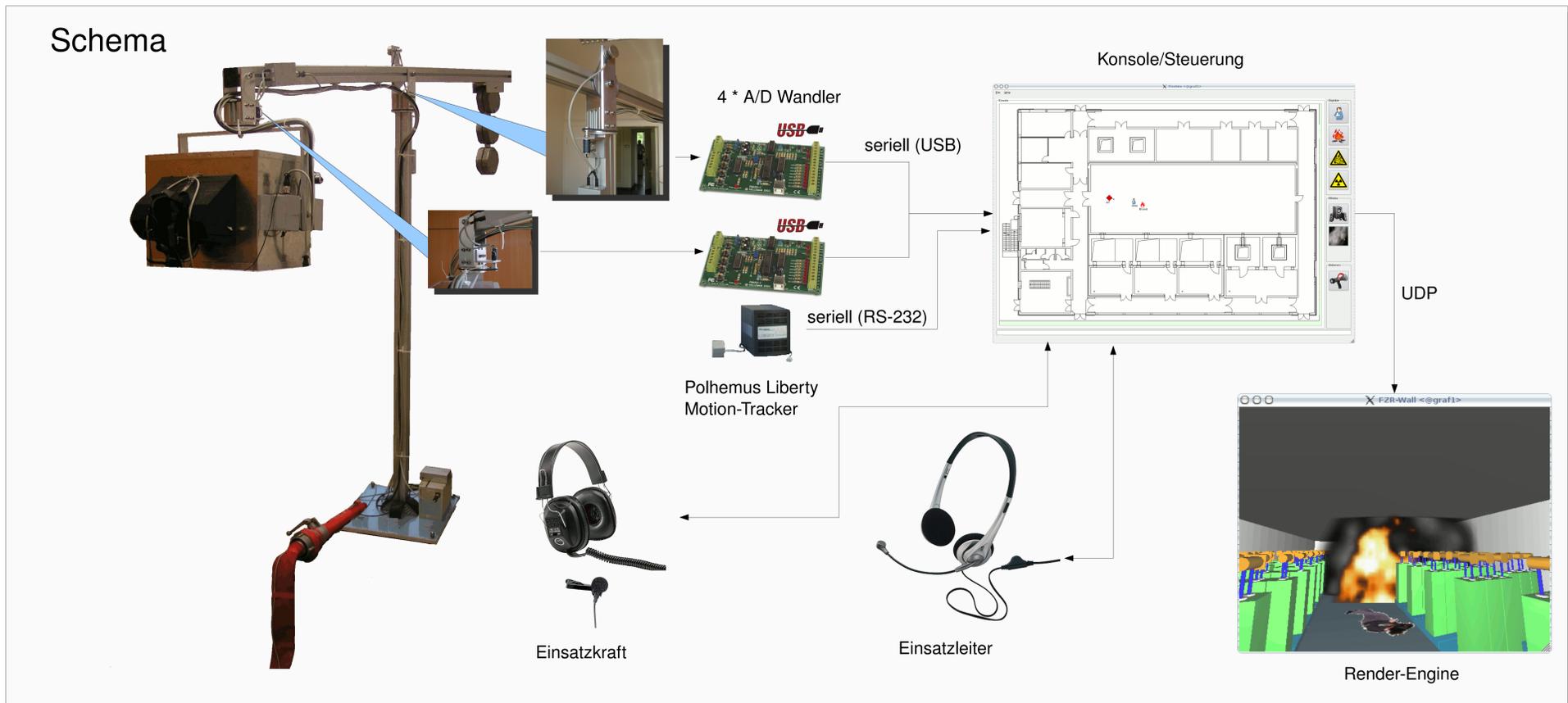


FireSim – im Detail

N. Schmeißer



Hardware

- 2 * AMD Quad-Core Opteron 1.8 GHz
- Grafik: nVidia Quadro FX 3700
- A/D Wandler und Digitaleingabe: 2 * Velleman K8055 USB
 - vorgeschalteter Verstärker (Eigenbau FZD)
- Polhemus Liberty
 - Elektromagnetischer "motion tracker" (6 Freiheitsgrade)
 - Teststellung VRLogic

Software

- Betriebssystem: Linux
- Simulation/Konsole: Eigenentwicklung C/C++, Qt
- Rendering: Eigenentwicklung OpenGL
 - VRML-Parser
 - Erweiterung um Sonderdarstellung (Vernebelung, Flammen, ...)
 - monoskopisch und stereoskopisch (aktiv und passiv)



Positionsbestimmung in der virtuellen Welt

- Beobachterreferenzpunkt O
- Messung der Achsstellungen (w_0, w_1, w_2, w_3), Erfassung der Schrittbewegung (n_{steps})
⇒ Beobachterposition P, Blickrichtung ($w_0 + w_1, w_2, w_3$)

$$Q_{t+\Delta t} = Q_t + n_{steps} \cdot \Delta s \cdot \begin{pmatrix} \cos(w_0 + w_1) \\ \sin(w_0 + w_1) \\ 0 \end{pmatrix}, P_{t+\Delta t} = Q_{t+\Delta t} + \begin{pmatrix} \cos(w_1) \cdot \cos(w_2) \cdot l_B \\ \sin(w_1) \cdot \cos(w_2) \cdot l_B \\ \sin(w_2) \cdot l_B \end{pmatrix}$$

- Erfassung der Lagedaten des Strahlrohrs
 - Polhemus Liberty liefert aktuelle Lageparameter in Quaternionendarstellung → Transformationsmatrix M_R
 - sei $P_L(\alpha, \lambda)$ die Wurfparabel (α - Auswurfwinkel) des Löschmittels, dann ergibt sich die Trajektorie $T_L(\alpha, \lambda)$ aus

$$t(\lambda) = M_R \cdot \begin{bmatrix} e_x^T \\ e_y^T \\ e_z^T \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot P_L(\alpha, \lambda); e_x - \text{Einheitsvektor X-Achse}, e_z - \text{Einheitsvektor Z-Achse}$$

