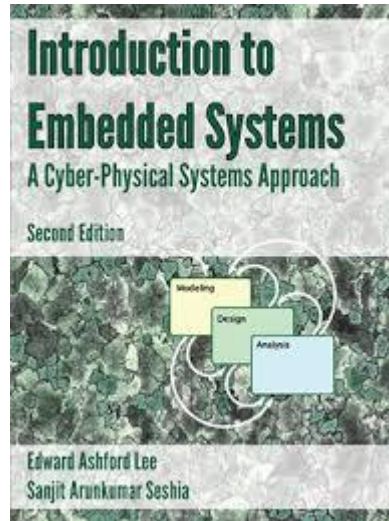


Kiberfizikai rendszerek

A “fizikai” vonatkozásokról ...
utolsó folytatás folytatása

2015. december 1.



Edward Ashford Lee, Sanjit Arunkumar Seshia

Department of Electrical Engineering and Computer Sciences (EECS) at UC Berkeley.

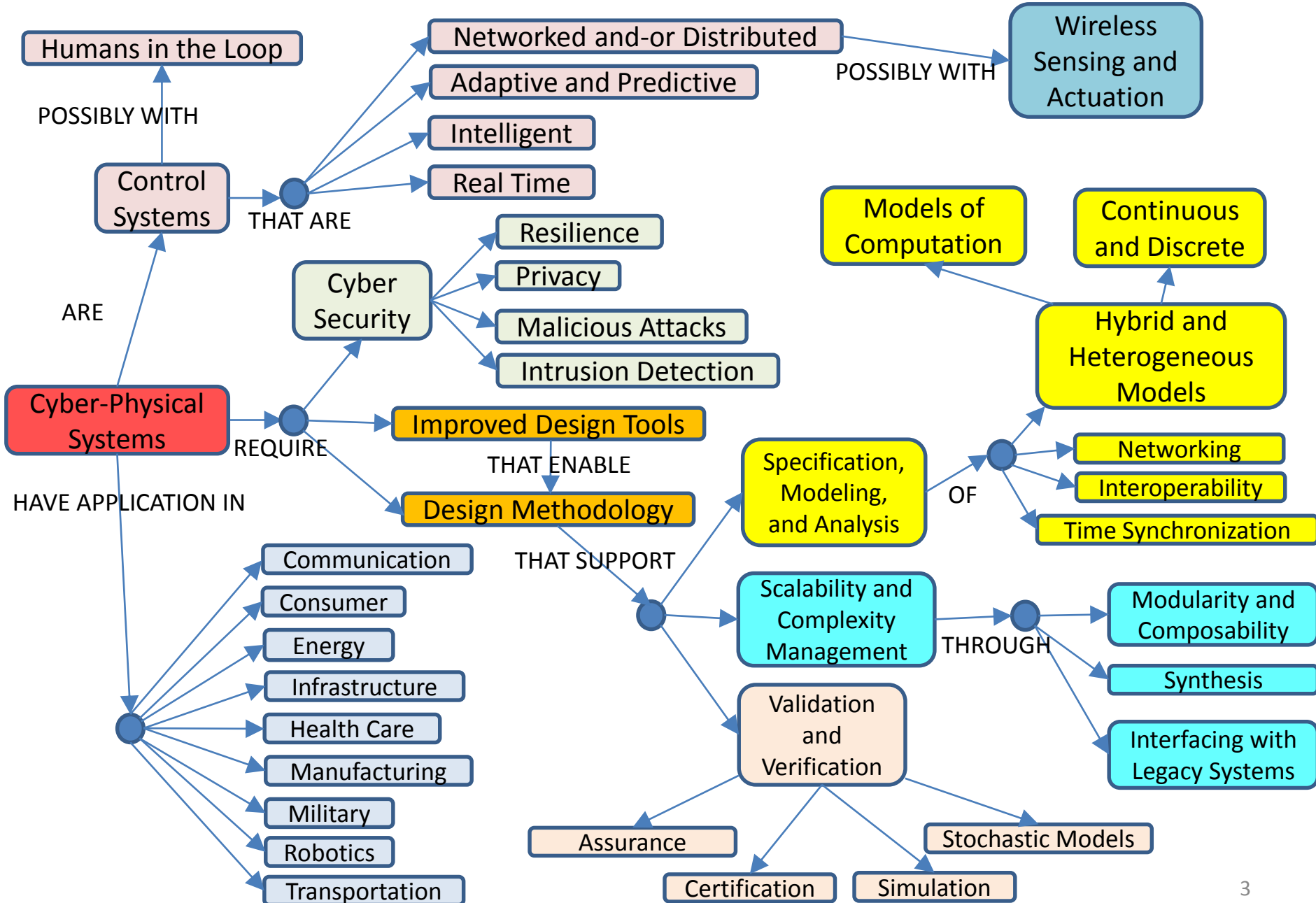
Introduction to Embedded Systems A Cyber-Physical Systems Approach

Second Edition, LeeSeshia.org, 2015.

I Modeling Dynamic Behaviors II Design of Embedded Systems III Analysis and Verification

Cyber-Physical Systems – a Concept Map

<http://CyberPhysicalSystems.org>



A konkurens szoftverek problémái

Edward A. Lee: A fő probléma az, hogy a konkurens szoftvereket dominálják a **szálak**.

→ **A szálak:** szekvenciális számítások megosztott memóriahasználattal.

→ **Az interruptok:** a hardver által indított **szálak**.

A **szálak** közötti érthetetlen kölcsönhatások számos problémát okoznak:
Holtpont, prioritás inverzió, ütemezési anomáliák, az időzítések változékonysága, nemdeterminisztikus viselkedés, buffer túlcsordulás, rendszer összeomlás ...

Állítja: A **szálakkal** megírt nemtriviális szoftverek az emberek számára **érthetetlenek**, és triviális esetektől eltekintve **képtelenek jósolható és megismételhető viselkedésre**.

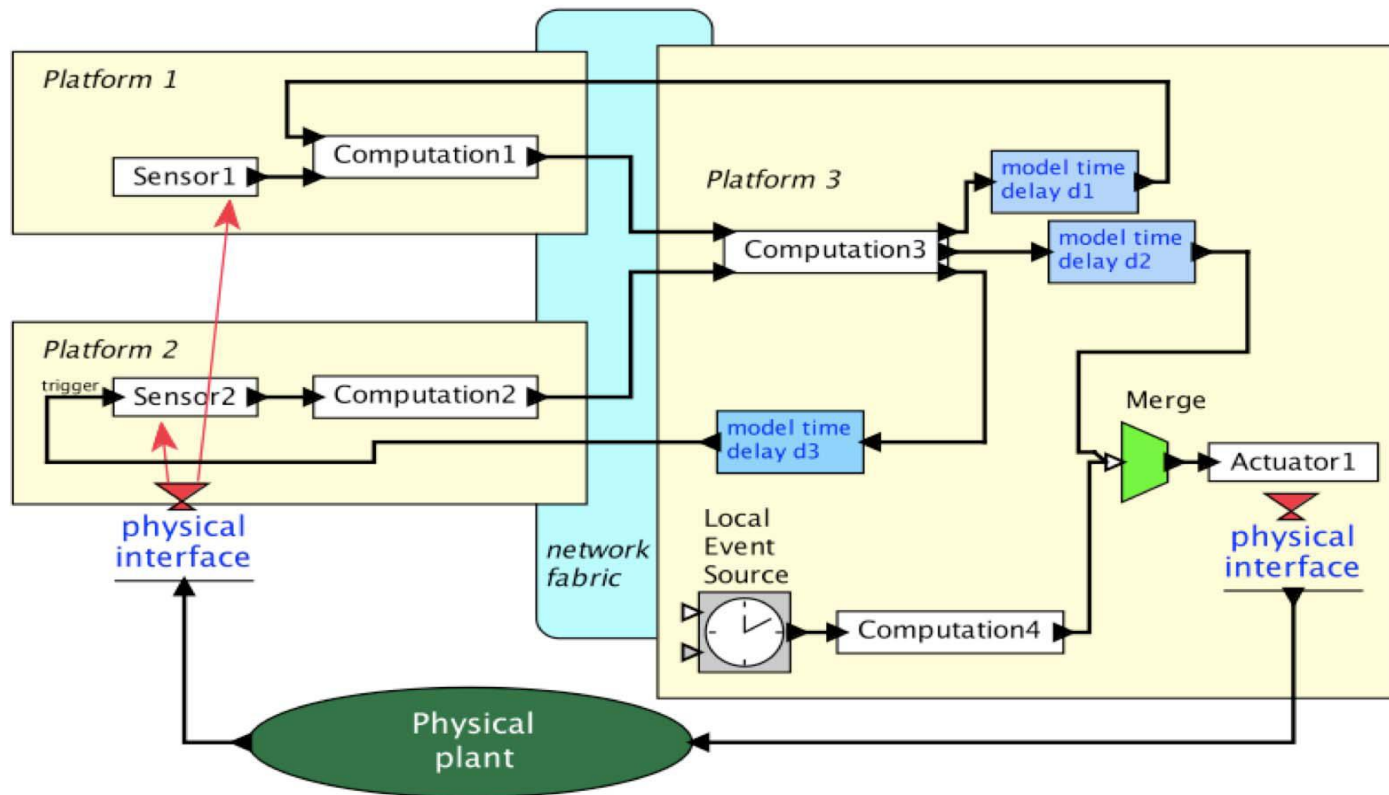
A **szálak** felől nézve a világegyetem egész állapota megváltozhat két atomikus művelet között: rosszul meghatározott elgondolás. *Még szerencse, hogy a fizikai világ nem így működik!*

A megosztott memóriát használó konkurens programok azért érthetetlenek, mert a fizikai világ konkurenciája nem így működik.

Jelenleg az a helyzet, hogy olyan absztrakciók alapján építünk beágyazott alkalmazásokat, amelyekben **az idő irreleváns**, és olyan konkurencia modelleket használunk, amelyek érthetetlenek.

A konkurens szoftverek problémái

A kiber oldal „fizikalizálása”: *a szoftver és hálózati komponensek felruházása olyan absztrakciókkal és interfészekkel, amelyek reprezentálják időbeni dinamikájukat.*



A fizikai oldal „kiberizálása”: *A fizikai alrendszerek felruházása kiber-szerű absztrakciókkal és interfészekkel.*

A konkurens szoftverek problémái

Edward A. Lee: → Legyen **a viselkedés jósolható**, ne csak átlagos viselkedésről beszéljünk.
→ Legyen az **időbeni viselkedés** is annyira fontos, mint a logikai funkció.

Javaslat: **Precision-Timed (PRET) Machines**

- Utasításkészletek időzítéssel kiegészítve
- Jósolható memória menedzsment
- Programozási nyelvek időzítéssel kiegészítve
- Jósolható konkurens viselkedés
- Precíziós hálózatok
- Komponálható időzített komponensek
- Memória hierarchia

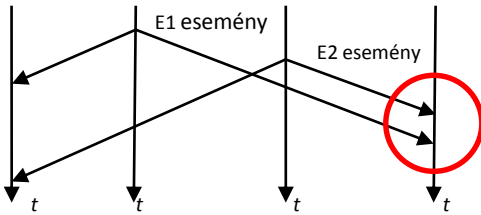
Objektum-orientált
vs.
aktor-orientált

Példák aktor-orientált rendszerekre

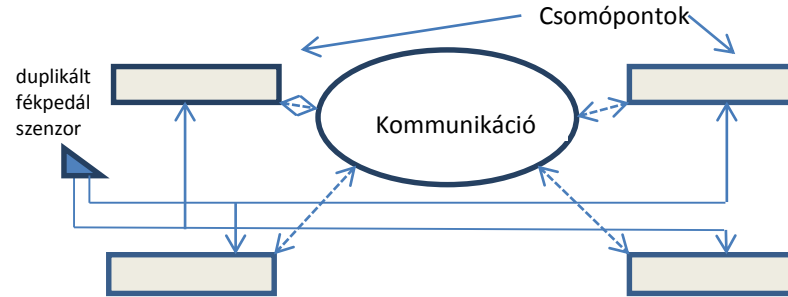
UML 2 and **SysML** (activity diagrams)
ASCET (time periods, interrupts, priorities, preemption,)
Autosar (software components w/ sender/receiver interfaces)
Simulink (continuous time, The MathWorks)
LabVIEW (structured dataflow, National Instruments)
SCADE (synchronous, based on Lustre and Esterel)
CORBA event service (distributed push-pull)
ROOM and **UML-2** (dataflow, Rational, IBM)
VHDL, **Verilog** (discrete events, Cadence, Synopsys, ...)
Modelica (continuous time, constraint-based, Linkoping)
OPNET (discrete events, Opnet Technologies)
SDL (process networks)
Occam (rendezvous)
SPW (synchronous dataflow, Cadence, CoWare)

További valós-idejű vonatkozások

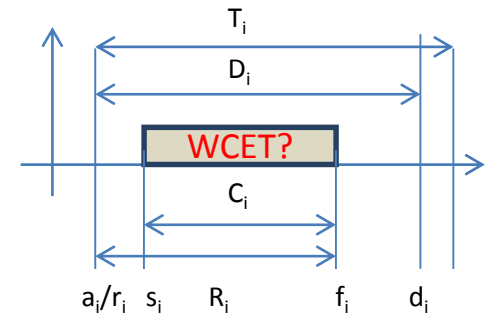
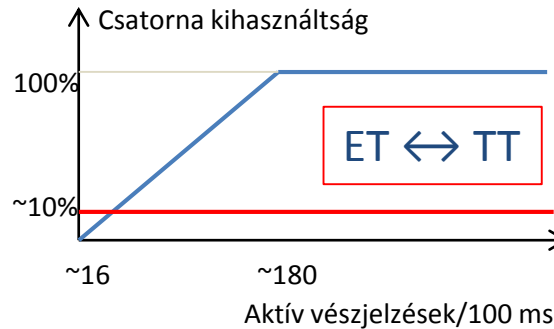
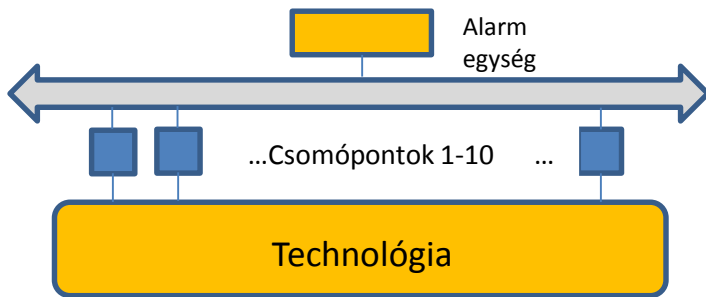
P kliens A server B server Q kliens



Relativisztikus hatás

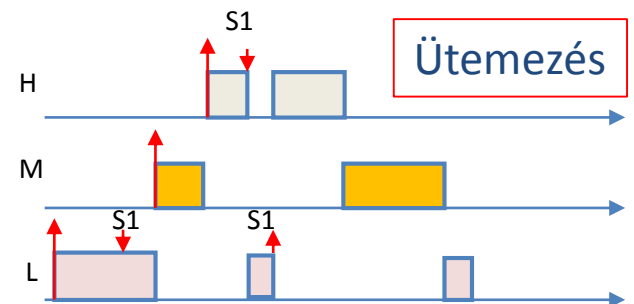


Megegyezés
protokollok

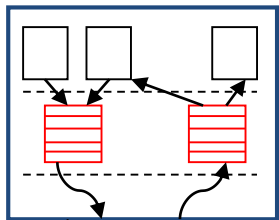


$$R_i^{n+1} = C_i + I_i = C_i + \sum_{\forall k \in hp_i} \left\lceil \frac{R_i^n}{T_k} \right\rceil C_k$$

Ütemezés

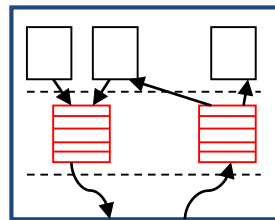


i-edik HOST



Transport Protocol Handler
Network-Access-Control Handler

k-adik HOST



TPH
NACH

Kommunikációs hálózat

Kommunikáció

Ami elmaradt ...

A. Modellezési és szimulációs eszközök:

1. Ptolemy II is an open-source software framework supporting **experimentation with actor-oriented design**. Actors are software components that execute concurrently and communicate through messages sent via interconnected ports. **A model is a hierarchical interconnection of actors**. In Ptolemy II, the semantics of a model is not determined by the framework, but rather by a software component in the model called a director, which implements a model of computation.

2. Modelica is an **object-oriented**, declarative, multi-domain **modeling language** for **component-oriented modeling** of complex systems, e.g., systems containing mechanical, electrical, electronic, hydraulic, thermal, control, electric power or process-oriented subcomponents.

B. Visszacsatolt CPS rendszerek stabilitáselméleti megközelítése

C. És még nagyon-nagyon sok minden!