

Extended Propositional Interaction Logic: Encoding order restrictions in architectures of component-based systems

Maria Pittou and George Rahonis

Aristotle University, School of Sciences, Department of Mathematics

Seminar on TCS, 8/12/2021

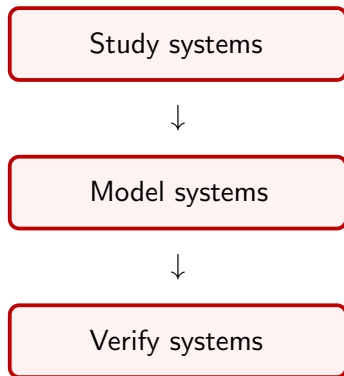
Motivation

Complex systems



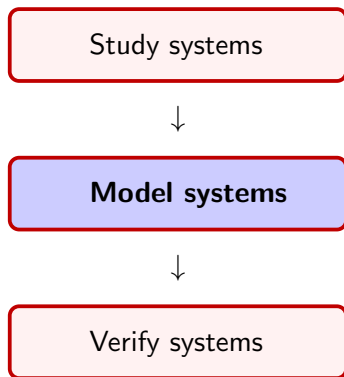
Motivation

System modeling



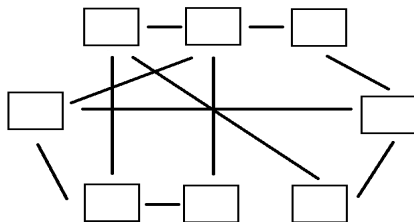
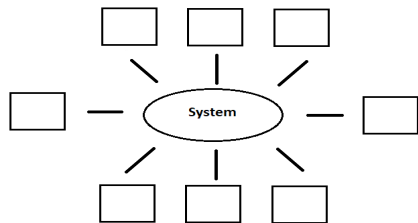
Motivation

System modeling



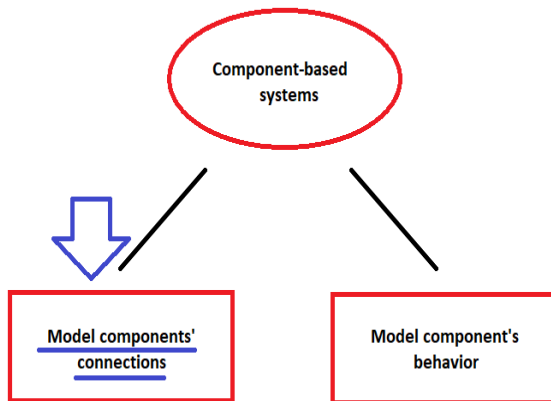
Motivation

Component-based modelling



Motivation

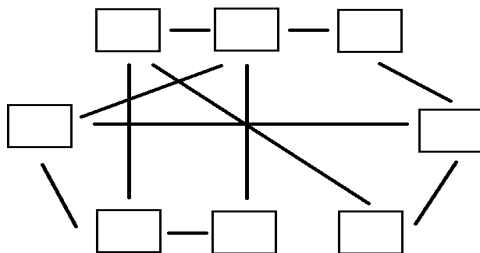
Component-based modelling



Motivation

Architectures

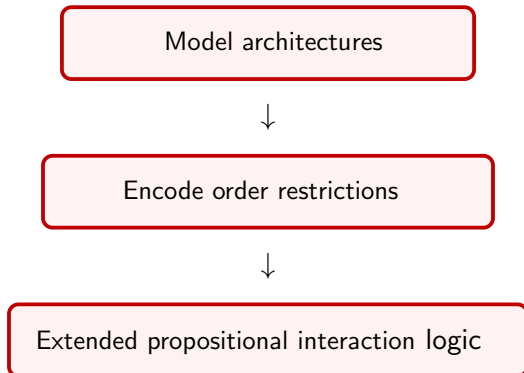
↔ **Architectures** enforce design rules on the components by characterizing the topology and the permissible **interactions** of systems.



Motivation

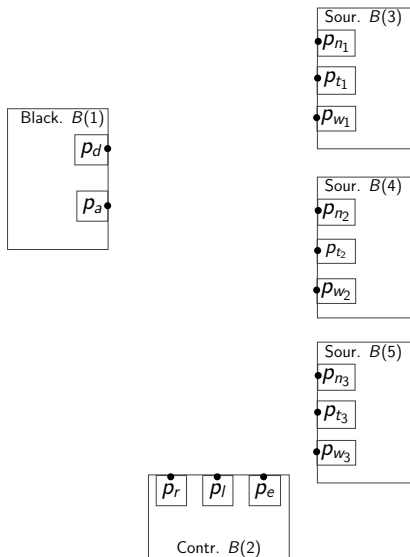
Problem to be solved

↪ What is the problem to be solved?



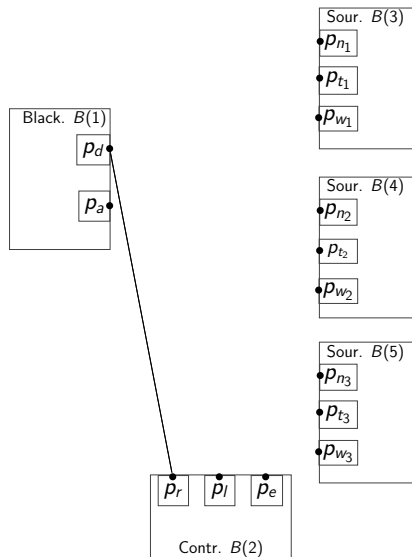
Architectures with order restrictions

Blackboard



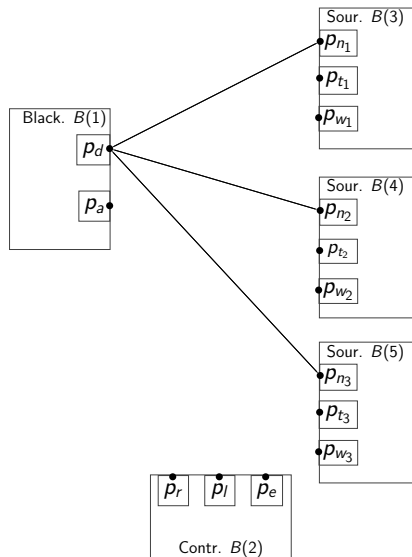
Architectures with order restrictions

Blackboard



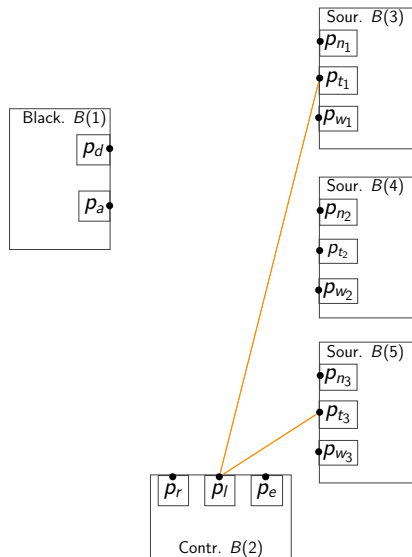
Architectures with order restrictions

Blackboard



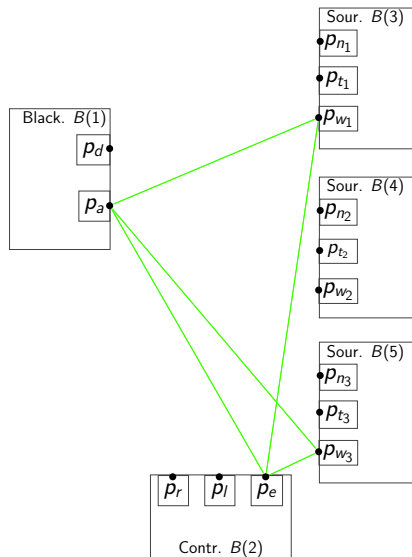
Architectures with order restrictions

Blackboard



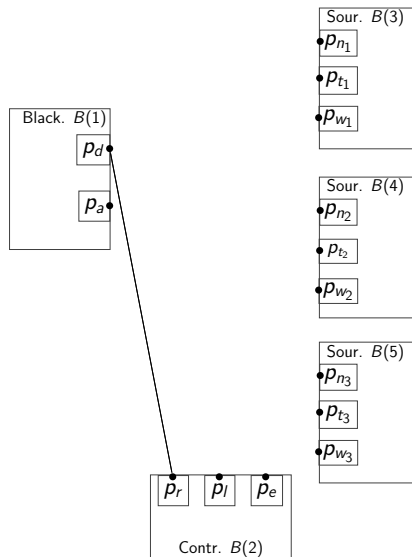
Architectures with order restrictions

Blackboard



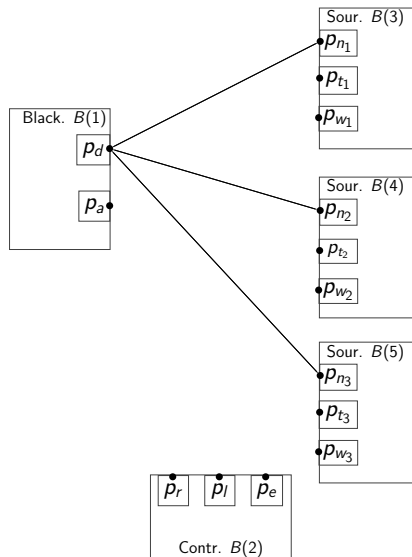
Architectures with order restrictions

Blackboard



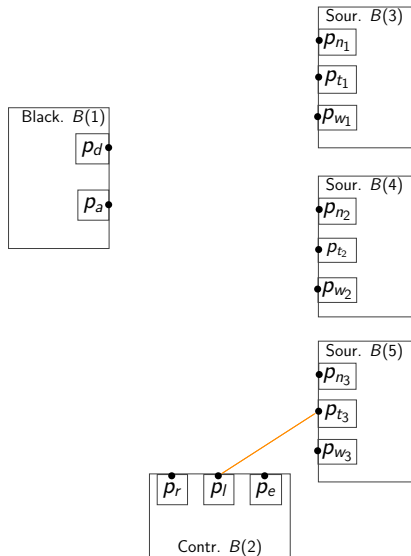
Architectures with order restrictions

Blackboard



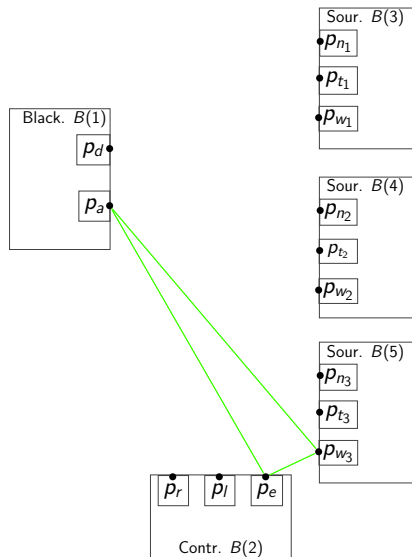
Architectures with order restrictions

Blackboard



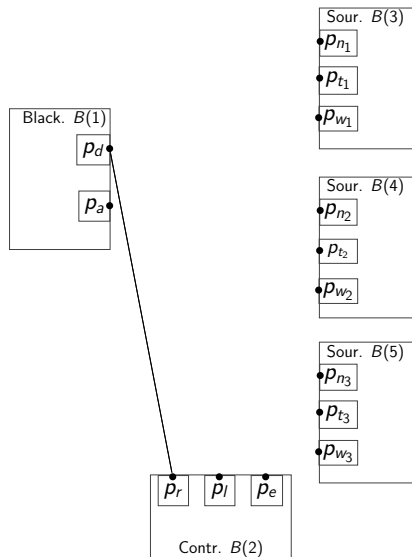
Architectures with order restrictions

Blackboard



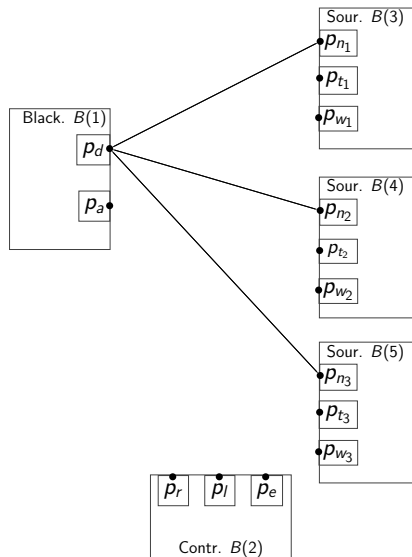
Architectures with order restrictions

Blackboard



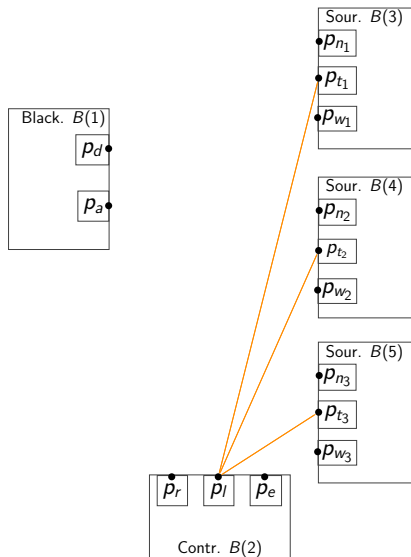
Architectures with order restrictions

Blackboard



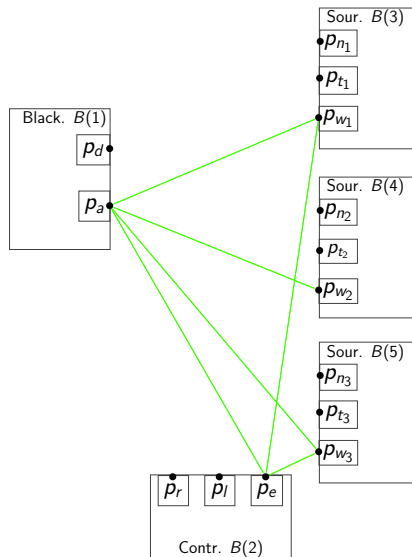
Architectures with order restrictions

Blackboard



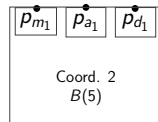
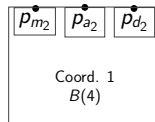
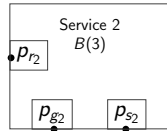
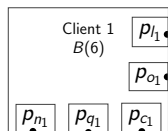
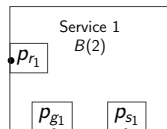
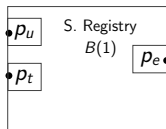
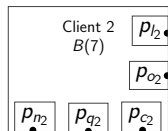
Architectures with order restrictions

Blackboard



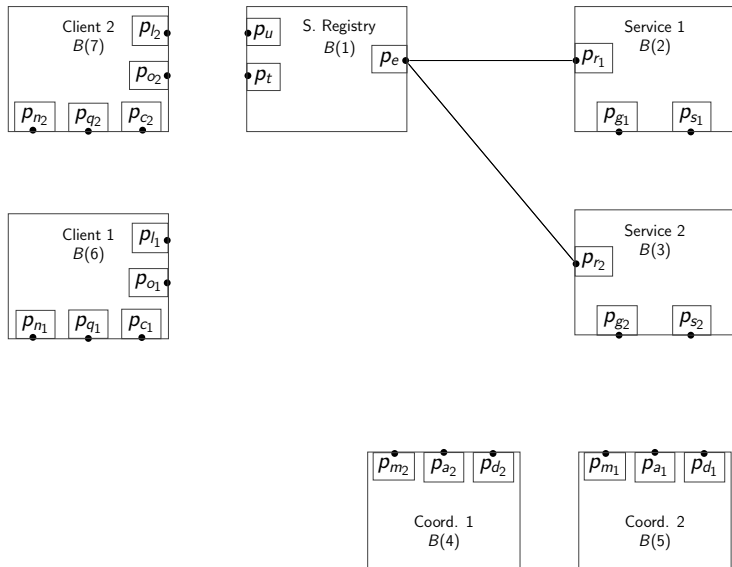
Architectures with order restrictions

Request/Response



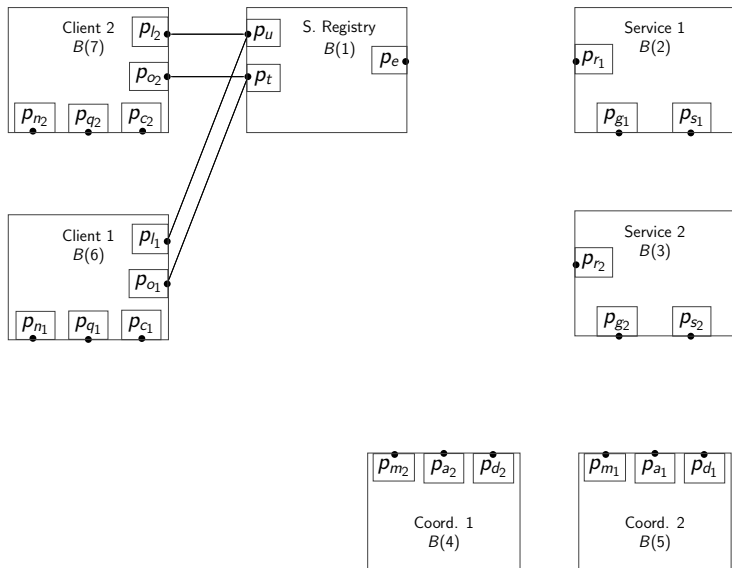
Architectures with order restrictions

Request/Response



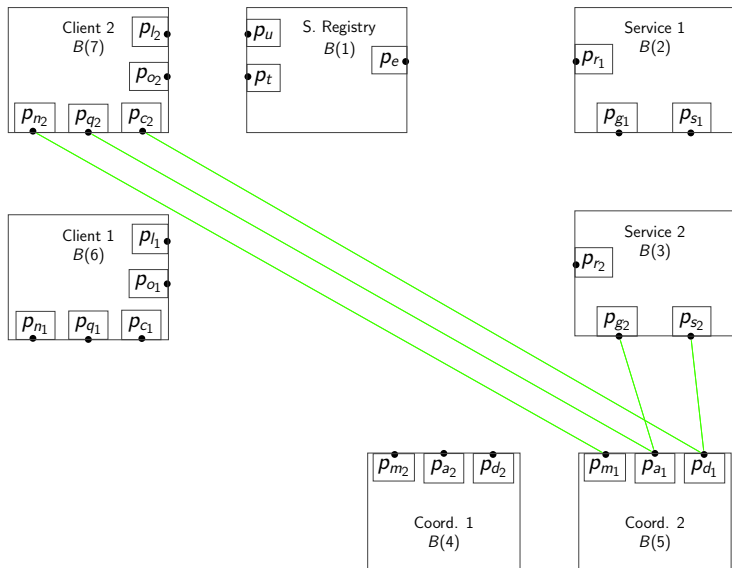
Architectures with order restrictions

Request/Response



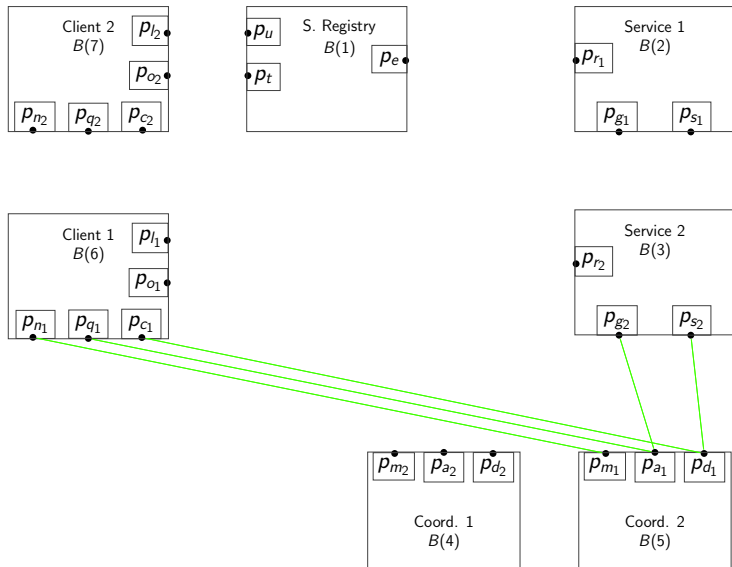
Architectures with order restrictions

Request/Response



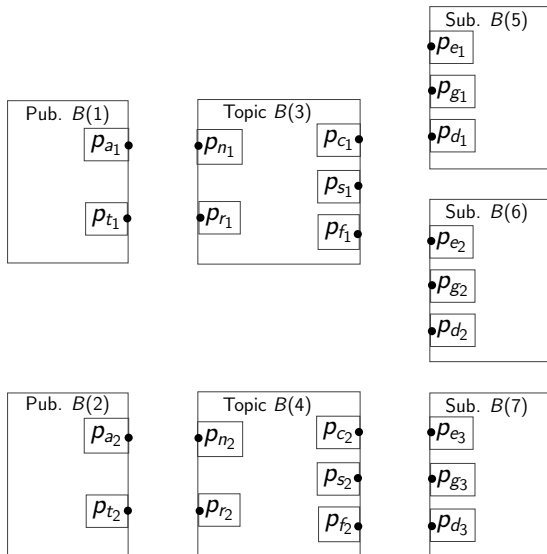
Architectures with order restrictions

Request/Response



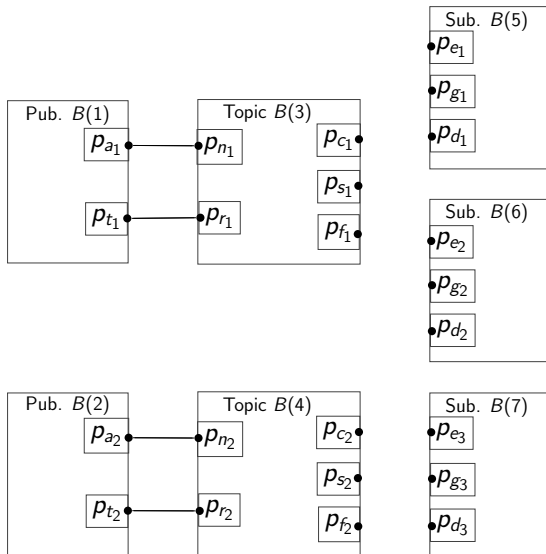
Architectures with order restrictions

Publish/Subscribe



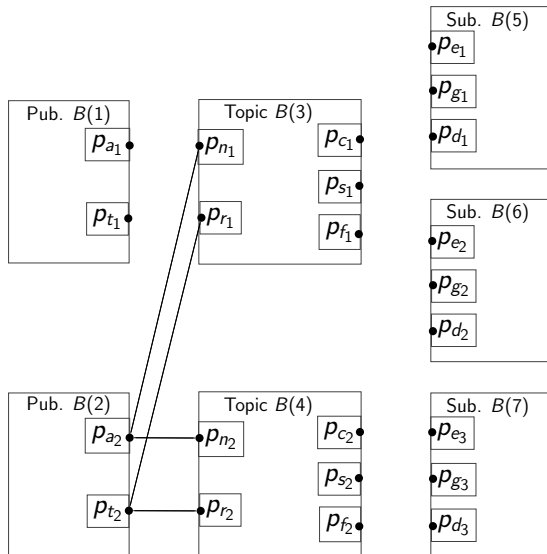
Architectures with order restrictions

Publish/Subscribe



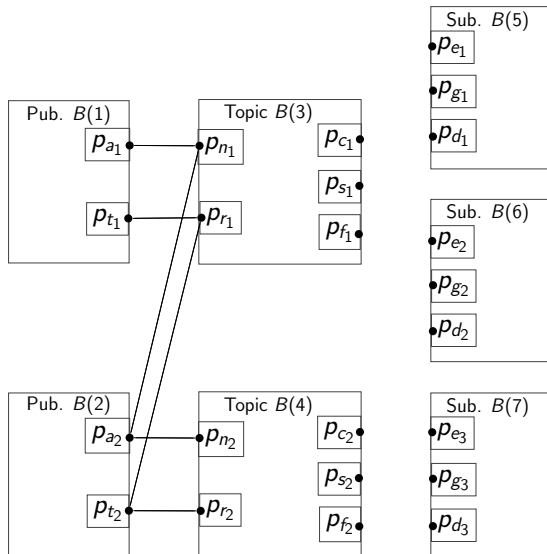
Architectures with order restrictions

Publish/Subscribe



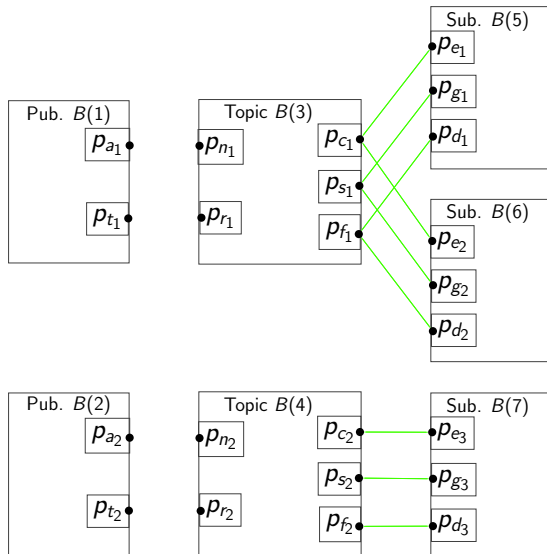
Architectures with order restrictions

Publish/Subscribe



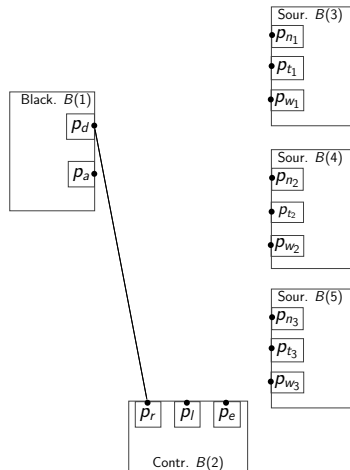
Architectures with order restrictions

Publish/Subscribe



How to encode interactions

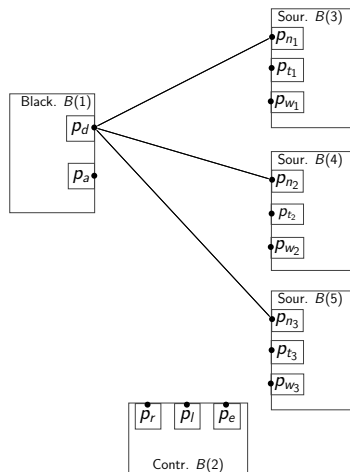
Example



$\{p_d, p_r\}$

How to encode interactions

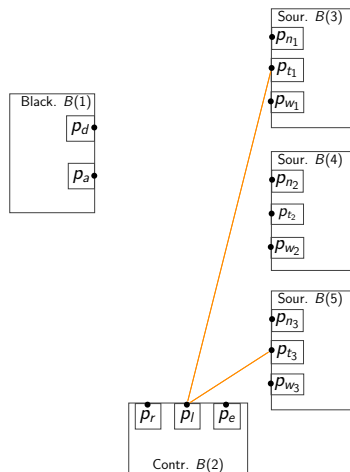
Example



$$\{p_d, p_r\} \mid \{p_d, p_{n_1}\}, \{p_d, p_{n_2}\}, \{p_d, p_{n_3}\}$$

How to encode interactions

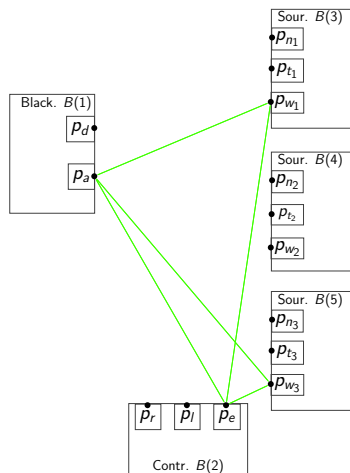
Example



$$\{p_d, p_r\} \mid \{p_d, p_{n_1}\}, \{p_d, p_{n_2}\}, \{p_d, p_{n_3}\} \mid \{p_l, p_{t_1}\}, \{p_l, p_{t_3}\}$$

How to encode interactions

Example



$$\{p_d, p_r\} \mid \{p_d, p_{n1}\}, \{p_d, p_{n2}\}, \{p_d, p_{n3}\} \mid \{p_l, p_{t1}\}, \{p_l, p_{t3}\} \mid \\ \{p_e, p_{w1}, p_a\}, \{p_e, p_{w3}, p_a\}$$

How to encode interactions

Example

$$\hookrightarrow \{p_d, p_r\} \mid \{p_d, p_{n_1}\}, \{p_d, p_{n_2}\}, \{p_d, p_{n_3}\} \mid \{p_l, p_{t_1}\}, \{p_l, p_{t_3}\} \mid \\ \{p_e, p_{w_1}, p_a\}, \{p_e, p_{w_3}, p_a\}$$

IDEA:

Encode interactions as **sequences**



Encode interactions as **letters of words of interactions**

- $w_1 = \{p_d, p_r\}\{p_d, p_{n_1}\}\{p_d, p_{n_2}\}\{p_d, p_{n_3}\}\{p_l, p_{t_1}\}\{p_l, p_{t_3}\} \\ \{p_e, p_{w_1}, p_a\}\{p_e, p_{w_3}, p_a\}$

How to encode interactions

Example

$$\hookrightarrow \{p_d, p_r\} \mid \{p_d, p_{n_1}\}, \{p_d, p_{n_2}\}, \{p_d, p_{n_3}\} \mid \{p_l, p_{t_1}\}, \{p_l, p_{t_3}\} \mid \{p_e, p_{w_1}, p_a\}, \{p_e, p_{w_3}, p_a\}$$

IDEA:

Encode interactions as **sequences**



Encode interactions as **letters of words of interactions**

- $w_1 = \{p_d, p_r\} \{p_d, p_{n_1}\} \{p_d, p_{n_2}\} \{p_d, p_{n_3}\} \{p_l, p_{t_1}\} \{p_l, p_{t_3}\} \{p_e, p_{w_1}, p_a\} \{p_e, p_{w_3}, p_a\}$
- $w_2 = \{p_d, p_r\} \{p_d, p_{n_2}\} \{p_d, p_{n_1}\} \{p_d, p_{n_3}\} \{p_l, p_{t_1}\} \{p_l, p_{t_2}\} \{p_e, p_{w_1}, p_a\} \{p_e, p_{w_2}, p_a\}$
- $w_2 = \{p_d, p_r\} \{p_d, p_{n_3}\} \{p_d, p_{n_1}\} \{p_d, p_{n_2}\} \{p_l, p_{t_1}\} \{p_l, p_{t_2}\} \{p_e, p_{w_1}, p_a\} \{p_e, p_{w_2}, p_a\}$

Propositional interaction logic

PIL

- Let P be a finite set of ports
- Let $I(P) = \mathcal{P}(P) \setminus \{\emptyset\}$ be the set of **interactions** over P

Propositional interaction logic

PIL

- Let P be a finite set of ports
- Let $I(P) = \mathcal{P}(P) \setminus \{\emptyset\}$ be the set of **interactions** over P

The syntax of PIL formulas ϕ over P is given by the grammar

$$\phi ::= \text{true} \mid p \mid \neg\phi \mid \phi \vee \phi$$

Propositional interaction logic

PIL

- Let P be a finite set of ports
- Let $I(P) = \mathcal{P}(P) \setminus \{\emptyset\}$ be the set of **interactions** over P

The syntax of PIL formulas ϕ over P is given by the grammar

$$\phi ::= \text{true} \mid p \mid \neg\phi \mid \phi \vee \phi$$

For every PIL formula ϕ and $a \in I(P)$ we define

- $a \models_{\text{PIL}} \text{true}$
- $a \models_{\text{PIL}} p$ iff $p \in a$
- $a \models_{\text{PIL}} \neg\phi$ iff $a \not\models_{\text{PIL}} \phi$
- $a \models_{\text{PIL}} \phi_1 \vee \phi_2$ iff $a \models_{\text{PIL}} \phi_1$ or $a \models_{\text{PIL}} \phi_2$

Extended propositional interaction logic

Syntax

- Let P be a finite set of ports

The syntax of EPIL formulas φ over P is given by the grammar

$$\zeta ::= \phi \mid \zeta * \zeta \mid \zeta \sqcup \zeta$$

$$\varphi ::= \zeta \mid \neg \zeta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi * \varphi \mid \varphi \sqcup \varphi \mid \varphi^+$$

- ϕ is a *PIL* formula over P ,
- $*$ is the concatenation operator,
- \sqcup is the shuffle operator, and
- $^+$ is the iteration operator

Extended propositional interaction logic

Notation

- Let $w_1, w_2 \in A$. The **concatenation** of w_1, w_2 is the word $w_1 \cdot w_2$

Example

- Let $A = \{a, b, c\}$, $w_1 = ab$, $w_2 = c$
- $w_1 \cdot w_2 = abc$

Extended propositional interaction logic

Notation

- Let $w_1, w_2 \in A$. The **concatenation** of w_1, w_2 is the word $w_1 \cdot w_2$

Example

- Let $A = \{a, b, c\}$, $w_1 = ab$, $w_2 = c$
- $w_1 \cdot w_2 = abc$

- Let $w_1, w_2 \in A$. The **shuffle** of w_1, w_2 is the language $w_1 \sqcup w_2 = \{w_1 u_1 \dots w_m u_m \mid w_1, \dots, w_m, u_1, \dots, u_m \in A^* \text{ and } w = w_1 \dots w_m, u = u_1 \dots u_m\}$

Example

- Let $A = \{a, b, c\}$, $w_1 = ab$, $w_2 = c$
- $w_1 \sqcup w_2 = \{abc, acb, cab\}$

Extended propositional interaction logic

Notation

- Let $w \in A$. The n -th iteration of w where $n \geq 1$ is defined inductively as follows:

$$w^1 = w, w^2 = ww, \dots, w^{n+1} = w^n w$$

- Let $w \in A$. The $+$ iteration of w is defined by $w^+ = \bigcup_{n \geq 1} w^n$

Example

- Let $A = \{a, b, c\}$, $w = ab$
- $w^2 = abab$
- $w^n = (ab)^n$
- $w^+ = \bigcup_{n \geq 1} (ab)^n$

The syntax of EPIL formulas φ over P is given by the grammar

$$\zeta ::= \phi \mid \zeta * \zeta \mid \zeta \sqcup \zeta$$

$$\varphi ::= \zeta \mid \neg \zeta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi * \varphi \mid \varphi \sqcup \varphi \mid \varphi^+$$

- Let $I(P) = \mathcal{P}(P) \setminus \{\emptyset\}$
- Let $I(P)^*$ be the set of words of interactions over $I(P)$
- Let $I(P)^+ : I(P)^* \setminus \{\epsilon\}$
- For $\nu \geq 1$ we define the EPIL formula φ^ν by induction on ν by:
 - $\varphi^1 = \varphi$,
 - $\varphi^{\nu+1} = \varphi^\nu * \varphi$.

The syntax of EPIL formulas φ over P is given by the grammar

$$\zeta ::= \phi \mid \zeta * \zeta \mid \zeta \sqcup \zeta$$

$$\varphi ::= \zeta \mid \neg \zeta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi * \varphi \mid \varphi \sqcup \varphi \mid \varphi^+$$

- Let φ be an EPIL formula over P and $w \in I(P)^+$.
- Then we define the satisfaction relation $w \models \varphi$ by induction on the structure of φ as follows:

Extended propositional interaction logic

Semantics

- $w \models \phi$ iff $w \models_{\text{PIL}} \phi$ hence w consists of a single letter
- $w \models \zeta_1 * \zeta_2$ iff there exist $w_1, w_2 \in I(P)^+$ s.t. $w = w_1 w_2$, $w_1 \models \zeta_1$, and $w_2 \models \zeta_2$
- $w \models \zeta_1 \sqcup \zeta_2$ iff there exist $w_1, w_2 \in I(P)^+$ such that $w \in w_1 \sqcup w_2$ and $w_i \models \zeta_i$ for $i = 1, 2$
- $w \models \neg \zeta$ iff $w \not\models \zeta$
- $w \models \varphi_1 \vee \varphi_2$ iff $w \models \varphi_1$ or $w \models \varphi_2$
- $w \models \varphi_1 \wedge \varphi_2$ iff $w \models \varphi_1$ and $w \models \varphi_2$
- $w \models \varphi_1 * \varphi_2$ iff there exist $w_1, w_2 \in I(P)^+$ s.t. $w = w_1 w_2$, $w_1 \models \varphi_1$, and $w_2 \models \varphi_2$
- $w \models \varphi_1 \sqcup \varphi_2$ iff there exist $w_1, w_2 \in I(P)^+$ s.t. $w \in w_1 \sqcup w_2$, $w_1 \models \varphi_1$, and $w_2 \models \varphi_2$
- $w \models \varphi^+$ iff there exists $\nu \geq 1$ such that $w \models \varphi^\nu$.

Examples of EPIL formulas

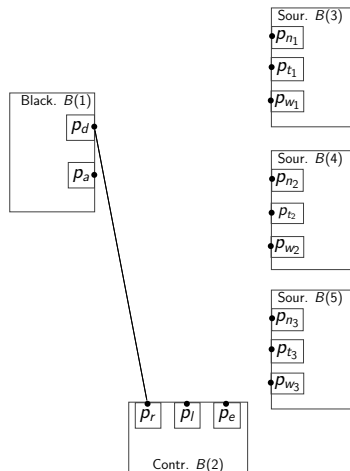
Notation

- Let $P = \{p_1, \dots, p_n\}$ be a set of ports
- Let $p_{i_1}, \dots, p_{i_m} \in P$ with $m < n$

$$\#(p_{i_1} \wedge \dots \wedge p_{i_m}) ::= p_{i_1} \wedge \dots \wedge p_{i_m} \wedge \bigwedge_{p \in P \setminus \{p_{i_1}, \dots, p_{i_m}\}} \neg p$$

Examples of EPIL formulas

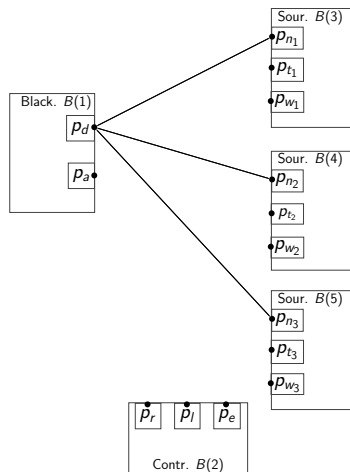
Blackboard



$$\varphi = \#(p_d \wedge p_r)^*$$

Examples of EPIL formulas

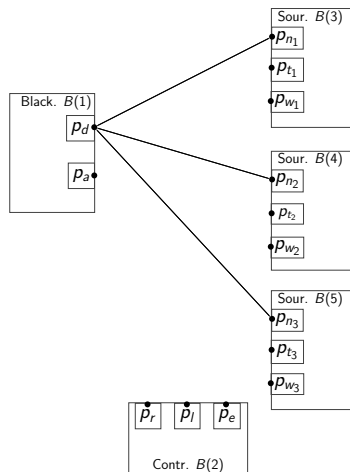
Blackboard



$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) ? (\#(p_d \wedge p_{n_2})) ? (\#(p_d \wedge p_{n_3})) \right) *$$

Examples of EPIL formulas

Blackboard

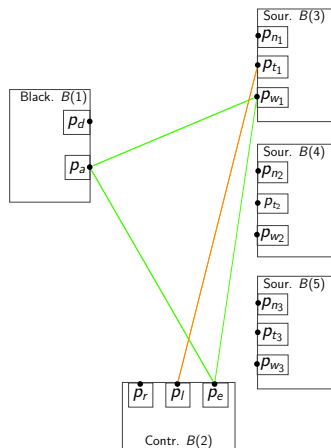


$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup \#(p_d \wedge p_{n_2}) \sqcup \#(p_d \wedge p_{n_3}) \right) *$$

Examples of EPIL formulas

Blackboard

$$\varphi_1 = \#(p_l \wedge p_{t_1}) * \#(p_e \wedge p_{w_1} \wedge p_a)$$

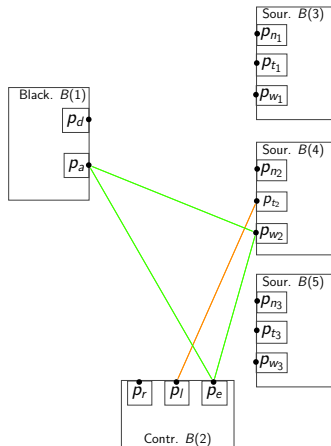


$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup \#(p_d \wedge p_{n_2}) \sqcup \#(p_d \wedge p_{n_3}) \right) * \left(\varphi_1 \right)$$

Examples of EPIL formulas

Blackboard

$$\varphi_2 = \#(p_l \wedge p_{t_2}) * \#(p_e \wedge p_{w_2} \wedge p_a)$$

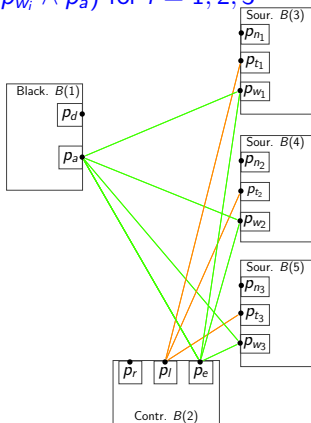


$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup \#(p_d \wedge p_{n_2}) \sqcup \#(p_d \wedge p_{n_3}) \right) * \left(\varphi_1 \vee \varphi_2 \vee \dots \right)$$

Examples of EPIL formulas

Blackboard

$\varphi_i = \#(p_l \wedge p_{t_i}) * \#(p_e \wedge p_{w_i} \wedge p_a)$ for $i = 1, 2, 3$



$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup \#(p_d \wedge p_{n_2}) \sqcup \#(p_d \wedge p_{n_3}) \right) * \left(\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee (\varphi_1 \sqcup \varphi_2) \vee (\varphi_1 \sqcup \varphi_3) \vee (\varphi_2 \sqcup \varphi_3) \vee (\varphi_1 \sqcup \varphi_2 \sqcup \varphi_3) \right)$$

Examples of EPIL formulas

Blackboard

- $\varphi_i = \#(p_l \wedge p_{t_i}) * \#(p_e \wedge p_{w_i} \wedge p_a)$ for $i = 1, 2, 3$

- EPIL formula for Blackboard architecture

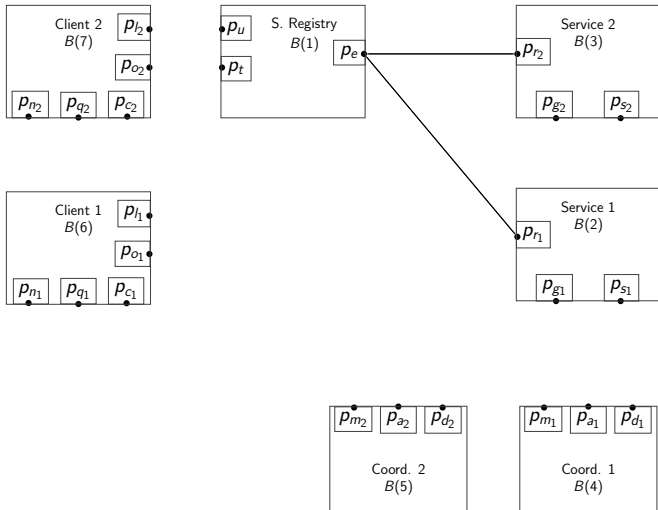
$$\varphi = \#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup (\#(p_d \wedge p_{n_2})) \sqcup (\#(p_d \wedge p_{n_3})) \right) * \left(\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee (\varphi_1 \sqcup \varphi_2) \vee (\varphi_1 \sqcup \varphi_3) \vee (\varphi_2 \sqcup \varphi_3) \vee (\varphi_1 \sqcup \varphi_2 \sqcup \varphi_3) \right)$$

- Application of iteration operator for recursive interactions

$$\varphi = \left(\#(p_d \wedge p_r) * \left(\#(p_d \wedge p_{n_1}) \sqcup (\#(p_d \wedge p_{n_2})) \sqcup (\#(p_d \wedge p_{n_3})) \right) * \left(\varphi_1 \vee \varphi_2 \vee \varphi_3 \vee (\varphi_1 \sqcup \varphi_2) \vee (\varphi_1 \sqcup \varphi_3) \vee (\varphi_2 \sqcup \varphi_3) \vee (\varphi_1 \sqcup \varphi_2 \sqcup \varphi_3) \right) \right)^+ \right)^+$$

Examples of EPIL formulas

Request/Response



Examples of EPIL formulas

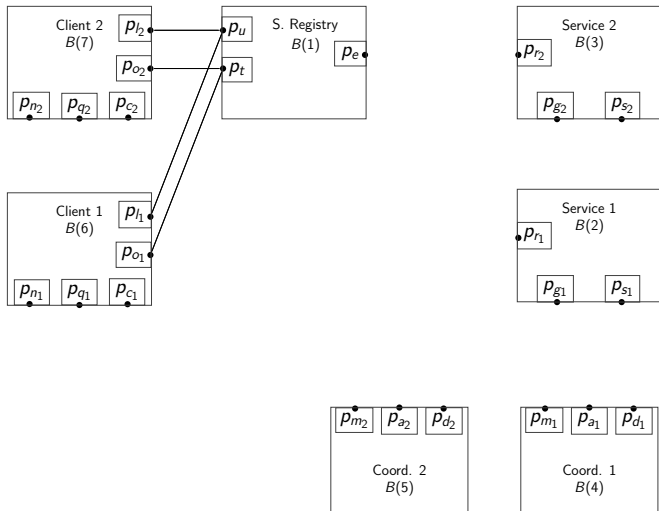
Request/Response

Connections of 1st and 2nd service with the service registry

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2}))^*$$

Examples of EPIL formulas

Request/Response



Examples of EPIL formulas

Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2}))^*$$

Connections of 1st and 2nd client with the service registry

- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$
- $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$

Examples of EPIL formulas

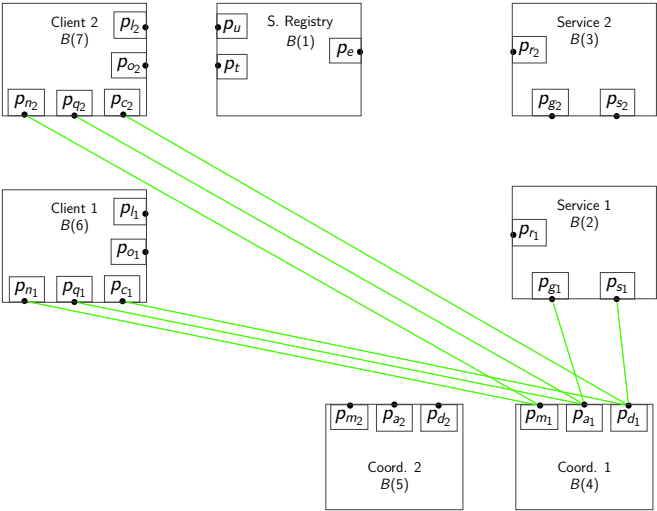
Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2)*$$

- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$ 1st client with s.registry
- $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$ 2nd client with s.registry

Examples of EPIL formulas

Request/Response



Examples of EPIL formulas

Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left(\left((\varphi_{11} \right. \right.$$

- $\xi_1 = \#(p_{h_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$, $\xi_2 = \#(p_{h_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$

Connections of 1st client with first coordinator and service

- $\varphi_{11} = \#(p_{n_1} \wedge p_{m_1}) * (\#p_{q_1} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_1} \wedge p_{d_1} \wedge p_{s_1})$

Examples of EPIL formulas

Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left((\varphi_{11} \vee \varphi_{21}) \right)$$

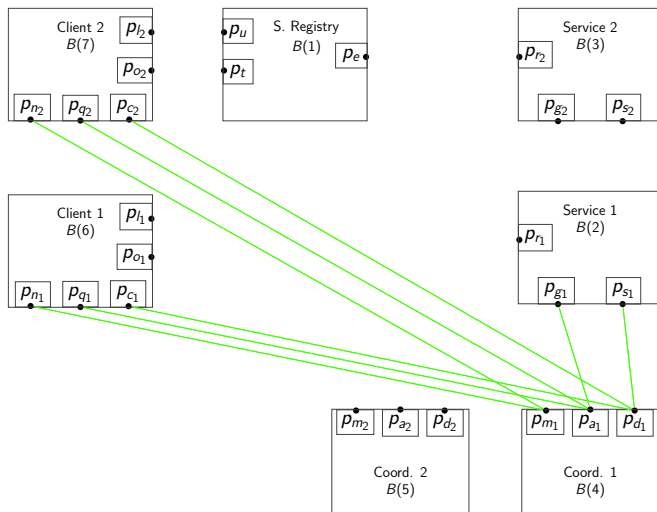
- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$, $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$
- $\varphi_{11} = \#(p_{n_1} \wedge p_{m_1}) * (\#p_{q_1} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_1} \wedge p_{d_1} \wedge p_{s_1})$

Connections of 2nd client with first coordinator and service

- $\varphi_{21} = \#(p_{n_2} \wedge p_{m_1}) * \#(p_{q_2} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_2} \wedge p_{d_1} \wedge p_{s_1})$

Architectures with order restrictions

Request/Response



Examples of EPIL formulas

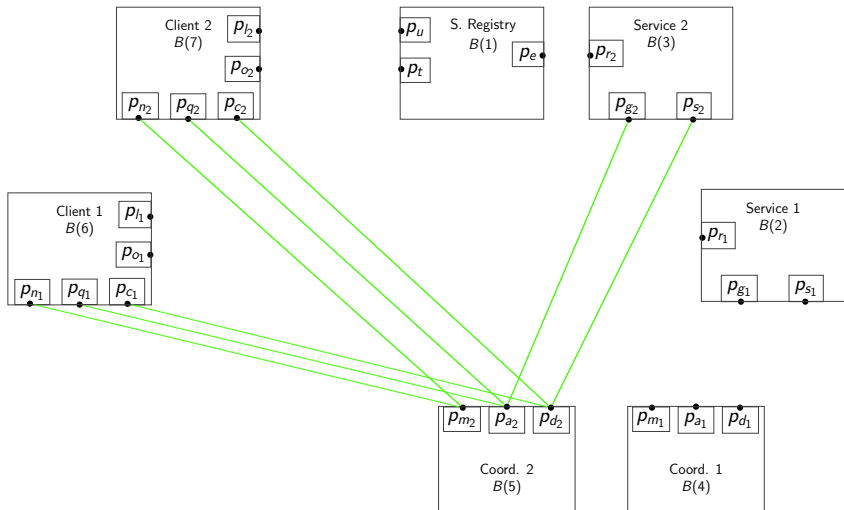
Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left(\left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \right) \vee \right)$$

- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$, $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$
- $\varphi_{11} = \#(p_{n_1} \wedge p_{m_1}) * \#(p_{q_1} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_1} \wedge p_{d_1} \wedge p_{s_1})$
- $\varphi_{21} = \#(p_{n_2} \wedge p_{m_1}) * \#(p_{q_2} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_2} \wedge p_{d_1} \wedge p_{s_1})$

Examples of EPIL formulas

Request/Response



Examples of EPIL formulas

Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left(\left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \vee \left(\varphi_{12} \vee \varphi_{22} \vee (\varphi_{12} * \varphi_{22}) \vee (\varphi_{22} * \varphi_{12}) \right) \vee \right)$$

Connections of 1st and 2nd client with 2nd coordinator and service

- $\varphi_{12} = \#(p_{n_1} \wedge p_{m_2}) * \#(p_{q_1} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_1} \wedge p_{d_2} \wedge p_{s_2})$
- $\varphi_{22} = \#(p_{n_2} \wedge p_{m_2}) * \#(p_{q_2} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_2} \wedge p_{d_2} \wedge p_{s_2})$

Examples of EPIL formulas

Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left(\left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \vee \left(\varphi_{12} \vee \varphi_{22} \vee (\varphi_{12} * \varphi_{22}) \vee (\varphi_{22} * \varphi_{12}) \right) \vee \left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \sqcup (\varphi_{12} \vee \varphi_{22} \vee (\varphi_{12} * \varphi_{22}) \vee (\varphi_{22} * \varphi_{12}) \right) \right) \right)$$

- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$, $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$
- $\varphi_{11} = \#(p_{n_1} \wedge p_{m_1}) * \#(p_{q_1} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_1} \wedge p_{d_1} \wedge p_{s_1})$
- $\varphi_{21} = \#(p_{n_2} \wedge p_{m_1}) * \#(p_{q_2} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_2} \wedge p_{d_1} \wedge p_{s_1})$
- $\varphi_{12} = \#(p_{n_1} \wedge p_{m_2}) * \#(p_{q_1} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_1} \wedge p_{d_2} \wedge p_{s_2})$
- $\varphi_{22} = \#(p_{n_2} \wedge p_{m_2}) * \#(p_{q_2} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_2} \wedge p_{d_2} \wedge p_{s_2})$

Examples of EPIL formulas

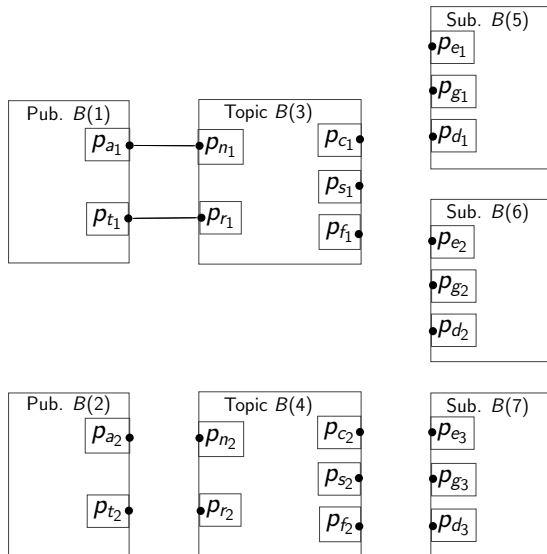
Request/Response

$$\varphi = (\#(p_e \wedge p_{r_1}) \sqcup \#(p_e \wedge p_{r_2})) * (\xi_1 \sqcup \xi_2) * \left(\left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \right)^+ \vee \left((\varphi_{12} \vee \varphi_{22} \vee (\varphi_{12} * \varphi_{22}) \vee (\varphi_{22} * \varphi_{12})) \right)^+ \vee \left((\varphi_{11} \vee \varphi_{21} \vee (\varphi_{11} * \varphi_{21}) \vee (\varphi_{21} * \varphi_{11})) \right)^+ \sqcup \left((\varphi_{12} \vee \varphi_{22} \vee (\varphi_{12} * \varphi_{22}) \vee (\varphi_{22} * \varphi_{12})) \right)^+ \right)^+$$

- $\xi_1 = \#(p_{l_1} \wedge p_u) * \#(p_{o_1} \wedge p_t)$, $\xi_2 = \#(p_{l_2} \wedge p_u) * \#(p_{o_2} \wedge p_t)$
- $\varphi_{11} = \#(p_{n_1} \wedge p_{m_1}) * \#(p_{q_1} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_1} \wedge p_{d_1} \wedge p_{s_1})$
- $\varphi_{21} = \#(p_{n_2} \wedge p_{m_1}) * \#(p_{q_2} \wedge p_{a_1} \wedge p_{g_1}) * \#(p_{c_2} \wedge p_{d_1} \wedge p_{s_1})$
- $\varphi_{12} = \#(p_{n_1} \wedge p_{m_2}) * \#(p_{q_1} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_1} \wedge p_{d_2} \wedge p_{s_2})$
- $\varphi_{22} = \#(p_{n_2} \wedge p_{m_2}) * \#(p_{q_2} \wedge p_{a_2} \wedge p_{g_2}) * \#(p_{c_2} \wedge p_{d_2} \wedge p_{s_2})$

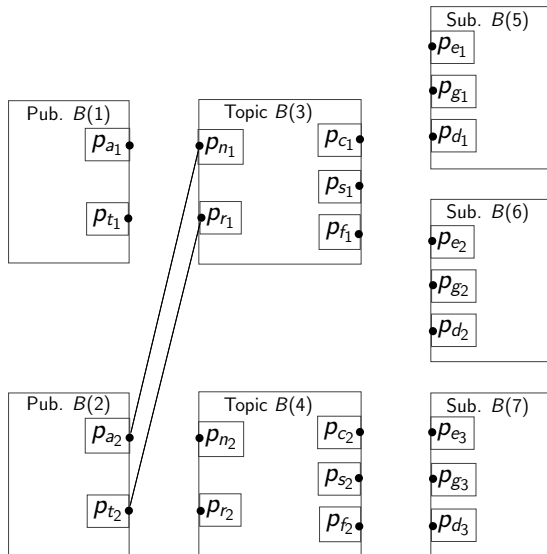
Examples of EPIL formulas

Publish/Subscribe



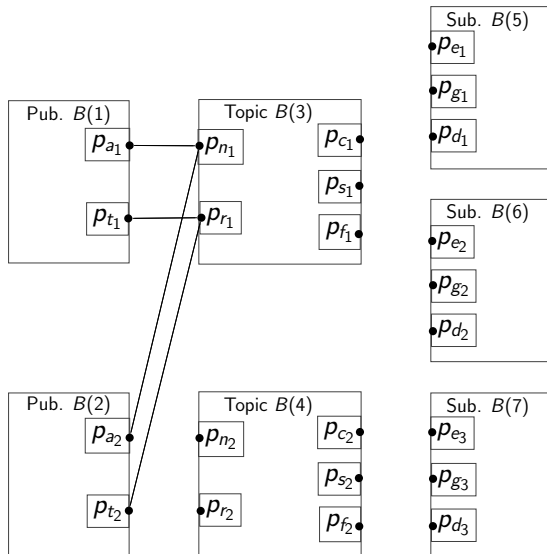
Examples of EPIL formulas

Publish/Subscribe



Examples of EPIL formulas

Publish/Subscribe



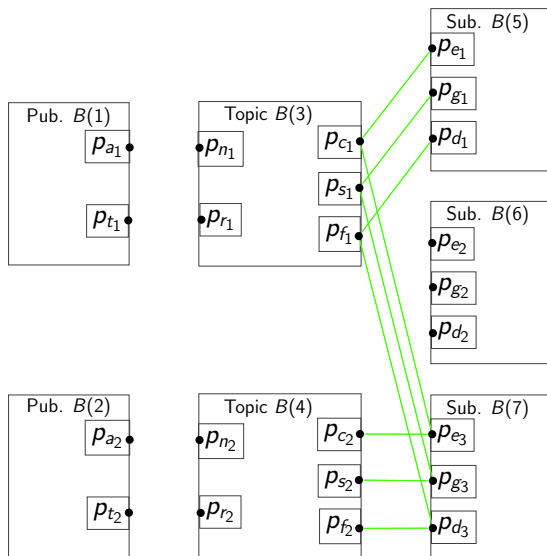
Connections of 1st topic with publishers

$$\xi_1 = \xi_{11} \vee \xi_{12} \vee (\xi_{11} \sqcup \xi_{12})$$

- $\xi_{11} = \#(p_{n_1} \wedge p_{a_1}) * \#(p_{r_1} \wedge p_{t_1})$
- $\xi_{12} = \#(p_{n_1} \wedge p_{a_2}) * \#(p_{r_1} \wedge p_{t_2})$

Examples of EPIL formulas

Publish/Subscribe



Examples of EPIL formulas

Publish/Subscribe

- $\xi_1 = \xi_{11} \vee \xi_{12} \vee (\xi_{11} \sqcup \xi_{12})$ connections of 1st topic with publishers

Connections for 1st topic with publishers and 1st, 2nd, 3rd subscriber

$$\varphi_1 = \left((\xi_1 * \varphi_{11}) \vee (\xi_1 * \varphi_{12}) \vee (\xi_1 * \varphi_{13}) \vee \right.$$

$$- \varphi_{11} = \#(p_{c_1} \wedge p_{e_1}) * \#(p_{s_1} \wedge p_{g_1}) * \#(p_{f_1} \wedge p_{d_1})$$

$$- \varphi_{12} = \#(p_{c_1} \wedge p_{e_2}) * \#(p_{s_1} \wedge p_{g_2}) * \#(p_{f_1} \wedge p_{d_2})$$

$$- \varphi_{13} = \#(p_{c_1} \wedge p_{e_3}) * \#(p_{s_1} \wedge p_{g_3}) * \#(p_{f_1} \wedge p_{d_3})$$

Examples of EPIL formulas

Publish/Subscribe

$$\varphi_1 = \left(\begin{aligned} &((\xi_1 * \varphi_{11}) \vee (\xi_1 * \varphi_{12}) \vee (\xi_1 * \varphi_{13}) \vee \\ &(\xi_1 * (\varphi_{11} \sqcup \varphi_{12})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{13})) \vee \\ &(\xi_1 * (\varphi_{12} \sqcup \varphi_{13})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{12} \sqcup \varphi_{13})) \end{aligned} \right)$$

$$- \varphi_{11} = \#(p_{c_1} \wedge p_{e_1}) * \#(p_{s_1} \wedge p_{g_1}) * \#(p_{f_1} \wedge p_{d_1})$$

$$- \varphi_{12} = \#(p_{c_1} \wedge p_{e_2}) * \#(p_{s_1} \wedge p_{g_2}) * \#(p_{f_1} \wedge p_{d_2})$$

$$- \varphi_{13} = \#(p_{c_1} \wedge p_{e_3}) * \#(p_{s_1} \wedge p_{g_3}) * \#(p_{f_1} \wedge p_{d_3})$$

Examples of EPIL formulas

Publish/Subscribe

$$\varphi_1 = \left(\begin{aligned} &((\xi_1 * \varphi_{11}) \vee (\xi_1 * \varphi_{12}) \vee (\xi_1 * \varphi_{13})) \vee \\ &(\xi_1 * (\varphi_{11} \sqcup \varphi_{12})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{13})) \vee \\ &(\xi_1 * (\varphi_{12} \sqcup \varphi_{13})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{12} \sqcup \varphi_{13})) \end{aligned} \right)$$

$$\varphi_2 = \left(\begin{aligned} &(\xi_2 * \varphi_{21}) \vee (\xi_2 * \varphi_{22}) \vee (\xi_2 * \varphi_{23}) \vee \\ &(\xi_2 * (\varphi_{21} \sqcup \varphi_{22})) \vee (\xi_2 * (\varphi_{21} \sqcup \varphi_{23})) \vee \\ &(\xi_2 * (\varphi_{22} \sqcup \varphi_{23})) \vee (\xi_2 * (\varphi_{21} \sqcup \varphi_{22} \sqcup \varphi_{23})) \end{aligned} \right)$$

Examples of EPIL formulas

Publish/Subscribe

The EPIL formula φ for the Publish/Subscribe architecture is

$$\varphi = (\varphi_1 \vee \varphi_2 \vee (\varphi_1 \sqcup \varphi_2))^+$$

$$\begin{aligned} \varphi_1 = & \left((\xi_1 * \varphi_{11}) \vee (\xi_1 * \varphi_{12}) \vee (\xi_1 * \varphi_{13}) \vee \right. \\ & (\xi_1 * (\varphi_{11} \sqcup \varphi_{12})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{13})) \vee \\ & \left. (\xi_1 * (\varphi_{12} \sqcup \varphi_{13})) \vee (\xi_1 * (\varphi_{11} \sqcup \varphi_{12} \sqcup \varphi_{13})) \right) \end{aligned}$$

$$\begin{aligned} \varphi_2 = & \left((\xi_2 * \varphi_{21}) \vee (\xi_2 * \varphi_{22}) \vee (\xi_2 * \varphi_{23}) \vee \right. \\ & (\xi_2 * (\varphi_{21} \sqcup \varphi_{22})) \vee (\xi_2 * (\varphi_{21} \sqcup \varphi_{23})) \vee \\ & \left. (\xi_2 * (\varphi_{22} \sqcup \varphi_{23})) \vee (\xi_2 * (\varphi_{21} \sqcup \varphi_{22} \sqcup \varphi_{23})) \right) \end{aligned}$$

- First-order level of EPIL for parametric architectures
- Weighted EPIL and weighted first-order level of EPIL over a commutative semiring
- Fuzzy EPIL and first-order level of fuzzy EPIL to address uncertainty in (parametric) architectures (submitted)

- Second-order level of our logics (in unweighted, weighted, fuzzy setup) for modelling architectures like Ring, Pipeline, Grid (work in progress)
- Logic-basic description of dynamic (parametric) architectures
- Architecture-oriented verification for (parametric) systems
- Implementation of the logic-based framework for automating the specification and identification of (parametric) architectures

Thank you!