Network-like System Relations and Their Meaning for IT-System Architecture

Johannes Reich, johannes.reich@sophoscape.de

2013-12-02

To be semantic or not to be?

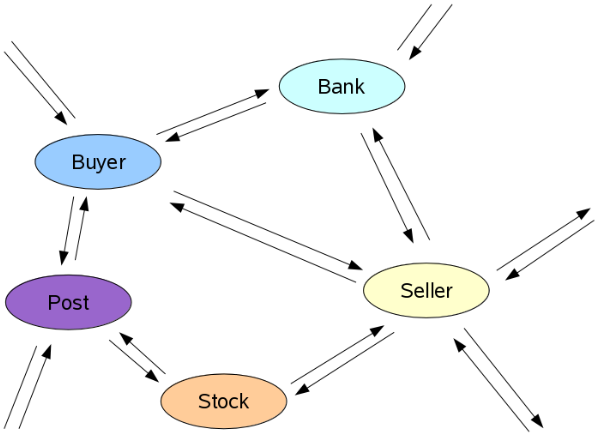
Claude E. Shannon (1948): A Mathematical Theory of Communication

> *"The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. Frequently the messages have meaning; that is they refer to or are correlated according to some system with certain physical or conceptual entities.* **These semantic aspects of communication are irrelevant to the engineering problem***."*

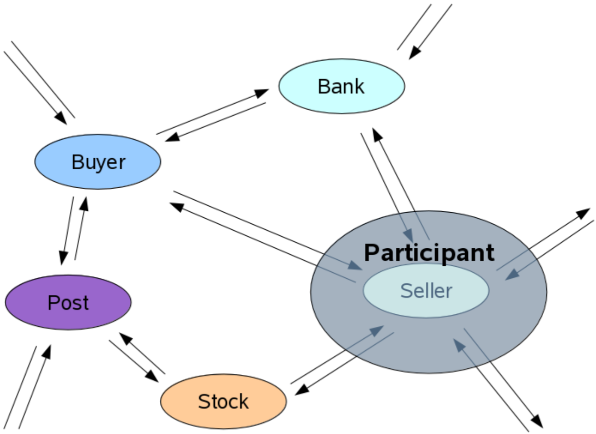
Frege Principle of Semantics

Two components are semantically equivalent if they can be exchanged.

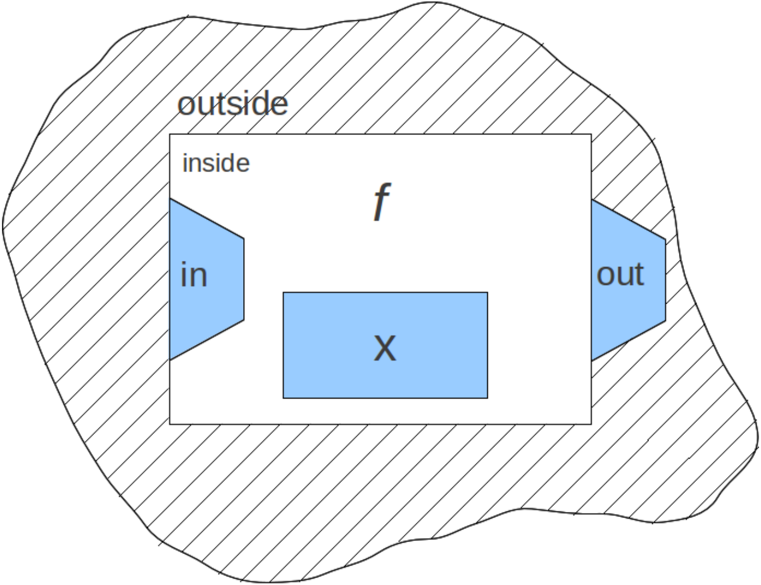
We are living in an open world of network-like relations between systems



We are living in an open world of network-like relations between systems: Focus on processes



Systems



Two Simple Example Systems

A simple multiplier

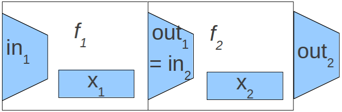
No internal state
Input state: $x, y \in \{-2^{31} \ldots 2^{31} - 1\}$
Output state: $z \in \{-2^{31} \ldots 2^{31} - 1\}$
System function: $z' = f(x, y) = x * y$ // watch for overflow!

A simple counter

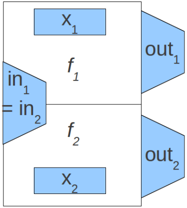
No input and no internal state
Output state: $z \in \{0 \ldots 2^{32} - 1\}$,
System function: $z' = f(z) = z + 1$, // watch for overflow!

System Composition/Super System Formation

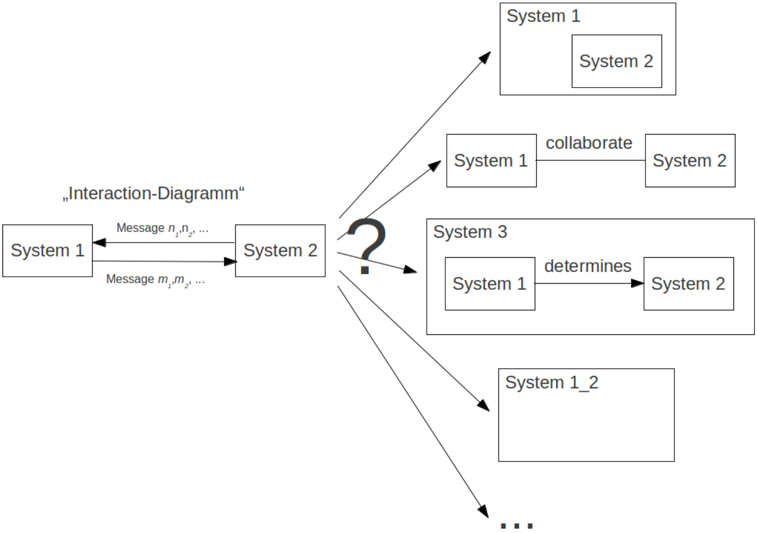
Sequential Composition ($\mathcal{S}_2 \circ \mathcal{S}_1$)



Parallel composition ($\mathcal{S}_2 || \mathcal{S}_1$)



Richer Interaction Semantics



Recursive System Relations

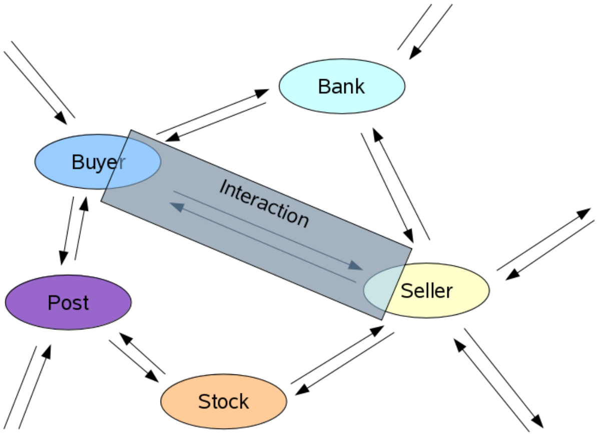
System $\mathcal{U}_1$

```
int fac1(int i) {
  if (i==0)
    return 1;
  else
    return i*fac2(i-1);
}
```

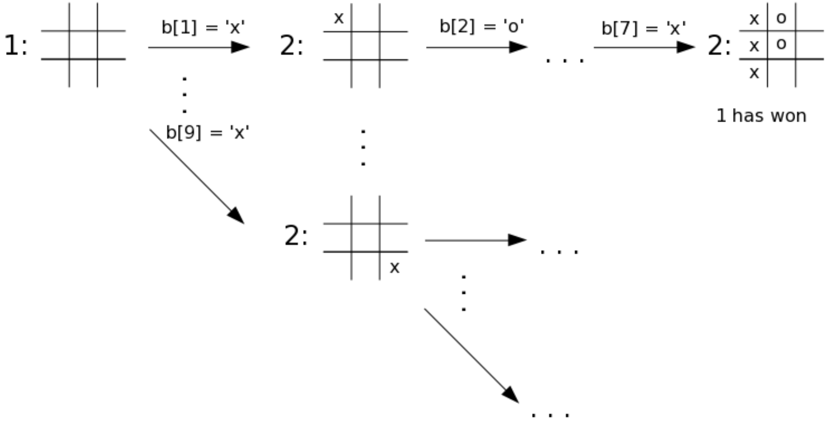
System $\mathcal{U}_2$

```
int fac2(int i) {
  if (i==0)
    return 1;
  else
    return i*fac1(i-1);
}
```

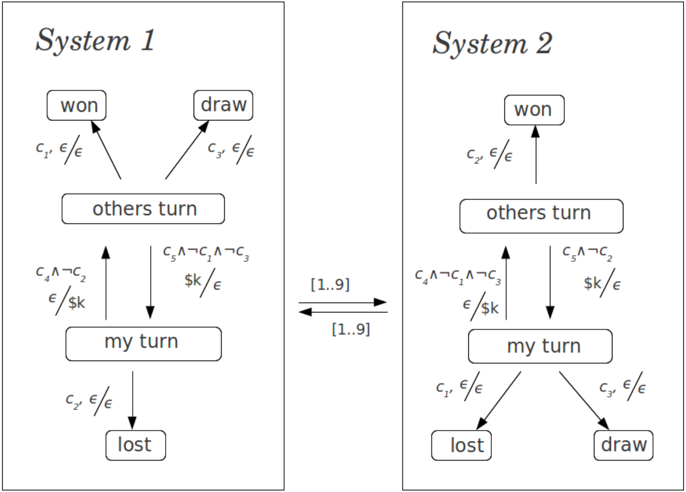
We are living in an open world of network-like relations between systems: Focus on interactions



Tic tac toe as an extensive form game



Tic tac toe as an interaction


System 1
won
draw
c_1, ϵ/ϵ
c_3, ϵ/ϵ
others turn
$c_4 \wedge \neg c_2$
$\epsilon/\$k$
$c_5 \wedge \neg c_1 \wedge \neg c_3$
$\$k/\epsilon$
my turn
c_2, ϵ/ϵ
lost
[1..9]
[1..9]
System 2
won
c_2, ϵ/ϵ
others turn
$c_4 \wedge \neg c_1 \wedge \neg c_3$
$\epsilon/\$k$
$c_5 \wedge \neg c_2$
$\$k/\epsilon$
my turn
c_1, ϵ/ϵ
c_3, ϵ/ϵ
lost
draw

c_1 if player 1 wins.
c_2 if player 2 wins.
c_3 if it is a draw.
c_4 if the k-th position is empty.
c_5 if the '$k'-th position is empty.

Two systems playing tic tac toe. System 1 makes the initial move.

Games and Protocols/Processes

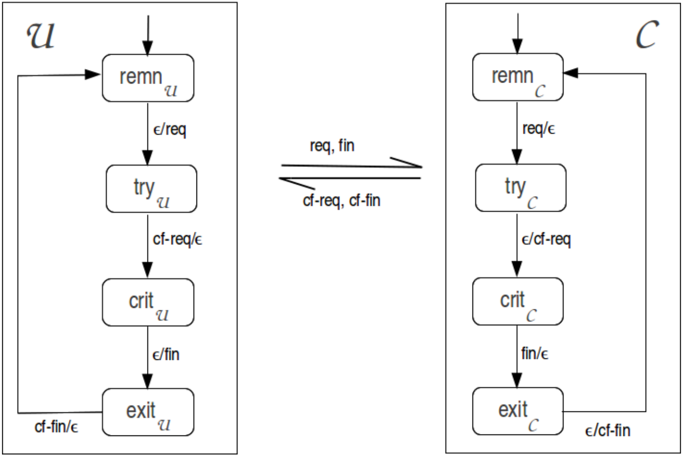
Relation between Games and Protocols

Protocols + Decisions = Games - Payoff

Consequences

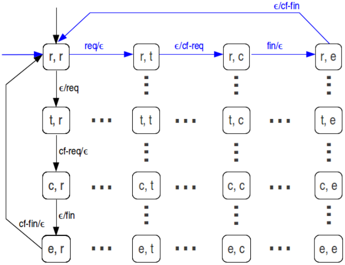
- Game theory and protocol/process theory should use the same interaction model!
- Processes [in game-like interactions] interact via shared states (Shannon-channels).
- Prozesses [in game-like interactions] interact statefully.

Example: the Protocol of Mutual Exclusion

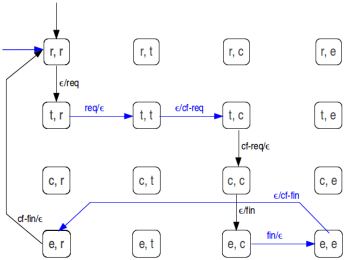

$\mathcal{U}$
$\text{remn}_{\mathcal{U}}$
ϵ/req
$\text{try}_{\mathcal{U}}$
cf-req/ϵ
$\text{crit}_{\mathcal{U}}$
ϵ/fin
$\text{exit}_{\mathcal{U}}$
cf-fin/ϵ
req, fin
cf-req, cf-fin
$\mathcal{C}$
$\text{remn}_{\mathcal{C}}$
req/ϵ
$\text{try}_{\mathcal{C}}$
ϵ/cf-req
$\text{crit}_{\mathcal{C}}$
fin/ϵ
$\text{exit}_{\mathcal{C}}$
ϵ/cf-fin

Causal Relation Between Output and Input of Different Systems - Channel Based Restriction

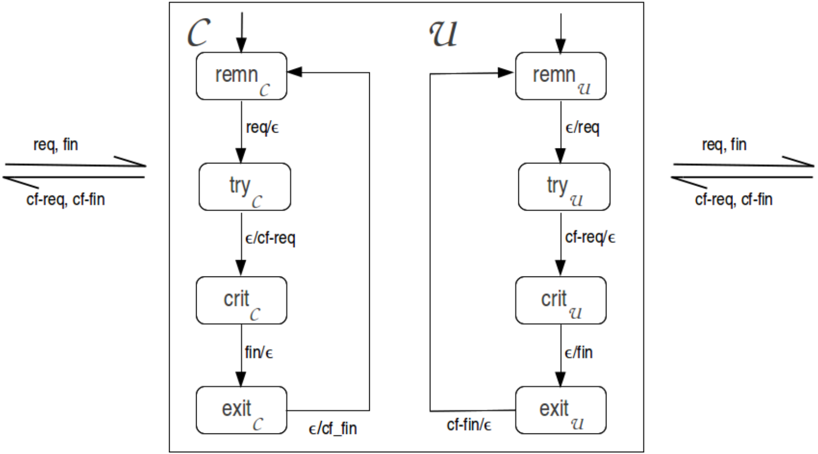
Weakly synchronized product



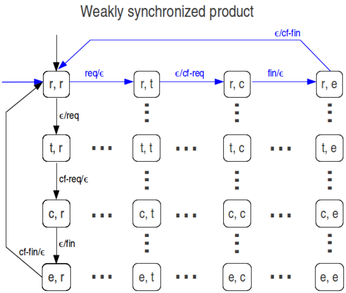
Channel based restricted product

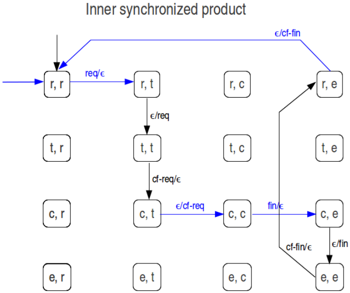


Example: Man in the Middle of the Protocol of Mutual Exclusion


req, fin
cf-req, cf-fin
$\mathcal{C}$
$\text{remn}_{\mathcal{C}}$
req/ϵ
$\text{try}_{\mathcal{C}}$
ϵ/cf-req
$\text{crit}_{\mathcal{C}}$
fin/ϵ
$\text{exit}_{\mathcal{C}}$
ϵ/cf_fin
$\mathcal{U}$
$\text{remn}_{\mathcal{U}}$
ϵ/req
$\text{try}_{\mathcal{U}}$
cf-req/ϵ
$\text{crit}_{\mathcal{U}}$
ϵ/fin
$\text{exit}_{\mathcal{U}}$
cf-fin/ϵ
req, fin
cf-req, cf-fin

Causal Relation Between Input and Output of the Same System - Condition Based Restriction


Weakly synchronized product


Inner synchronized product

Semantic is key!

Common phrases and what the imply . . .

- "The process is in the objects"
- "message exchange patterns"
- "communication is about sending data from one system to another"
- "message based integration"
- "loose coupling is just asynchronous message exchange"
- "process interactions should be based on idempotent, stateless methods with no side effects"
- "the actual meaning of an interface is independent of its implementation"
- "for integration, write services, expose your object model!"

Architectural Principles

A process oriented application architecture is based on

- Clear system borders, clear layering
- A dedicated top process layer - separating reusable from non-reusable parts
- An internal event model - formalizing upward communication
- Sending/receiving documents - the basis for non-deterministic interactions
- Roles implementing protocols
- An adequate component model with protocol signatures

A process oriented application architecture simplifies

- Integration
- Reuse
- Security

Thank You!

Any questions?

Johannes.Reich@sophoscape.de

Literature

J. Reich (2008), Modelling the interaction of distributed systems as protocols. Proceedings of the MCETECH 2008, pp. 16-24.

J. Reich (2009), The relation between protocols and games, in S. Fischer, E. Maehle und R. Reischuk (Hrsg.), GI Lecture Notes in Informatics, Proceedings of the 39. Annual Conference of the Deutsche Gesellschaft für Informatik 2009 in Lübeck, pp. 3453-3464

J. Reich (2010), Finite system composition and interaction, in Klaus-Peter Fähnrich, Bogdan Franczyk (Eds), GI Lecture Notes in Informatics, Proceedings of the 40. Annual Conference of the dt. Gesellschaft für Informatik 2010 in Leipzig, Vol. 2, pp. 624-637.

J. Reich (2011), Process synthesis from multiple interaction specifications, Proceedings of the 41. Annual Conference of the dt. Gesellschaft für Informatik 2011 in Berlin.

J. Reich (2012), Processes, Roles and Their Interactions, in Johannes Reich and Bernd Finkbeiner: Proceedings Second International Workshop on Interactions, Games and Protocols (IWIGP 2012), Tallinn, Estonia, 25th March 2012, Electronic Proceedings in Theoretical Computer Science 78, pp. 24–38.