Towards a Theory of Glue

Simon Bliudze

Rigorous System Design Laboratory
EPFL IC, Station 14, CH-1015, Lausanne, Switzerland
Simon.Bliudze@epfl.ch

Interaction and Concurrency Experience 2012
Stockholm, June 16$^{\text{th}}$, 2012

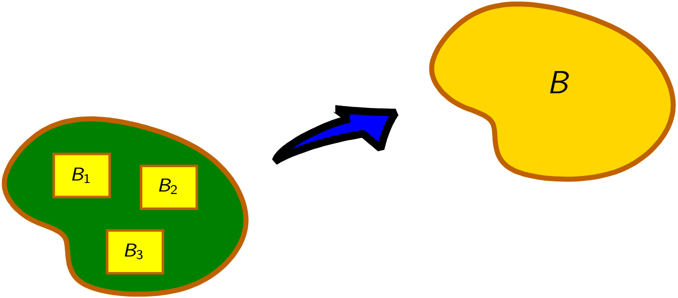
Outline

- Why behaviour types?
- Definition
- Co-algebraic example
- Some speculation about glue

Outline

- Why behaviour types?
- Definition
- Co-algebraic example
- Some speculation about glue

Component-based design



Components: semantic behaviour domain $\mathcal{B}$ (objects, actors, machines, CORBA components, JavaBeans, threads, etc.)

Composition: operators $\mathcal{B}^n \to \mathcal{B}$ or $2^{\mathcal{B}} \to \mathcal{B}$

Comparing component-based frameworks

How do we compare two component-based frameworks $\mathcal{C}_1, \mathcal{C}_2$?

Common approach:

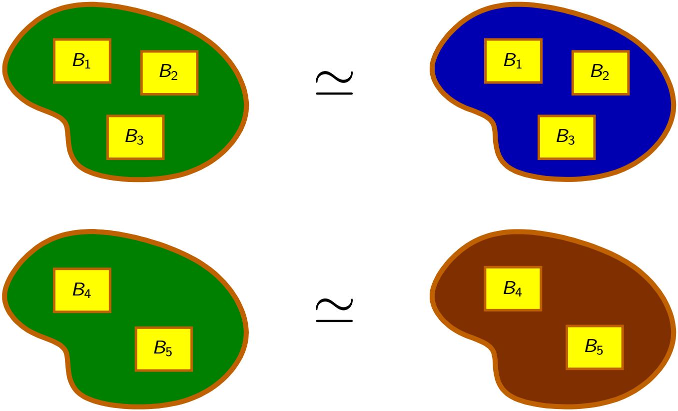
- What behaviours can be obtained in a given model?

$$\mathcal{C}_1(2^{\mathcal{B}}) \stackrel{?}{=} \mathcal{C}_2(2^{\mathcal{B}})$$

- Not satisfactory:
  - *id* $\in \mathcal{C}$ implies $\mathcal{B} \subseteq \mathcal{C}(2^{\mathcal{B}})$
  - $\mathcal{C}(2^{\mathcal{B}}) \subseteq \mathcal{B}$
  - Hence, $\mathcal{C}(2^{\mathcal{B}}) = \mathcal{B}$
  - Most *"behaviour"* models are Turing complete.

Example: Strong Expressiveness Preorder

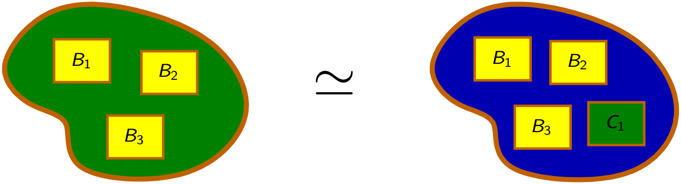
From Bliudze & Sifakis, "A Notion of Glue Expressiveness for Component-Based Systems". In CONCUR 2008.



$$\mathcal{C}_1 \leq \mathcal{C}_2 \quad \stackrel{def}{\Longleftrightarrow} \quad \forall g_1 \in \mathcal{C}_1, \forall \mathbf{B} \subset \mathcal{B},\ \exists g_2 \in \mathcal{C}_2 : g_1(\mathbf{B}) \simeq g_2(\mathbf{B})$$

What if behaviour domains are not the same? ($\mathcal{B}_1 \neq \mathcal{B}_2$)

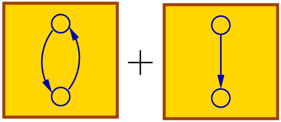
Separation of concerns

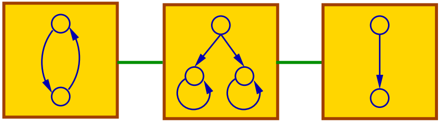


- Encapsulating features into different entities
- Layers (e.g. network protocol stack, hardware abstraction)
- Base functionality vs.
  - component life-cycle management
  - introspection
  - non-functional behaviour
- Sequential computation vs. (memoryless) coordination

Memoryless coordination intuitively

Non-deterministic choice:

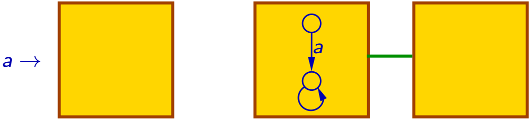




It is important to distinguish from interleaving

$$\frac{P \to P'}{P|Q \to P'|Q} \quad \frac{Q \to Q'}{P|Q \to P|Q'} \quad \Longrightarrow \quad \frac{P \to P' \qquad Q \to Q'}{P|Q \to P'|Q \to P'|Q'}$$

Prefixing:



Outline

- Why behaviour types?
- Definition
- Co-algebraic example
- Some speculation about glue

Understanding the types of behaviour

- What is the behaviour of several parallel components without any coordination constraints?
- How does one compare two behaviours?
- How does one model the simultaneous application of several composition operators?

Parallel composition

$(\mathcal{B}, \|, \mathbf{0})$ — a monoid (associative, not necessarily commutative).

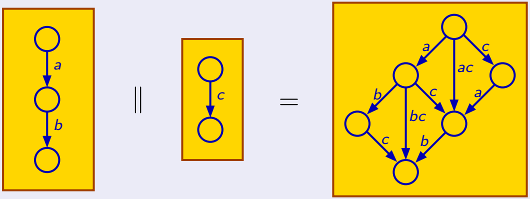
Traces

$B = (A, T)$, where $A \subseteq \mathbf{A}$ and $T \subseteq A^*$ $\qquad \mathbf{0} = (\emptyset, \{\varepsilon\})$

$B_1 \parallel B_2 = (A_1 \cup A_2, T_{12})$: $\bullet\bullet\bullet\bullet\bullet\bullet\bullet \parallel \bullet\bullet\bullet\bullet\bullet \rightsquigarrow \bullet\bullet\bullet\bullet\bullet\bullet\bullet\bullet\bullet\bullet\bullet\bullet$

Labelled Transition Systems

$B = (Q, A, \rightarrow)$, where Q is finite, $A \subseteq \mathbf{A}$, $\rightarrow \subseteq Q \times 2^A \times Q$



$\mathbf{0} = (\{*\}, \emptyset, \emptyset)$

A is a universal set of actions

Comparing behaviours

Property satisfaction, specification conformance

Simulation preorder: $\sqsubseteq \subseteq \mathcal{B} \times \mathcal{B}$, s.t.

- it is preserved by $\|$ $\quad (B_2 \sqsubseteq B_3 \Rightarrow B_1 \| B_2 \sqsubseteq B_1 \| B_3)$,
- $\mathbf{0} \sqsubseteq B$.

(Trace containment, simulation — not necessarily pre-congruences.)

Traces: $B = (A, T)$, $T \subseteq A^*$

$B_1 \sqsubseteq B_2 \overset{def}{\iff} A_1 \subseteq A_2 \ \wedge \ T_1 \subseteq T_2 \wr_{A_1}$ $\qquad$ ($T \wr_A$ — subsequences in A^*)

Labelled Transition Systems: $B = (Q, A, \rightarrow)$, $\rightarrow \subseteq Q \times 2^A \times Q$

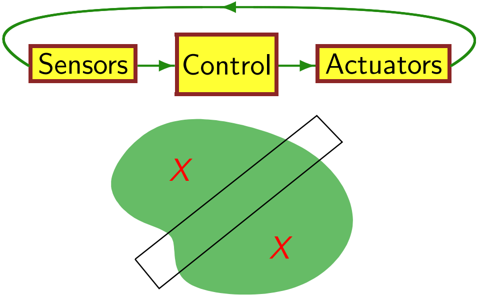
Consider the maximal relation $R \subseteq Q_1 \times Q_2$ s.t.

$$q_1 R q_2 \Rightarrow \forall q_1 \xrightarrow{a}_1 q_1',\ \exists\, q_2' \in Q_2, b \subseteq A_2 : \left(q_2 \xrightarrow{b} q_2' \ \wedge \ a \subseteq b \ \wedge \ q_1' R q_2'\right)$$

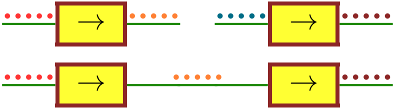
$B_1 \sqsubseteq B_2 \overset{def}{\iff} (A_1 \subseteq A_2) \ \wedge (R \text{ is total on } Q_1)$

Composition as constraints

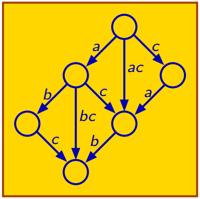
Control:



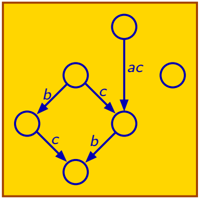
Dataflow:



Concurrency coordination:



$$a = c$$
$$b \neq c$$
$$\rightsquigarrow$$



Comparing behaviours & Composition operators

Component substitutability

Composition operators: $f : \mathcal{B}^n \to \mathcal{B}$

- $f(B_1, \ldots, B_n) \sqsubseteq B_1 \parallel \cdots \parallel B_n$,
- $B_i \preccurlyeq \tilde{B} \Rightarrow f(B_1, \ldots, B_i, \ldots, B_n) \preccurlyeq f(B_1, \ldots, \tilde{B}, \ldots, B_n)$.

Semantic preorder: $\preccurlyeq \subseteq \mathcal{B} \times \mathcal{B}$ s.t.

- $\preccurlyeq$ is preserved by $\parallel$ (i.e. $B_2 \preccurlyeq B_3 \Rightarrow B_1 \parallel B_2 \preccurlyeq B_1 \parallel B_3$)

(Ready simulation, bisimulation relations — precongruences.)

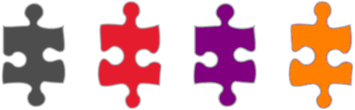
Meet operator: $\otimes : \mathcal{B} \times \mathcal{B} \to \mathcal{B}$ s.t.

- $(\mathcal{B}/\simeq, \preccurlyeq, \otimes)$, with $\simeq \stackrel{def}{=} \preccurlyeq \cap \preccurlyeq^{-1}$, is a meet-semilattice.

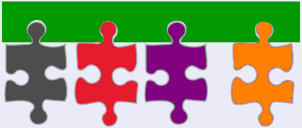
Traces, LTS

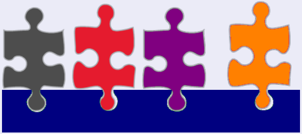
$\preccurlyeq = \sqsubseteq$. (Another example below.)

Software architectures

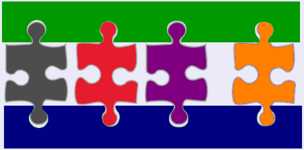
Components:  $\{B_i\}_{i=1}^4$

$f_1(B_1, B_2, B_3, B_4)$





$f_2(B_1, B_2, B_3, B_4)$



$(f_1 \otimes f_2)(B_1, B_2, B_3, B_4)$

Parallel composition

$(\mathcal{B}, \|, \mathbf{0})$ — a monoid (associative, not necessarily commutative).

Simulation preorder

$\sqsubseteq\ \subseteq \mathcal{B} \times \mathcal{B}$, s.t.

- it is preserved by $\|$ $\quad (B_2 \sqsubseteq B_3 \Rightarrow B_1 \| B_2 \sqsubseteq B_1 \| B_3)$,
- $\mathbf{0} \sqsubseteq B$.

Semantic preorder & meet operator

$\preccurlyeq\ \subseteq \mathcal{B} \times \mathcal{B}$ and $\otimes : \mathcal{B} \times \mathcal{B} \to \mathcal{B}$, s.t.

- $\preccurlyeq$ is preserved by $\|$ $\quad (B_2 \preccurlyeq B_3 \Rightarrow B_1 \| B_2 \preccurlyeq B_1 \| B_3)$
- $(\mathcal{B}/\!\simeq, \preccurlyeq, \otimes)$, with $\simeq \stackrel{def}{=} \preccurlyeq \cap \preccurlyeq^{-1}$, is a meet-semilattice.

Outline

- Why behaviour types?
- Definition
- Co-algebraic example
- Some speculation about glue

Deterministic functor coalgebras

F-coalgebras:
$(S, f : S \to F(S))$, where S — state-space, f — transition structure

Some examples:

$M = (\mathbf{B} \times \mathbf{Id})^{\mathbf{A}}$	input-enabled Mealy machines
$D = \mathbb{B} \times (1 + \mathbf{Id})^{\mathbf{A}}$	deterministic automata (no initial state)
$N = \mathbb{B}^2 \times \mathcal{P}_\omega(\mathbf{Id})^{\mathbf{A}}$	non-deterministic automata

Coalgebra homomorphism:

$$\begin{tikzcd} S_1 \arrow[r, "h"] \arrow[d, "f_1"'] & S_2 \arrow[d, "f_2"] \\ F(S_1) \arrow[r, "F(h)"'] & F(S_2) \end{tikzcd}$$

Coalgebra bisimulation:

$$\begin{tikzcd} S_1 \arrow[d, "f_1"'] & R \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "g"'] & S_2 \arrow[d, "f_2"] \\ F(S_1) & F(R) \arrow[l, "F(\pi_1)"] \arrow[r, "F(\pi_2)"'] & F(S_2) \end{tikzcd}$$

Non-Deterministic functors:
$F ::= \mathbf{Id} \mid \mathbf{B} \mid F \times F \mid F^{\mathbf{A}} \mid F + F \mid \mathcal{P}^*_\omega(F)$ ($\mathbf{B}$ is a join-semilattice with $\perp$)

Example: DF-coalgebra-based behaviour type (1/3)

Parallel composition & zero behaviour

$$(S_1, f_1) \parallel (S_2, f_2) \stackrel{def}{=} (S_1 \times S_2, (s_1, s_2) \mapsto f_1(s_1) \underset{S_1,S_2}{\overset{F}{\bowtie}} f_2(s_2))$$

$$x \underset{X,Y}{\overset{F}{\bowtie}} y \quad \stackrel{def}{=}$$

$$\begin{cases} (x, y), & F = \mathbf{Id}, \\ x \vee y, & F = \mathbf{B}, \\ \left(x_1 \underset{X,Y}{\overset{F_1}{\bowtie}} y_1,\ x_2 \underset{X,Y}{\overset{F_2}{\bowtie}} y_2\right), & F = F_1 \times F_2, x = (x_1, x_2), y = (y_1, y_2), \\ \lambda a. \left(x(a) \underset{X,Y}{\overset{G}{\bowtie}} y(a)\right), & F = G^{\mathbf{A}}. \end{cases}$$

$$\mathbf{0}_F = (1, f_F^0) \qquad f_F^0(*) \stackrel{def}{=} \begin{cases} *, & F = \mathbf{Id}, \\ \perp, & F = \mathbf{B}, \\ (f_{F_1}^0(*), f_{F_2}^0(*)), & F = F_1 \times F_2, \\ \lambda a. f_G^0(*), & F = G^{\mathbf{A}}, \end{cases}$$

Example: DF-coalgebra-based behaviour type (2/3)

Simulation preorder

$$x \leq_R^F y \stackrel{def}{\iff} \qquad R \subseteq S_1 \times S_2, x \in F(S_1), y \in F(S_2)$$

$$\begin{cases} (x, y) \in R, & F = \mathbf{Id}, \\ x \vee y = y, & F = \mathbf{B}, \\ x_1 \leq_R^{F_1} y_1 \wedge x_2 \leq_R^{F_2} y_2, & F = F_1 \times F_2, x = (x_1, x_2), y = (y_1, y_2), \\ \forall a \in \mathbf{A},\ x(a) \leq_R^G y(a), & F = G^{\mathbf{A}}, \end{cases}$$

$$(S_1, f_1) \sqsubseteq (S_2, f_2) \stackrel{def}{\iff}$$

$$\exists R \subseteq S_1 \times S_2 : \text{total on } S_1 \text{ and s.t.} \forall (s_1, s_2) \in R,\ f_1(s_1) \leq_R^F f_2(s_2)$$

Semantic preorder & meet operator

$(S_1, f_1) \preccurlyeq (S_2, f_2) \stackrel{def}{\iff} \exists R \subseteq S_1 \times S_2$: bisimulation total on S_1

$(S_1, f_1) \otimes (S_2, f_2) \stackrel{def}{=} (R, g)$ — maximal bisimulation (g unique for DF)

Example: DF-coalgebra-based behaviour type (3/3)

Proposition: $(DF/\simeq, \preccurlyeq, \otimes)$ is a meet-semilattice.

Proof — Let $(R, g) = (S_1, f_1) \otimes (S_2, f_2)$ and $(S, f) \preccurlyeq (S_1, f_1), (S_2, f_2)$

$$
\begin{tikzcd}
(T,h) \arrow[r] \arrow[d] & (R_1,g_1) \arrow[r] \arrow[d] & (S_1,f_1) \\
(R_2,g_2) \arrow[r] \arrow[d] & (S,f) & {} \arrow[u] \\
(S_2,f_2) & {} \arrow[l] & (R,g) \arrow[l] \arrow[u]
\end{tikzcd}
$$

1. $T = \{(s', s_1, s'', s_2) \,|\, (s', s_1) \in R_1, (s'', s_2) \in R_2 \text{ and } s' = s''\}$.
2. Projection $T_{2,4} \subseteq R$.
3. $T \cong \{(s, s_1, s_2) \,|\, (s, s_1) \in R_1, (s, s_2) \in R_2\}$ — bisimulation on (S, f) and (R, g) total on S. $\square$

Outline

- Why behaviour types?
- Definition
- Co-algebraic example
- Some speculation about glue

Memoryless glue operators

Separation of concerns: separating the computation of a system from the application of *glue operators* coordinating its atomic components.

1. A glue operator should not add *state* to the coordinated system.
2. The actions possible in a global state of the composed system should be completely determined by the properties of the corresponding states of the constituent subsystems.

Glue operators: Let $(\mathcal{B}, \dots)$ be a behaviour type over F-coalgebras. Composition operator $gl : \mathcal{B}^n \to \mathcal{B}$ is a *glue operator* iff there exists a natural transformation $sync : F^n \to F$, such that, for any $\{B_i = (S_i, f_i)\}_{i=1}^n$, $gl(B_1, \dots, B_n) = (S, f)$ with

1. $S \; = \; \prod_{i=1}^{n} S_i$,
2. $f(s) \; = \; sync(f_1(s_1), \dots, f_n(s_n))$, for all $s = (s_1, \dots, s_n) \in S$.

Some speculation

Choice hypothesis

Choice operator: $(S_1, f_1) + (S_2, f_2) \stackrel{def}{=} (S_1 + S_2, f_1 + f_2)$.

Hypothesis: There is no glue operator gl, such that $gl \simeq +$.

/Consider something like $(\{q_1\}, \{q_1 \xrightarrow{a} q_1\})$ and $(\{q_2\}, \{q_2 \xrightarrow{b} q_2\})$./

Some speculation

Choice hypothesis

Choice operator: $(S_1, f_1) + (S_2, f_2) \stackrel{def}{=} (S_1 + S_2, f_1 + f_2)$.

Hypothesis: There is no glue operator gl, such that $gl \simeq +$.

Consider something like $(\{q_1\}, \{q_1 \xrightarrow{a} q_1\})$ and $(\{q_2\}, \{q_1 \xrightarrow{b} q_2\})$.

Some speculation

Choice hypothesis

Choice operator: $(S_1, f_1) + (S_2, f_2) \stackrel{def}{=} (S_1 + S_2, f_1 + f_2)$.

Hypothesis: There is no glue operator gl, such that $gl \simeq +$.

Consider something like $(\{q_1\}, \{q_1 \xrightarrow{a} q_1\})$ and $(\{q_2\}, \{q_1 \xrightarrow{b} q_2\})$.

 $\rightsquigarrow$  ? 

Some speculation

Choice hypothesis

Choice operator: $(S_1, f_1) + (S_2, f_2) \stackrel{def}{=} (S_1 + S_2, f_1 + f_2)$.

Hypothesis: There is no glue operator gl, such that $gl \simeq +$.

Consider something like $(\{q_1\}, \{q_1 \xrightarrow{a} q_1\})$ and $(\{q_2\}, \{q_1 \xrightarrow{b} q_2\})$.

 $\rightsquigarrow$  ?  $\simeq$ 

Some speculation
Choice hypothesis

Choice operator: $(S_1, f_1) + (S_2, f_2) \stackrel{def}{=} (S_1 + S_2, f_1 + f_2)$.

Hypothesis: There is no glue operator gl, such that $gl \simeq +$.

Consider something like $(\{q_1\}, \{q_1 \xrightarrow{a} q_1\})$ and $(\{q_2\}, \{q_1 \xrightarrow{b} q_2\})$.

 $\rightsquigarrow$  ?  $\simeq$ 

Hypothesis 2: For any glue operator gl,

$$gl(B_1, B_2) \simeq B_1 + B_2 \implies B_1 \simeq B_2.$$