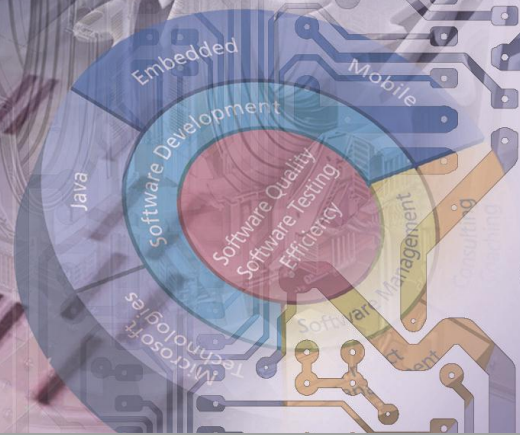


Teile und Herrsche: Multi-Prozess Embedded Software

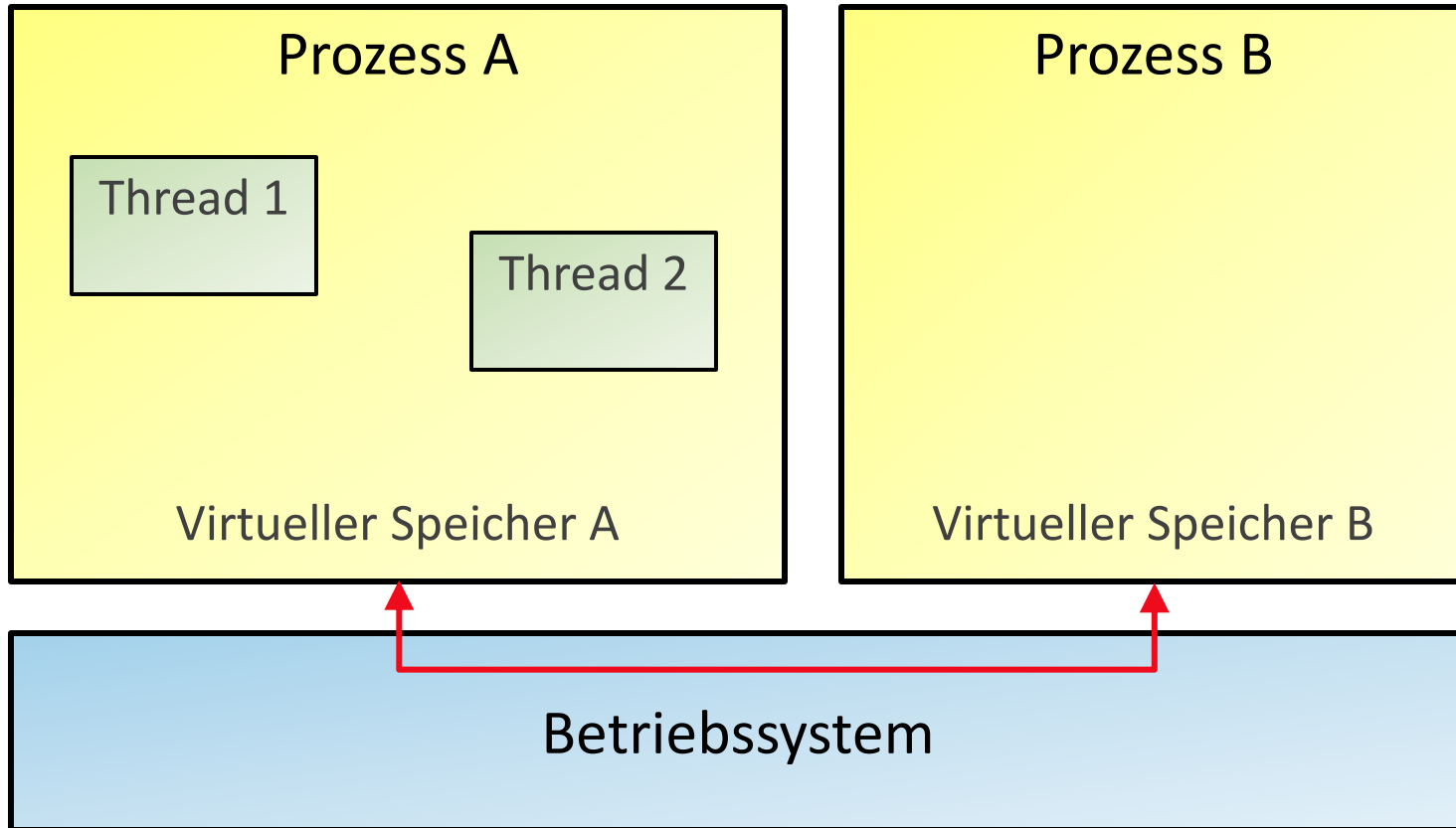
bbv Embedded Event, 27. Oktober 2011
Franco Chiappori



Wenn es eng wird...



Autonome Prozesse



Aufgabe: Erweitere Steuerung mit GUI



Lohnt sich ein Multi-Prozess System für mich?

Technische Aspekte

Unterstützt durch Betriebssystem

Speicher und Prozessorressourcen vorhanden

Lose Kopplung zwischen Komponenten

Sollen Komponenten verteilt laufen

Sind unterschiedliche Sprachen / Architekturen im Einsatz

Lohnt sich ein Multi-Prozess System für mich?

Kommerzielle Aspekte

- Komponenten einzeln lizenzieren
- Unterschiedliche Konfigurationen
- Unterschiedliche Release-Zyklen
- Validierung / Verifizierung wichtig

Organisatorische Aspekte

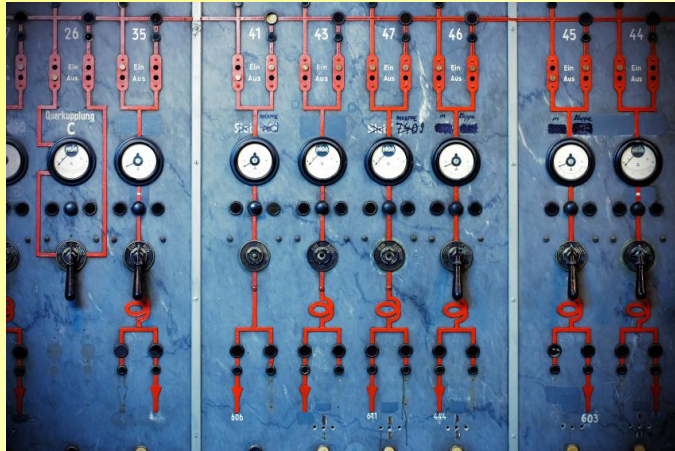
- Grosses Projekt
- Verteiltes Team
- Laufzeit für Build / Tests

Technische Aspekte

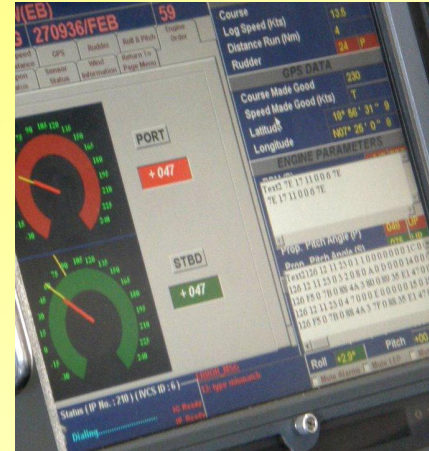
- Unterstützt durch Betriebssystem
- Speicher und Prozessorressourcen vorhanden
- Lose Kopplung zwischen Komponenten
- Sollen Komponenten verteilt laufen
- Sind unterschiedliche Sprachen / Architekturen im Einsatz

Entscheid: Steuerung und GUI als einzelne Prozesse

Prozess A: Steuerung



Prozess B: GUI



Betriebssystem



Wer ist **Client**, wer ist **Server**

Single Client oder **Multi** Client

Datenmenge der Kommunikation

Reaktionsgeschwindigkeit

Laufen Prozesse **verteilt**

Peer-to-Peer oder Bus **Architektur**

Welche **IPC Mechanismen** verwenden

Welche **Protokolle** verwenden



Populäre IPC Mechanismen

Higher-Level IPC Mechanism

D-Bus

Message Bus

SOAP

<Many Others>

Operating System

Event Based

Signals

Semaphores

Message Queues

Stream Based

Sockets

Pipes

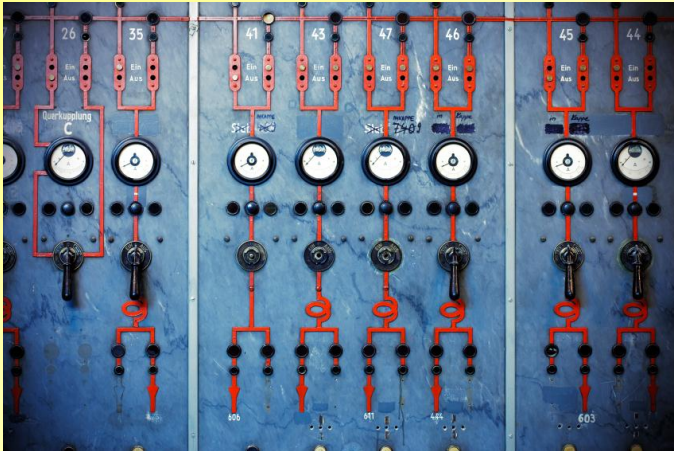
Random Access

Shared Memory

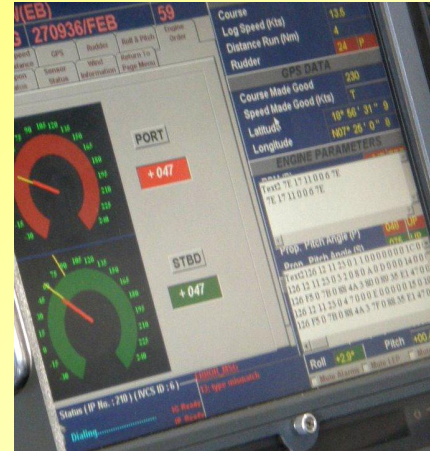
Files

Entscheidung: XML über TCP, Binary über UDP

Prozess A: Steuerung



Prozess B: GUI



XML über TCP für Kommandierung

Binary über UDP für Istwerte

Externes GUI

Wichtige Punkte bei der Umsetzung

- Interface klar spezifizieren
 - IPC Mechanismen, XML Schema, Kommandos, Byteorder, etc.
- Start/Ende der Kommunikation regeln
- Abbruch der Kommunikation behandeln
- Versionen-Mismatch vorbeugen
- Sicherheitsaspekte überdenken
 - Authentifizierung, Verschlüsselung, etc.
- Einsatz von IDL / Generatoren prüfen
- Evtl. gemeinsamen Code in Bibliothek auslagern

Modularität als Hauptgewinn



Teile und herrsche um...



Grosse Projekte **in den Griff** zu bekommen

Komponenten **verteilen** zu können

Komponenten **losgelöst** zu entwickeln und testen

Übersicht zu bewahren

Danke!

Franco Chiappori

info@bbv.ch

www.bbv.ch

Bilder

- Bundesarchiv, Bild 102-08642, Foto: o.Ang., November 1929
- Control panel switches, iStockphoto
- IVCS aboard the INS Shivalik, Ajai Shukla
- Niccolo Machiavelli, Public Domain
- Diverse Clip Arts, Microsoft