Component Interfaces with Contracts on Ports: Meta-Theory and Instantiation

Rolf Hennicker

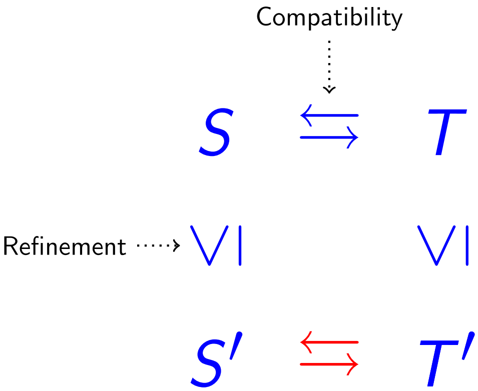
Ludwig-Maximilians-Universität München, Germany

Joint work with Sebastian Bauer, Axel Legay

Introduction

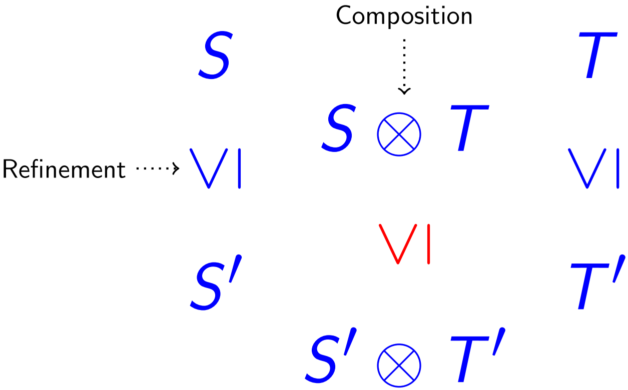
- Context: *Specification of reactive software components.*
  They interact with their environment and have a significant dynamic behavior depending on states.
- *Interface specifications* are important for both, the **correct usage** and the **correct implementation** of a component.
  They provide a "black box" view of a component.
- Crucial aspects:
  - *Refinement* of interface specifications
    (to obtain a correct implementation!)
  - *Compatibility* of interfaces of interacting components
    (to avoid communication errors!)
  - *Composition* of interface specifications
    (to construct larger systems from smaller ones!)

Requirement 1: Preservation of Compatibility by Refinement



Requirement 2: Preservation of Refinement by Composition

if $S \leftrightarrows T$, then



Interface Theory

Definition (inspired by De Alfaro, Henzinger)

An **interface theory** is a tuple $(\mathfrak{S}, \leq, \leftrightarrows, \otimes)$ consisting of

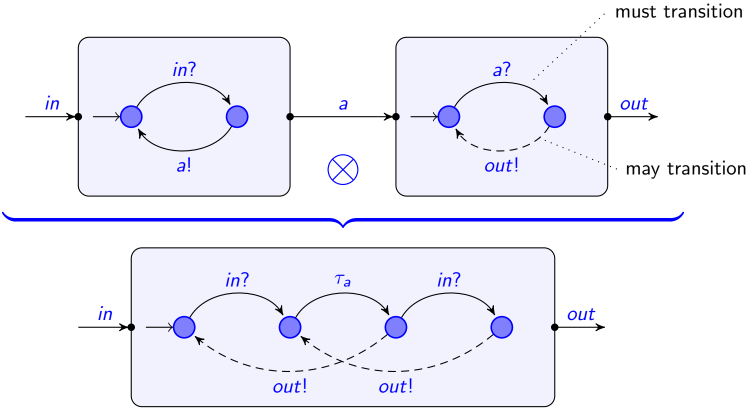
- a class $\mathfrak{S}$ of **interface specifications**
- a reflexive and transitive **refinement relation** $\leq \subseteq \mathfrak{S} \times \mathfrak{S}$
- a symmetric **compatibility relation** $\leftrightarrows \subseteq \mathfrak{S} \times \mathfrak{S}$
- a partial, commutative **composition operator** $\otimes : \mathfrak{S} \times \mathfrak{S} \to \mathfrak{S}$

satisfying

1. **Preservation of compatibility**
2. **Compositionality of refinement**

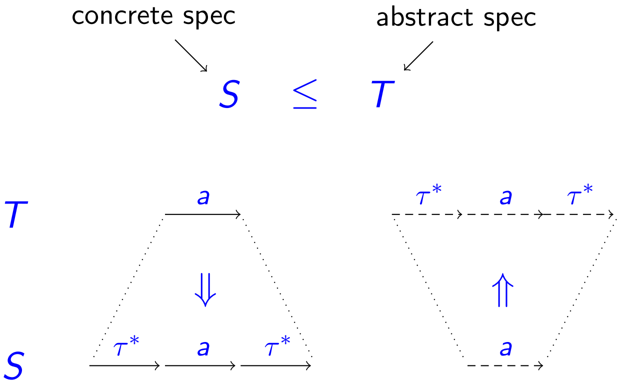
Example: Modal I/O-Transition Systems (MIOs)

[Larsen, Nyman, Wasowski 2007]



"must ⊗ must = must"

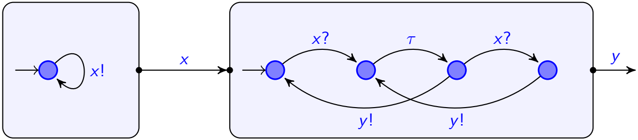
Weak Modal Refinement [Hüttel, Larsen 1989]



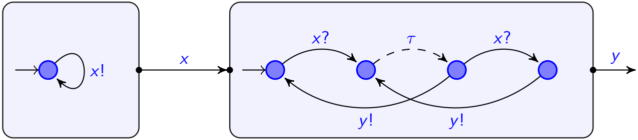
- If all transitions are "may", then $\leq$ is weak trace inclusion.
- If all transitions are "must", then $\leq$ is weak bisimulation.

Weak Compatibility [Bauer et al. 2010]

Weakly compatible MIOs:



Incompatible MIOs:



Theorem: MIOs with weak modal refinement, weak compatibility and synchronous composition form an interface theory.

We need more ...

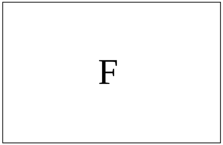
Interface Theories provide

- a nice abstract framework focusing on rudimentary requirements for component-based design.

But

- there is a lack of structure; they do not provide any mechanism to identify communication points.

Interface specification (no structure)



Labeled Interface Theory

Idea: Any interface specification should be equipped with a set of labels (representing visible actions).

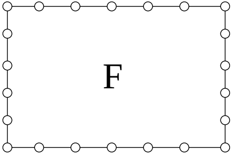
Definition

A *labeled interface theory* is a tuple $(\mathfrak{S}, \leq, \leftrightarrows, \otimes, \mathcal{L}, \ell)$ consisting of

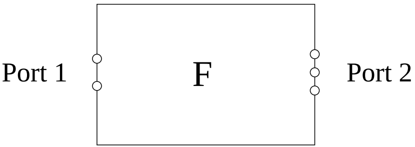
- an interface theory $(\mathfrak{S}, \leq, \leftrightarrows, \otimes)$,
- a set $\mathcal{L}$ of labels,
- a function $\ell : \mathfrak{S} \to \wp_{\text{fin}}(\mathcal{L})$ assigning a finite set of labels, such that
  - if $\ell(S) \cap \ell(T) = \emptyset$, then $S \otimes T$ is defined,
  - If $S \otimes T$ is defined, then $\ell(S \otimes T) = (\ell(S) \cup \ell(T)) \setminus (\ell(S) \cap \ell(T))$,
  - ...

From Labeled Interfaces to Component Interfaces

(1) Interface specification **with labels**



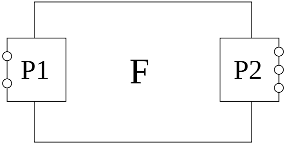
(2) Interface specification **with ports**



We want more: Behavior specifications on ports!

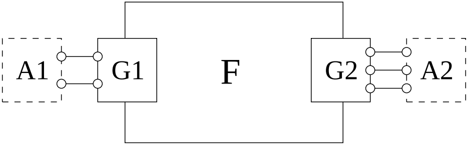
From Labeled Interfaces to Component Interfaces

(3) Interface specification with **port specifications (protocols)**

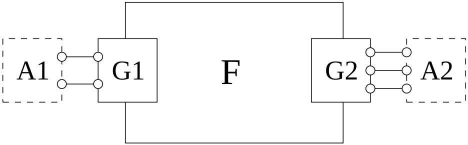


Problem: Obligations for user and implementor often mixed up!

(4) Interface specification with **port contracts**



Semantic Requirements



1. **Compatibility on ports:**
   Each port contract should have compatible assumptions and guarantees, i.e.

   $A1 \leftrightarrows G1$ and $A2 \leftrightarrows G2$.

2. **Reliability:**
   The frame specification F should satisfy each guarantee (on a port) under the given assumptions (on the other ports), i.e.

   $A1 \otimes F \leq G2$ and $A2 \otimes F \leq G1$.

Port Contracts and Component Interfaces (formally)

Given a labeled interface theory $(\mathfrak{S}, \leq, \leftrightarrows, \otimes, \mathcal{L}, \ell)$.

Definition

A *port contract* is a pair (A, G) with $A, G \in \mathfrak{S}$ such that $\ell(A) = \ell(G)$ and $G \leftrightarrows A$.

Definition

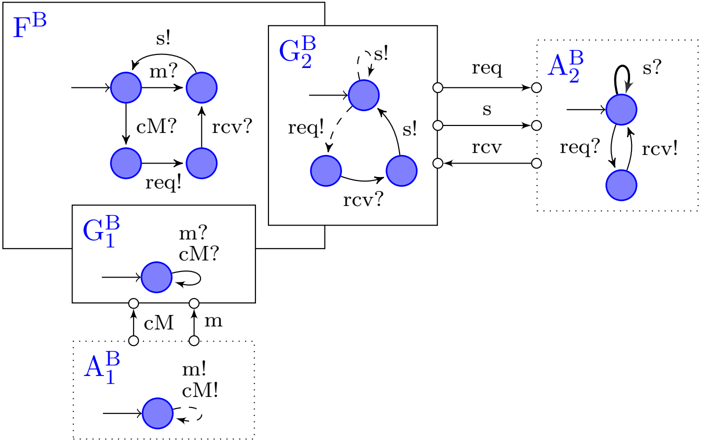
A *component interface* is a pair $C = (F, \{P_1, \dots P_n\})$ such that

- $F \in \mathfrak{S}$ is an interface specification, called *component frame*,
- $\{P_1, \dots P_n\}$ is a set of port contracts $P_i = (A_i, G_i)$.

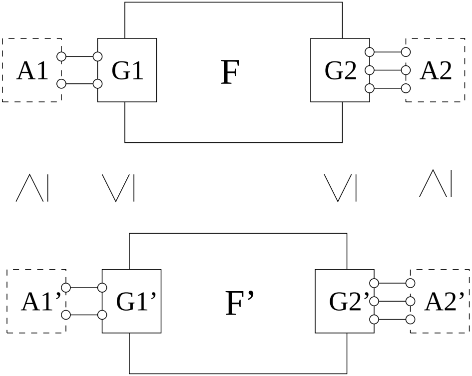
such that:

1. $\ell(P_i) \cap \ell(P_j) = \emptyset$ for all $i \neq j$,
2. $\ell(F) = \ell(P_1) \cup \ldots \cup \ell(P_n)$,
3. Reliability on each port.

Example: Broker with Port Contracts

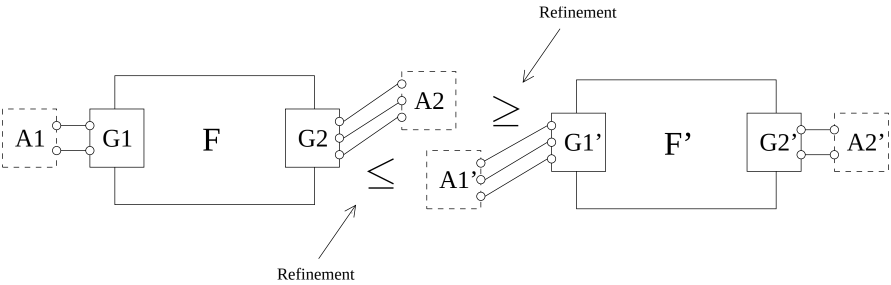


Refinement of Component Interfaces



Notation: $C' \sqsubseteq C$

Compatibility of Component Interfaces

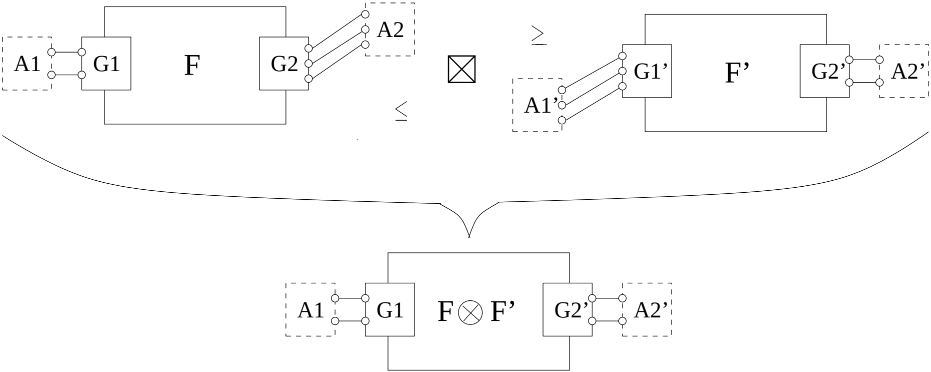


Notation: $C \leftleftarrows\!\!\rightrightarrows C'$

Facts: If $C \leftleftarrows\!\!\rightrightarrows C'$ then

- $G2 \leftrightarrows G1'$,
- if $E1 \leq A1$, $A1 \otimes I \leq A1 \otimes F$ and $E2' \leq A2'$, $A2' \otimes I' \leq A2' \otimes I'$, then $E1 \otimes I \leftrightarrows E2' \otimes I'$.

Composition of Compatible Component Interfaces



Fact: Composition preserves reliability!

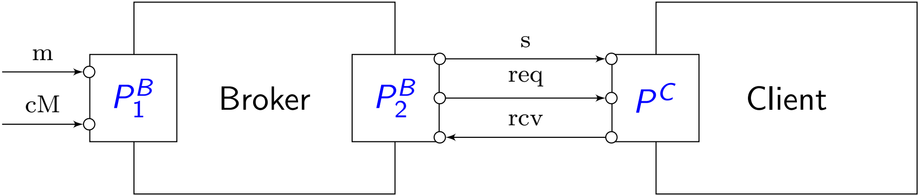
$(A1 \otimes F \otimes F') \leq G2'$ and $(A2' \otimes F' \otimes F) \leq G1$.

Results

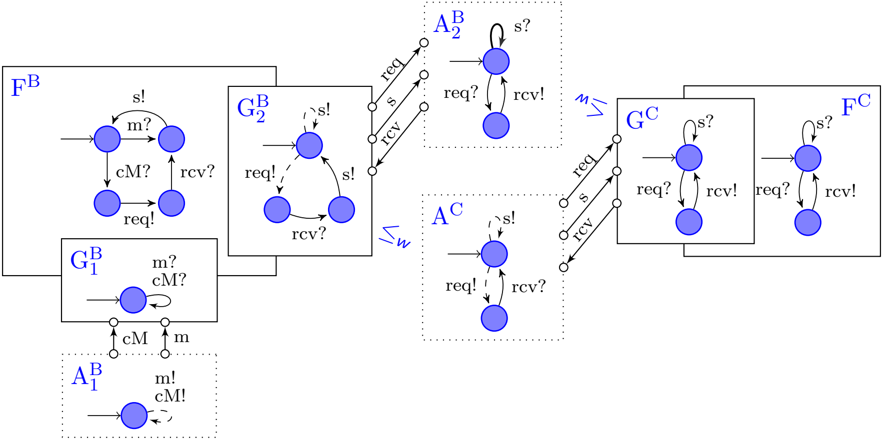
- Preservation of component compatibility:
  if $C \leftrightarrows D, C' \sqsubseteq C$ and $D' \sqsubseteq D$ then $C' \leftrightarrows D'$.
- Compositionality of component refinement:
  if $C' \sqsubseteq C$ and $D' \sqsubseteq D$ then $C' \boxtimes D' \sqsubseteq C \boxtimes D$.

Theorem:
Let $LTh = (\mathfrak{S}, \leq, \leftrightarrows, \otimes, \mathcal{L}, \ell)$ be an arbitrary labeled interface theory.
The class of component interfaces over LTh is itself an interface theory with $\sqsubseteq, \leftrightarrows$ and $\boxtimes$.

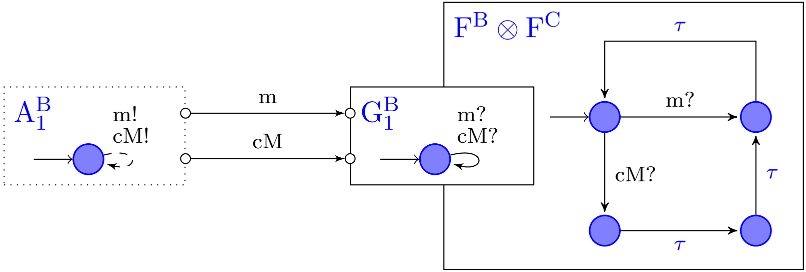
Example: Broker and Client Components



Example: Broker and Client Component Interfaces

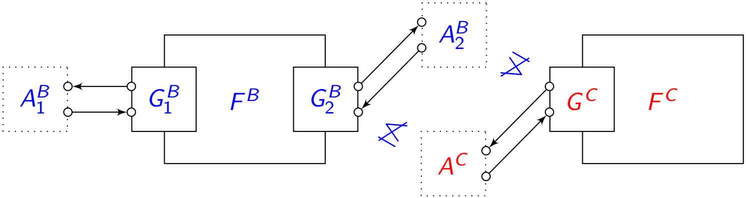


Example: Composition of Broker and Client Interfaces

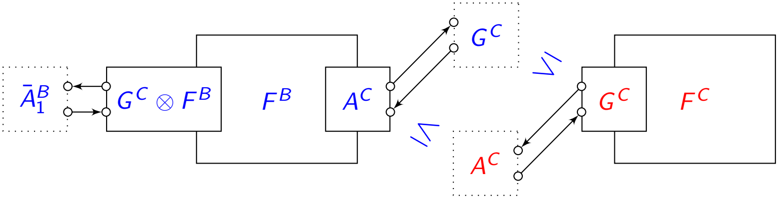


Adaptation of Component Interfaces

Initial situation:



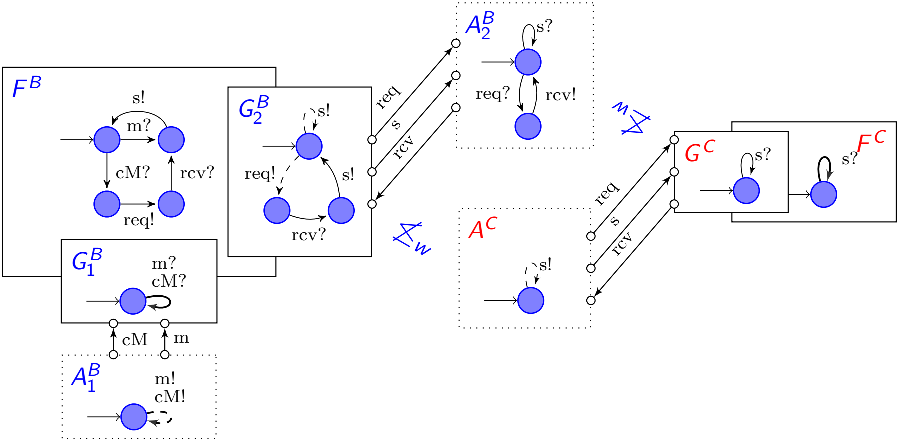
Final situation:



Task: Find assumption $\bar{A}_1^B$ such that $\bar{A}_1^B \leftrightarrows G^C \otimes F^B$ and $\bar{A}_1^B \otimes F^B \leq A^C$!

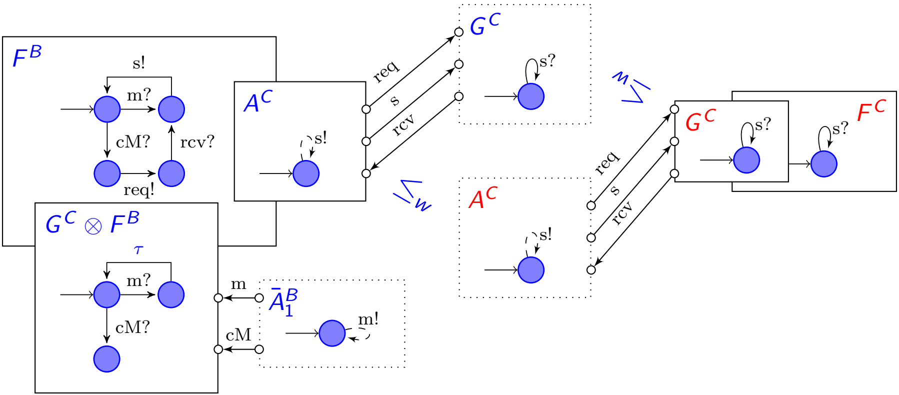
Adaptation: Example

Initial situation:



Adaptation: Example continued

Final situation:



Conclusion

- Interface theories are a nice abstract framework but they lack structure for proper component-based design.
- Just by introducing labels for interfaces one can do a lot more.
- One can construct a generic, contract-based framework for component interfaces with ports *on top of any labeled interface theory*.
- The framework provides design and adaptation guidelines.
- Instantiation by modal I/O-transition systems.
- Further instantiations should be studied,
  e.g. integrating data constraints, asnychronous communication, ...
- Tool support: MIO-Workbench [Bauer, Mayer et al. 2010].