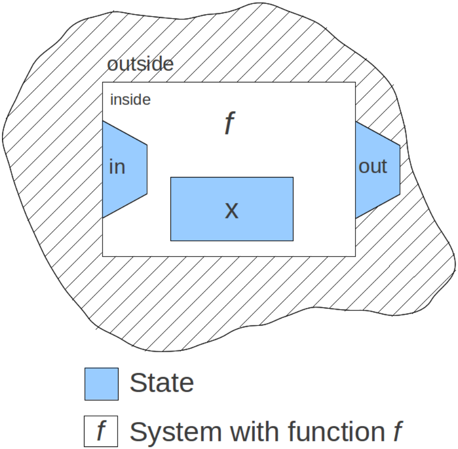
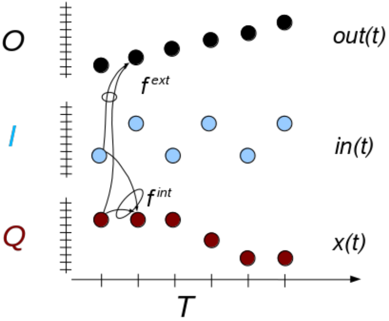
A logical perspective on (finite) software systems and their composition

Johannes Reich, johannes.reich@sophoscape.de

2012-12-15

Systems - Informally





Finite Systems - Formal Definition

A **finite system** is defined by a tuple $\mathcal{S} = (T, succ, Q, I, O, x, in, out, f)$.

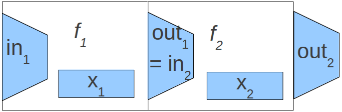
- T is the enumerable set of time values starting with 0 such that $succ : T \rightarrow T$ is the invertible time successor function.
- Q, I and O are the finite sets of state values for the internal, input and output states $(x, in, out) : T \rightarrow (Q, I, O^\epsilon)$.
- $f = (f^{ext}, f^{int}) : I \times Q \rightarrow O^\epsilon \times Q$ is a function describing the time evolution or system operation triggered by an update of its input parameters and updating the internal and output state in one time step for each $t \in T$:

$$\begin{pmatrix} out(t+1) \\ x(t+1) \end{pmatrix} = \begin{pmatrix} f^{ext}(in(t), x(t)) \\ f^{int}(in(t), x(t)) \end{pmatrix}.$$

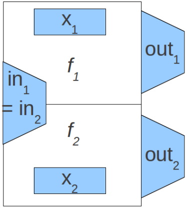
ϵ symbolizes the empty character and $I^\epsilon = I \cup \epsilon$ and $O^\epsilon = O \cup \epsilon$.
The n-fold application of $succ$ is written as $t +_{\mathcal{S}} n := succ^n_{\mathcal{S}}(t)$

System Composition/Super System Formation

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

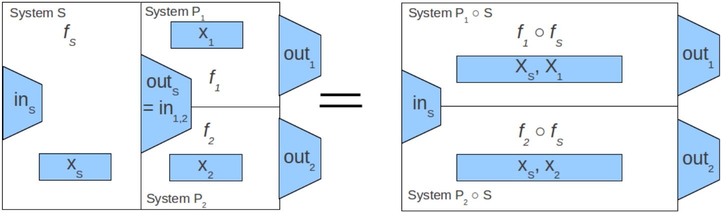


Parallel composition ($\mathcal{S}_2 || \mathcal{S}_1$)

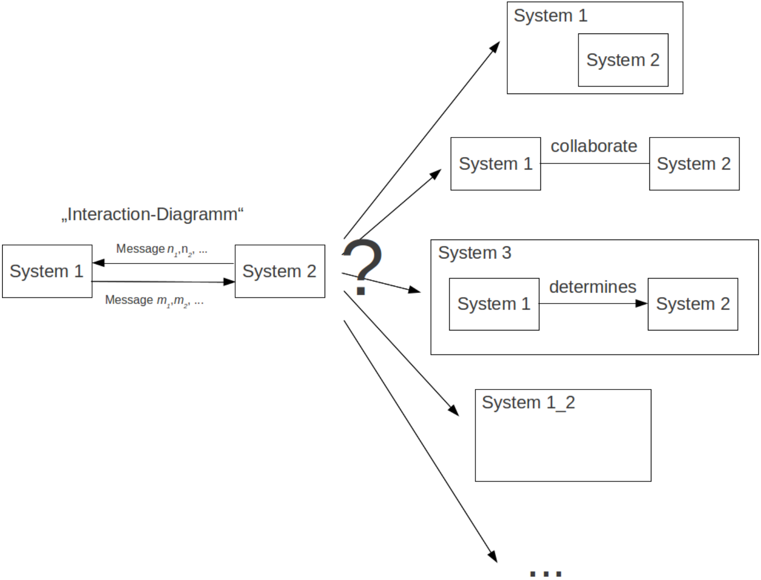


Combined System Composition

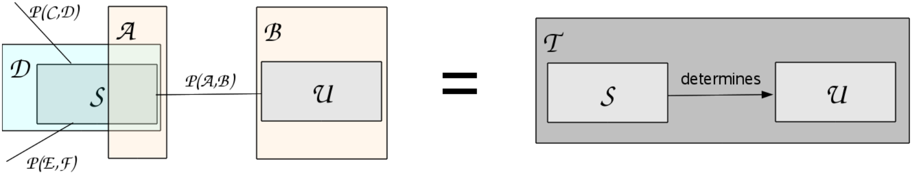
The right distribution law holds: $(\mathcal{P}_1||\mathcal{P}_2) \circ \mathcal{S} = (\mathcal{P}_1 \circ \mathcal{S})||(\mathcal{P}_2 \circ \mathcal{S})$



Richer Interaction Semantics



Deterministically Interacting Finite Systems and Supersystem Formation



Proposition: Let $\mathcal{S}$ and $\mathcal{U}$ be two systems described by DFIOAs $\mathcal{D}$ and $\mathcal{B}$ respectively. $\mathcal{U}$ interacts with $\mathcal{S}$ only by the consistent protocol $\mathcal{P}(\mathcal{A},\mathcal{B})$ with a set of final states as acceptance component, where $\mathcal{A}$ is an NFIOA describing only a projection of $\mathcal{S}$. $\mathcal{S}$ additionally interacts with other systems, denoted by $\sim\mathcal{U}$, by other consistent protocols. Then $\mathcal{S}$ and $\mathcal{U}$ are subsystems of a larger system $\mathcal{T}$.

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);
}
```

Summary

The effect of system interaction on system composition can be classified as:

- Parallel processing or strict sequential interaction results in strictly hierarchical super system formation
- Deterministic bidirectional interactions together with certain consistency conditions result in (recursive) super system formation
- Nondeterministic bidirectional interactions together with certain consistency conditions results in provably no super system formation
- further, non-classified relations.

Literature:

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.

Thank You!

Any questions?

Johannes.Reich@sophoscape.de