtsprotokoll für große, verteilte, virtuelle Welten (large-scale)
- anwendungsunabhängig
- setzt nicht auf einem bestimmten Netzwerkprotokoll auf
- skalierbar (u.a. durch Multicast)
- heterogen, d.h. überträgt verschiedene Datentypen: Events, Nachrichten, Dateien, Ströme

Skalierbarkeit in DWTP

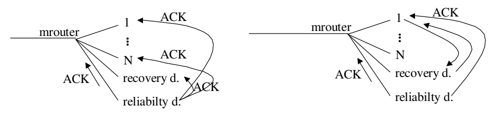
- 2 Rollen: Teilnehmer und Dämonen
  - Ein Dämon kann gleichzeitig für mehrere Welten zuständig sein
  - Arten von Dämonen:
    - world (überträgt Welt an neue Teilnehmer)
    - reliability (entdeckt Übertragungsfehler)
    - recovery (bietet Unicast-Verbindung zum Übertragen verlorener Pakets)
    - unicast (erlaubt Verbindung mit nicht multicast-fähigen Teilnehmern)
- Skalierbarkeit: Dämonen können auf verschiedenen Rechnern laufen!

Fehlerbehandlung in DWTP

- Traditionell:

Server hat Übertragungsproblem und erhält dazu noch N NACKs

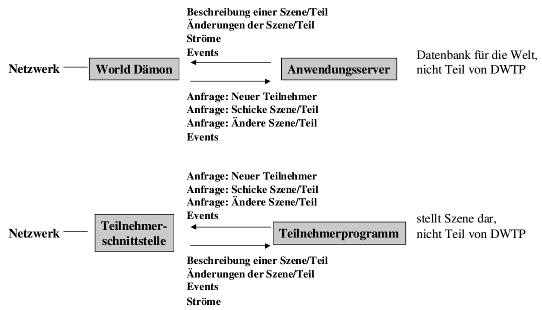
- DWTP



ACK enthält Paketnummer!
Teilnehmer erkennt Problem, wenn er ACK für ein Paket erhält, aber nicht das Paket.

Falls Recovery Dämon das Paket auch nicht erhalten hat, muß die gesamte, aktuelle Welt von einem World Dämon neu geladen werden.

DWTP: Anfragen und Antworten



Virtual Reality Transfer Protocol VRTP

- Status: Laufendes Projekt, keine fertige Implementierung oder Spezifikation
- **Rollen:** (ein Rechner kann mehrere Rollen erfüllen)
  - **Client:** Anzeigen der Welt (z.B. VRML-Browser)
  - **Server:** Übertragen der Welt an andere (z.B. über http)
- Wichtige Eigenschaften
  - **Peer-to-Peer:** Multicast Gruppen, Streaming
  - **Monitoring:** Übertragungsprobleme feststellen und lösen
- **VRTP kombiniert existierende Protokolle und Techniken**
  - http, DIS, multicast streaming, Java Agenten, Netzwerkmonitoring

VRTP: heterogene Daten

- Leight-Weight Interactions:
  - Events, Zustands- und Kontrollinformationen, PDUs, unzuverlässig
- Network Pointer:
  - Referenzen auf Objekte in einem globalen Adreßraum, z.B. URLs
- **Heavy-Weight Objects:**
  - Dateien, zuverlässig
- **Real-Time Streams:**
  - Live Video, Audio, Verhaltensströme, über Multicast, unzuverlässig