Congruences for Contextual Graph-Rewriting

Vladimiro Sassone, Pawel Sobocinski

Dagstuhl, June 7, 2004

Plan of the talk

1. Deriving bisimulation congruences
2. Cospans as generalised contexts
3. Bisimulation for graph rewriting

Deriving Congruences

- Many syntactic formalisms for concurrency and mobility
- Unification efforts:

1. Milner et al '90s-now: action calculi, bigraphs
2. Montanari et al '90s-now: tile systems.
3. Sewell, Leifer, Milner, Sassone and Sobocinski: meta theory of process calculi

Labels in LTS

- *Slogan:* Labels should be smallest contexts which allow reaction/interaction
  - eg. simple CCS-style calculus $a.P + P' \xrightarrow{-|\bar{a}} P$
  - *Sewell (1998)*: Detailed **syntactic analysis** of simplified process calculi
  - *Leifer and Milner (2000)*: General notion of smallest context - the **relative pushout**.
  - *Sassone and Sobocinski (2002)*: 2-categorical generalisation to allow handling of **structural congruences**.

Reactive Systems

- A reactive system
  - objects = typed *"holes"*
  - arrows = contexts
  - 2-cells = *"structural congruence"*

$$a \longrightarrow\!\!\!\!\!\triangleright\; b$$

if there exists

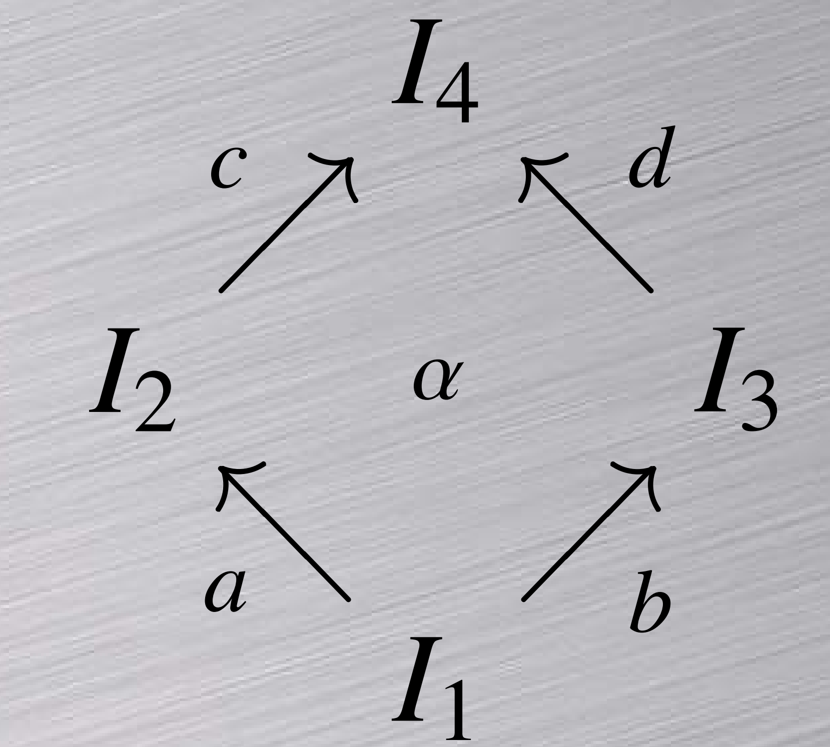
$$\langle l, r \rangle,\; d \in \mathbb{D}$$

and

$$\rho : dl \Rightarrow a,\; \rho' : b \Rightarrow dr$$

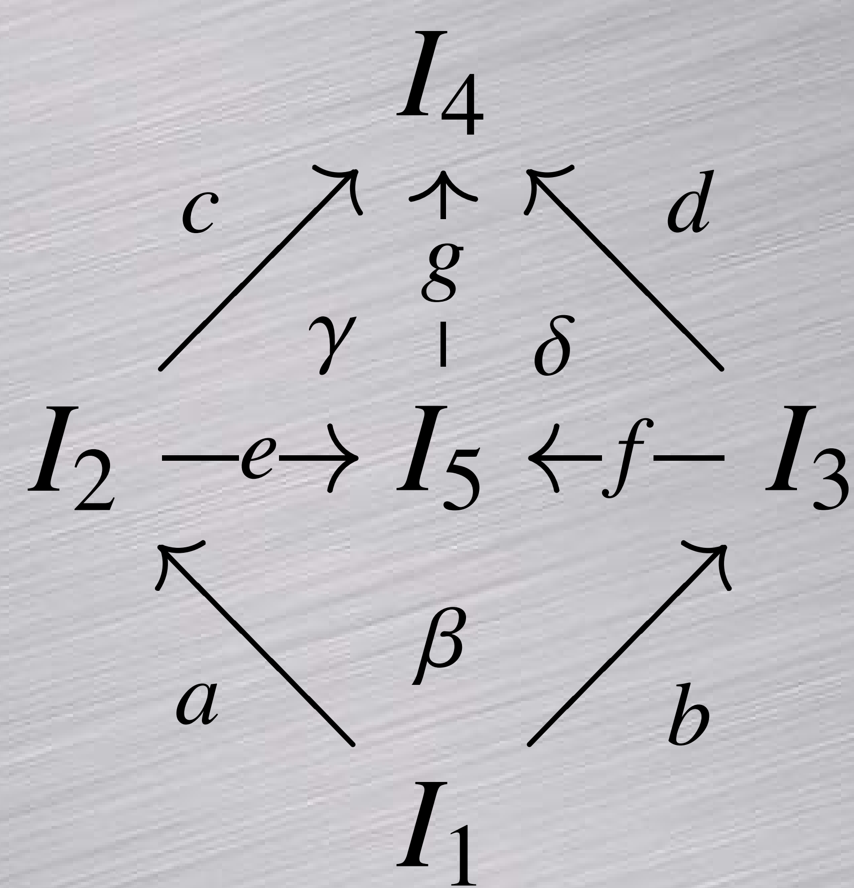
GRPOs

- Given $\alpha\colon ca \Rightarrow db$



$$\langle I_5, e, f, g, \beta, \gamma, \delta \rangle$$

$$\delta b \bullet g\beta \bullet \gamma a = \alpha$$

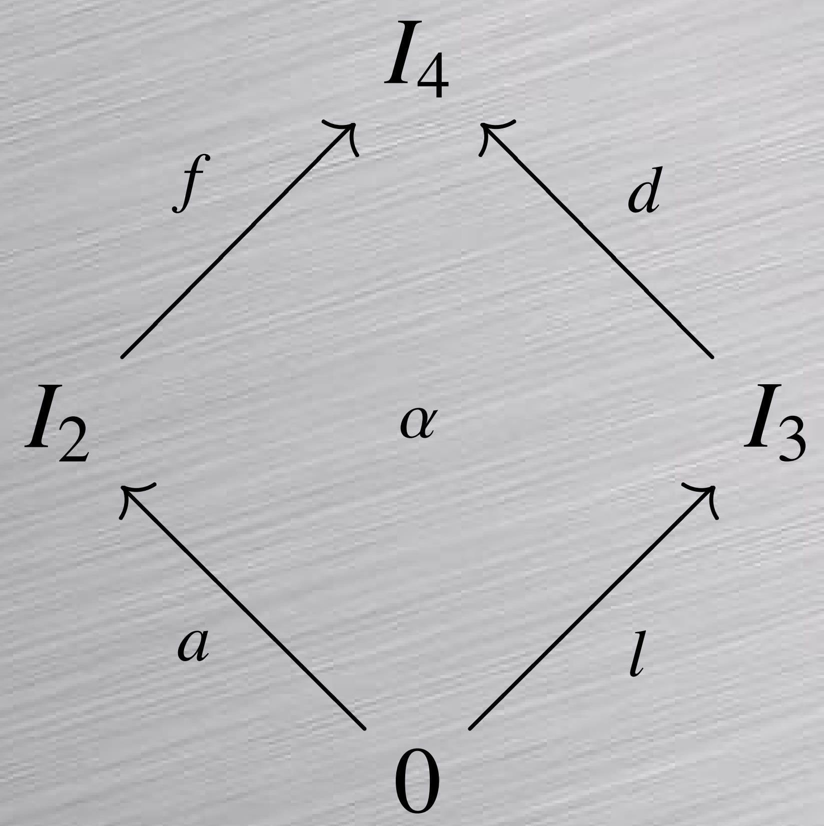


LTS

- Nodes: $[a] : 0 \to I_1$
- Labels: $[a] \xrightarrow{[f]} [a']$

$$\exists \langle l, r \rangle \in \mathcal{R} \quad \exists f \in \mathbb{C} \quad \exists d \in \mathbb{D} \quad \exists \alpha : fa \Rightarrow dl \quad \exists \alpha' : dr \Rightarrow a'$$

and



is a GRPO wrt itself

Properties of LTS

- *Bisimulation* is a congruence
- *Trace equivalence* is a congruence
- *Failures equivalence* is a congruence

What's the point?

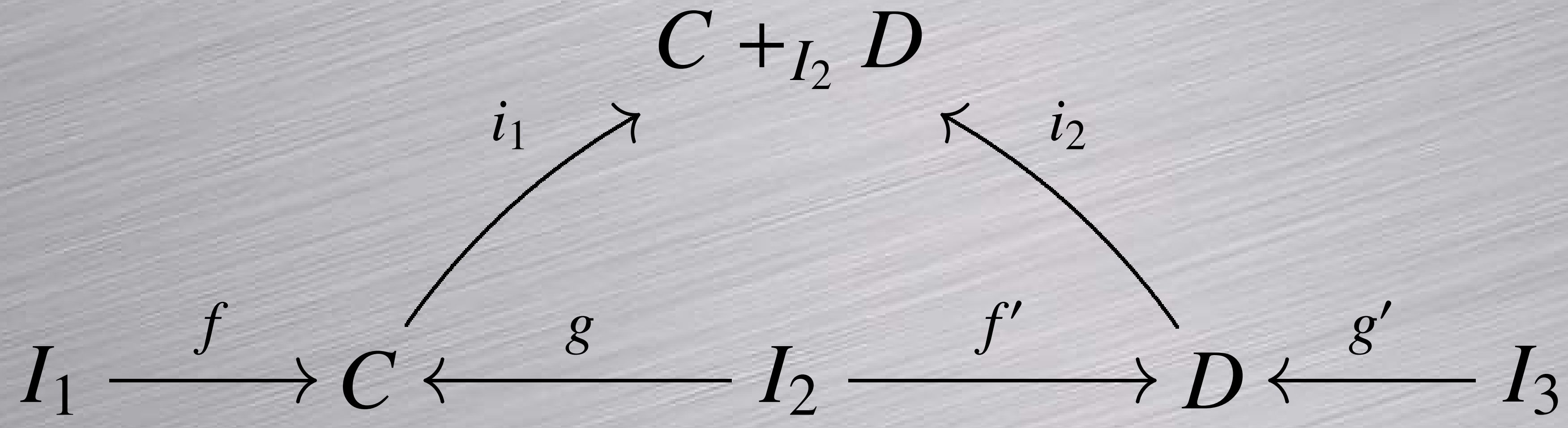
- Why am I telling you all this??

Cospan Bicategories

- Given **C**, **Cospan**(**C**) has
  - Objects: those of **C**
  - Arrows: cospans $$I_1 \xrightarrow{f} C \xleftarrow{g} I_2$$
  - 2-cells: cospan "homorphisms"
  - Composition by pushout along common interfaces.
  - intuitively: category of contexts over **C**.

Composition

- Identities: $I_1 \xrightarrow{id} I_1 \xleftarrow{id} I_1$
- Composition by pushout



$$a : (C +_{I_2} D) +_{I_3} E \to C +_{I_2} (D +_{I_3} E)$$

$$e_l : (I_1 +_{I_1} C) \to C$$

$$e_r : (C +_{I_2} I_2) \to C$$

satisfying coherence

Cospans on Graphs

- What is this when **C** is **Graphs**?

Desiderata

- For a suitable, general class of categories **C**, **Cospan**(**C**) has redex-GRPOs.
- Would allow to derive a coinduction principle for each “category of contexts” over a suitable **C**.

Adhesive Categories

- What is an adhesive category?

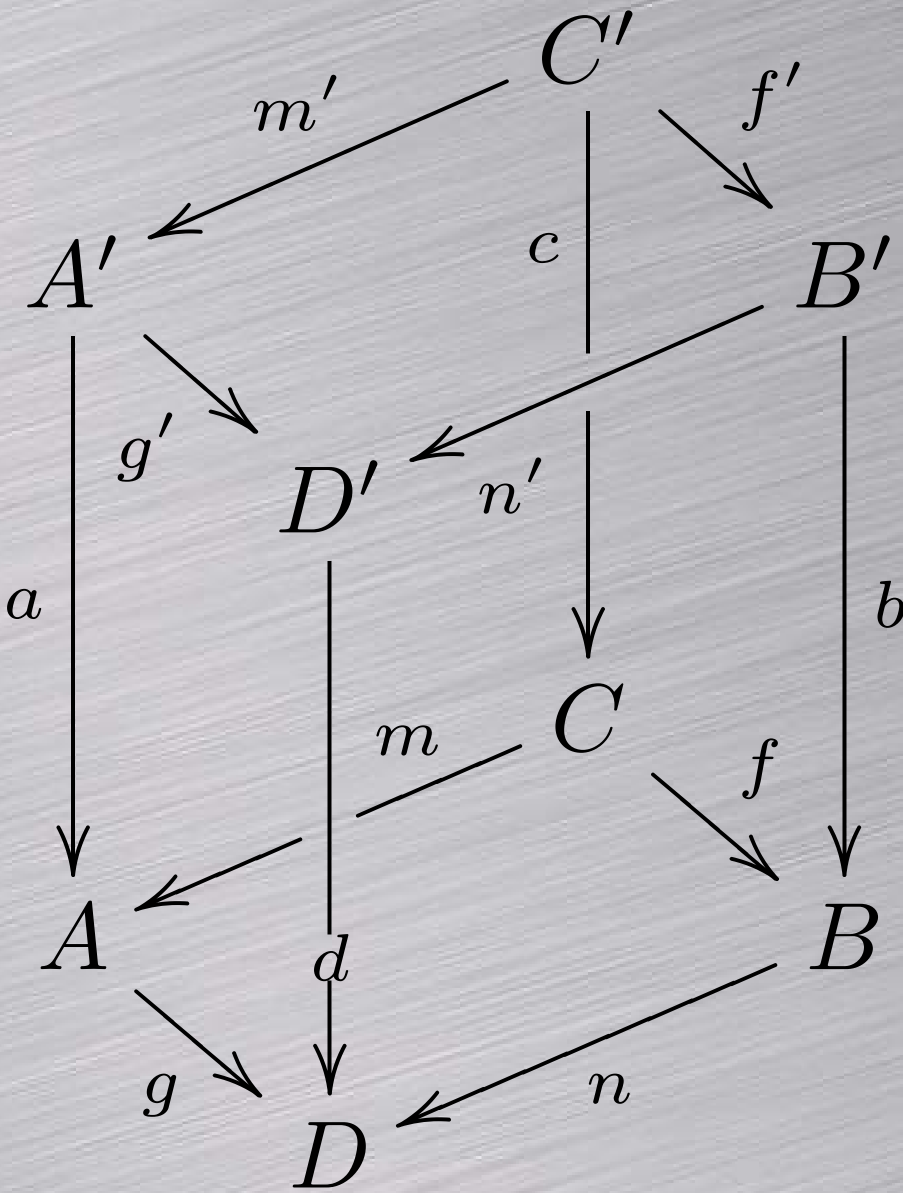
Adhesive Categories

- A category **C** is adhesive when

1. It has pushouts along monos
2. It has pullbacks
3. pushouts along monos are VK squares

Van Kampen Square

- Given a cube with back faces pullbacks:
- top face pushout iff front faces pullbacks



Graphs is Adhesive

- You didn't expect otherwise, did you??

Left-Linear Cospans

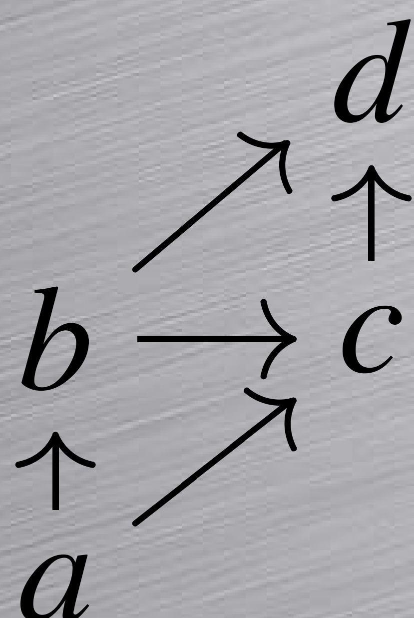
- When **C** is adhesive **LLC**(**C**) is the bicategory
  - objects as in **C**
  - arrows cospans $I_1 \overset{m}{\rightarrowtail} C \overset{g}{\longleftarrow} I_2$

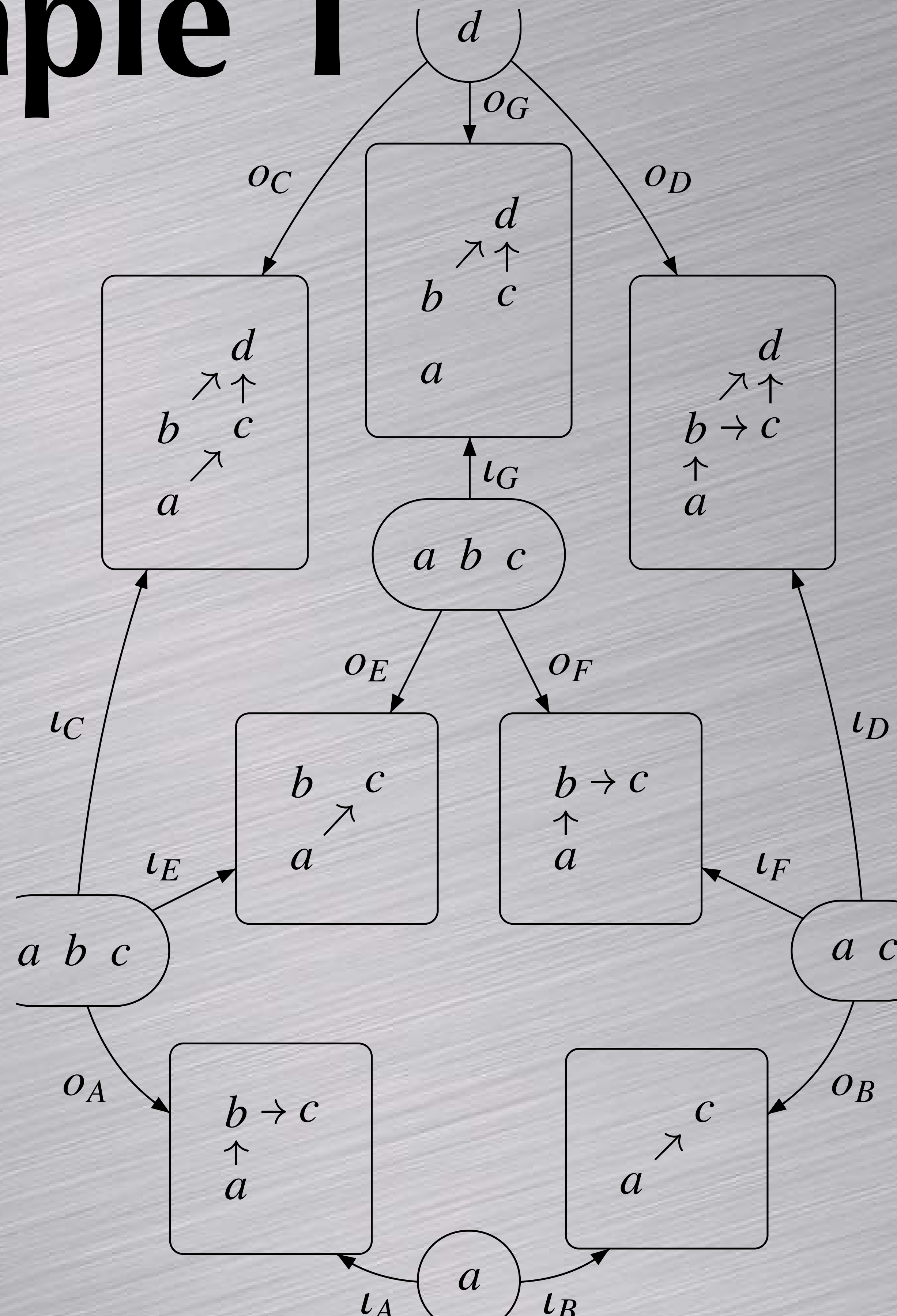
GRPOs for cospans

- ***Theorem***: Suppose that **C** is an adhesive category.
- Then, **LLC**(**C**) has redex-GRPOs.

Example 1

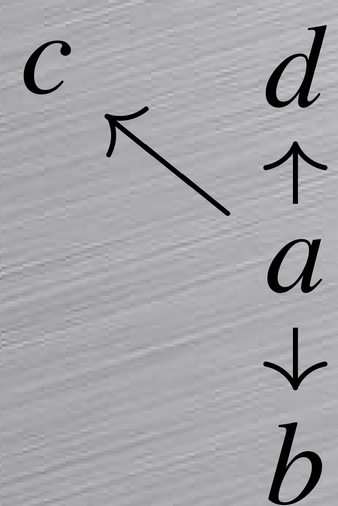
- All morphisms mono

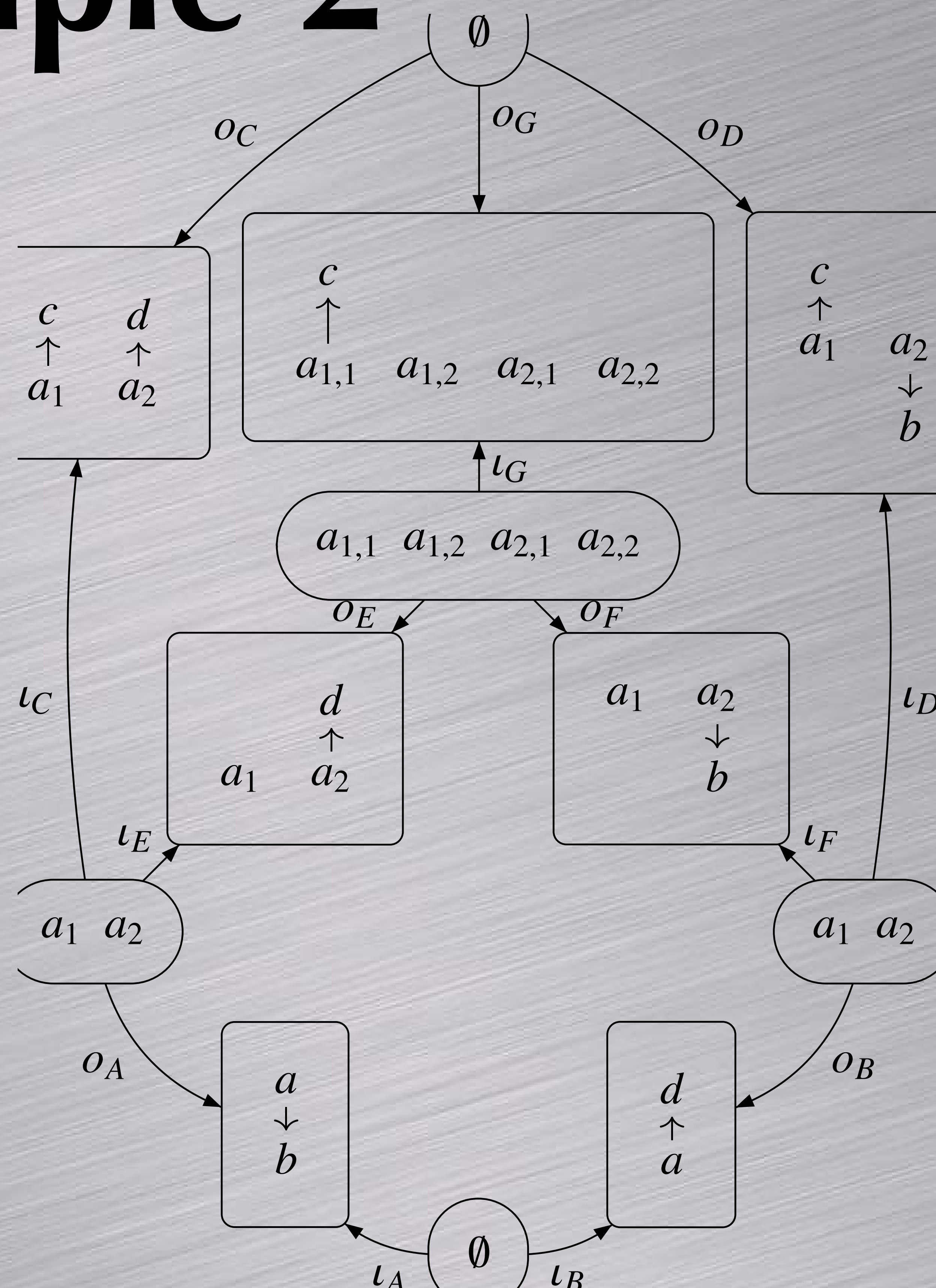




Example 2

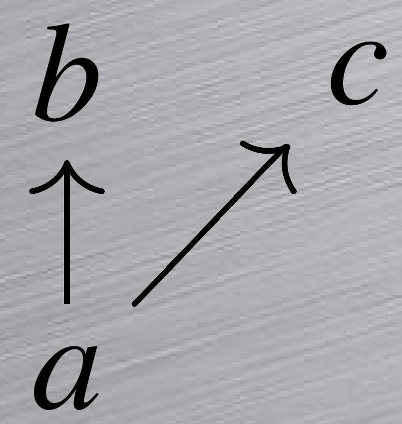
- o_A and o_L not mono

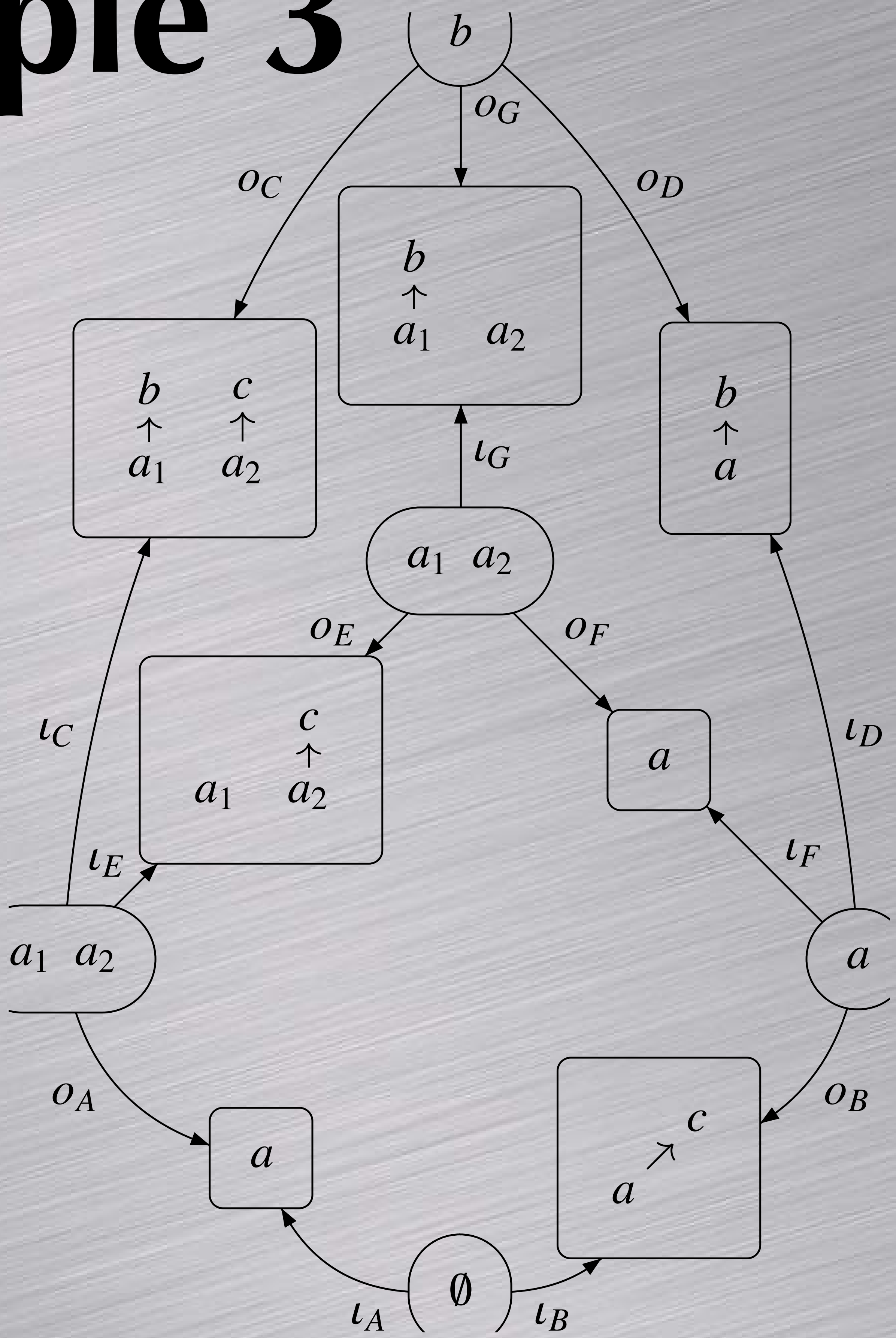




Example 3

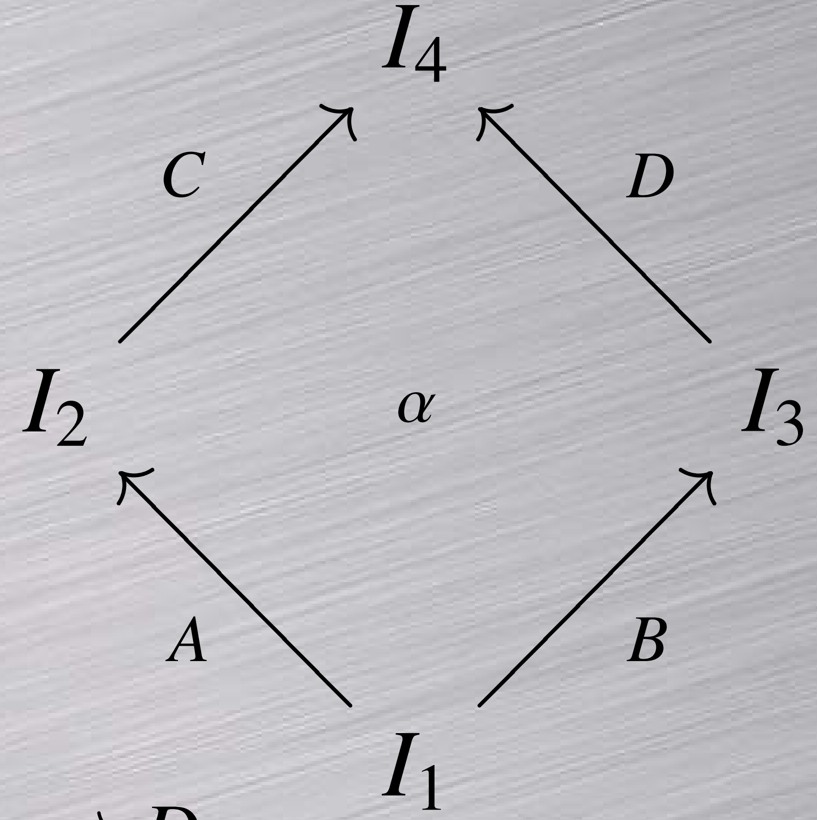
- o_A not mono

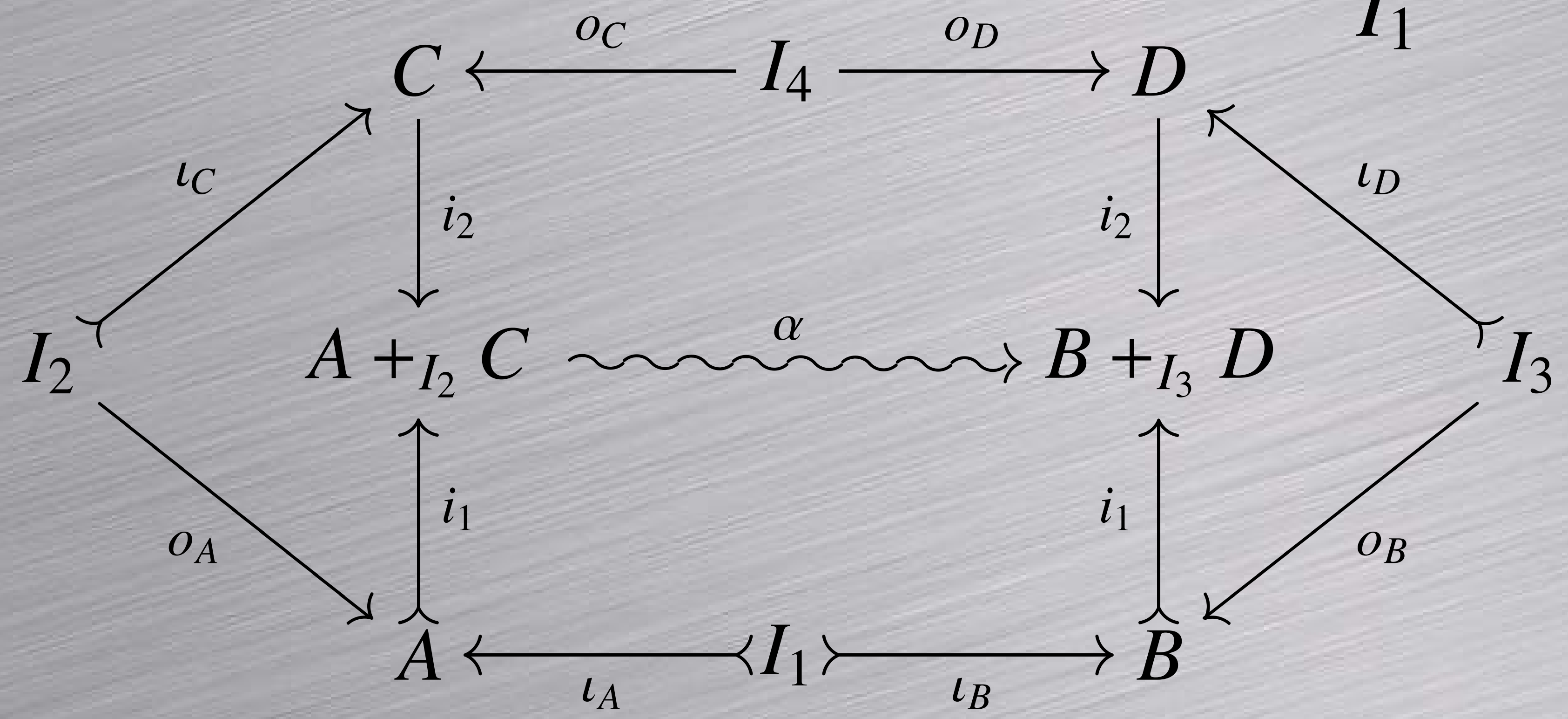




GRPOs in LLC(C)

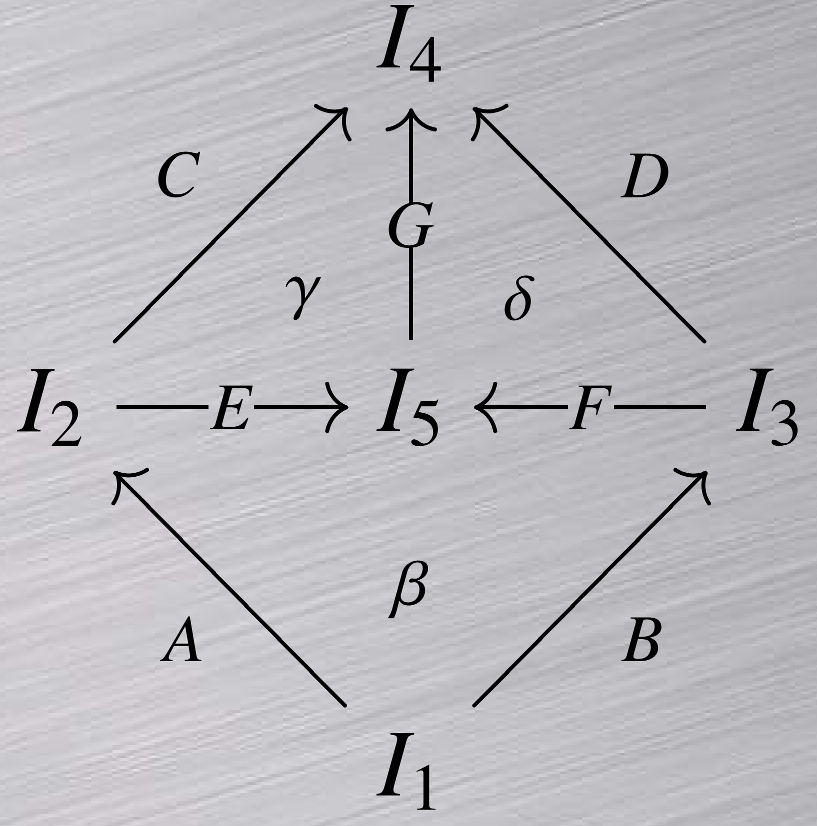
- Given redex square...

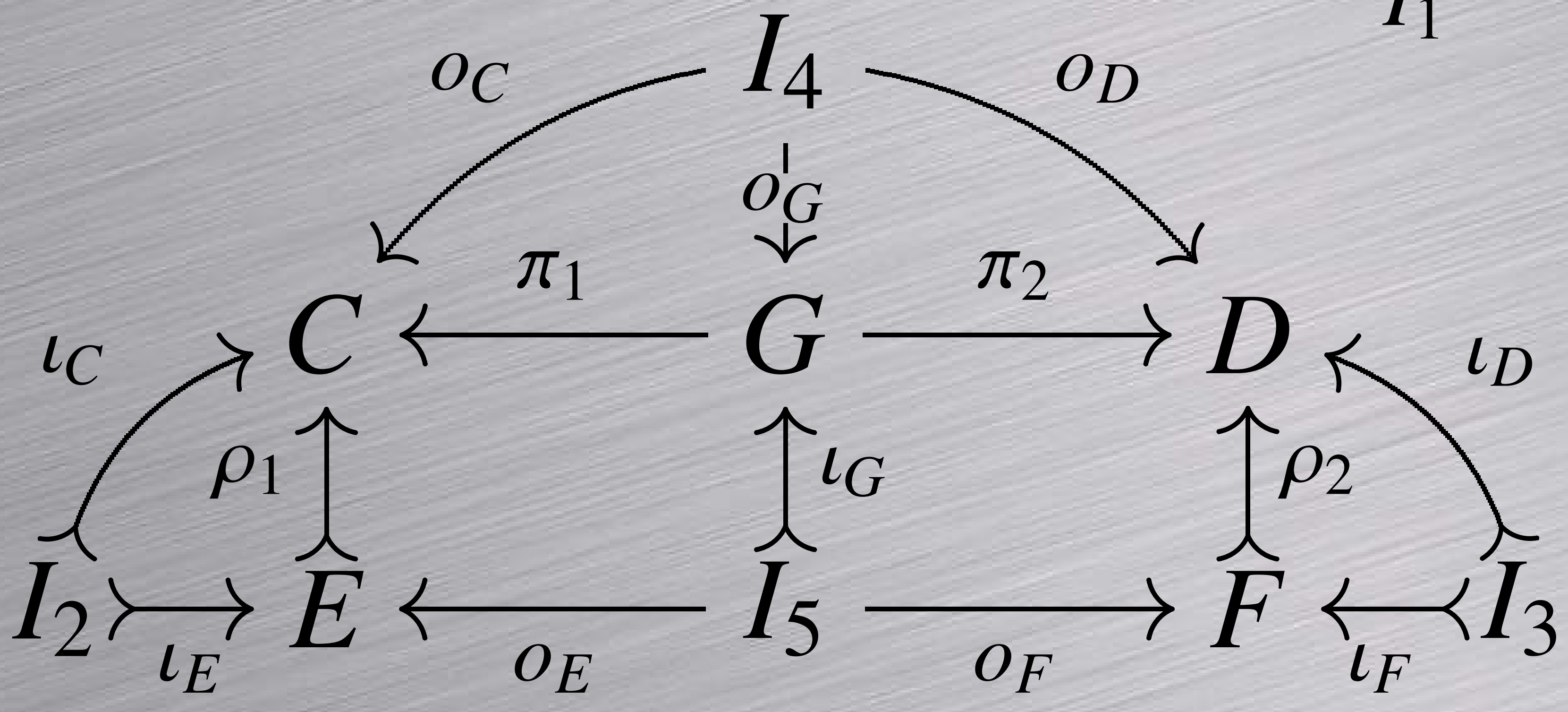




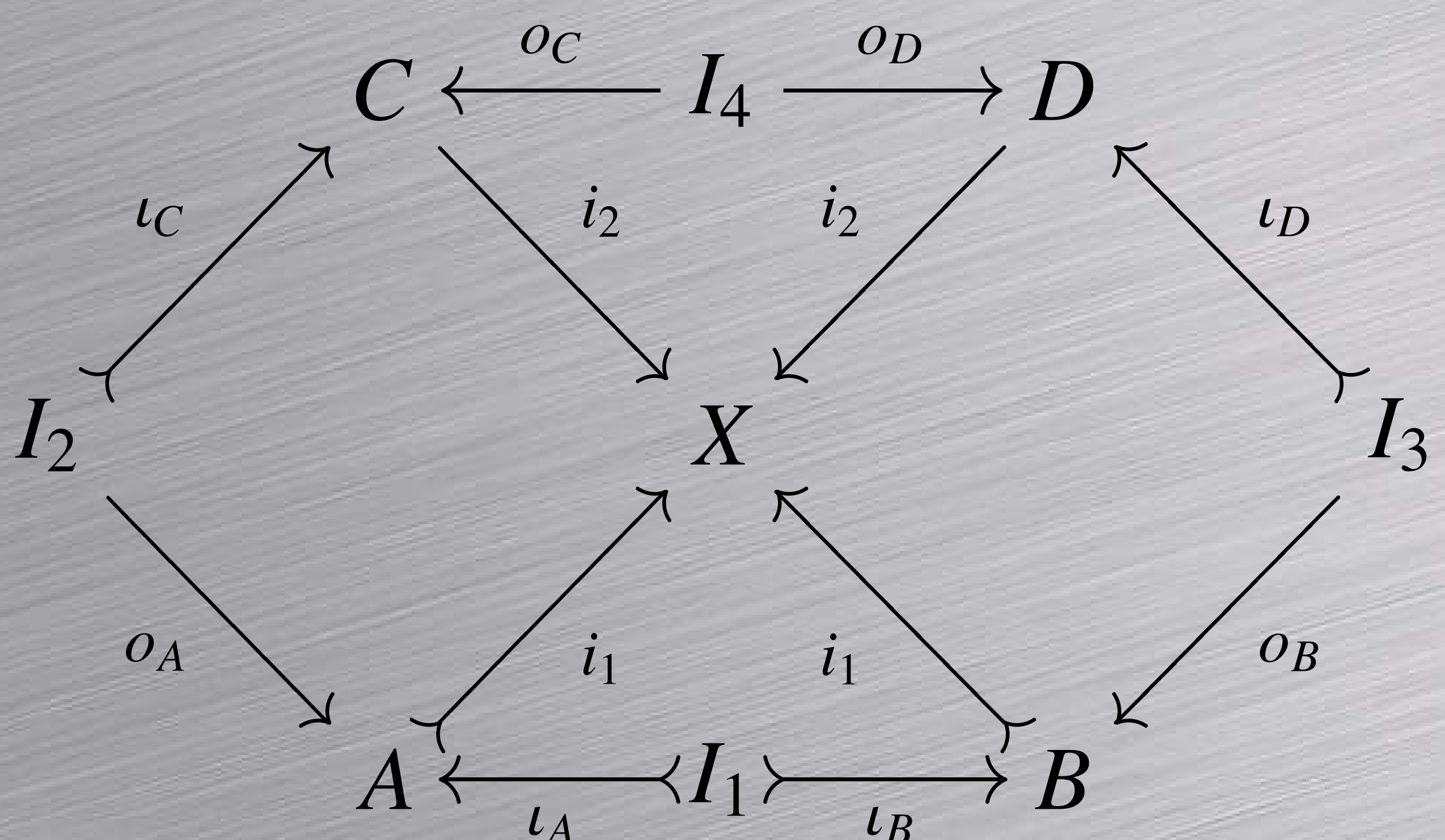
GRPOs of Cospans

- ... find minimal factorisation

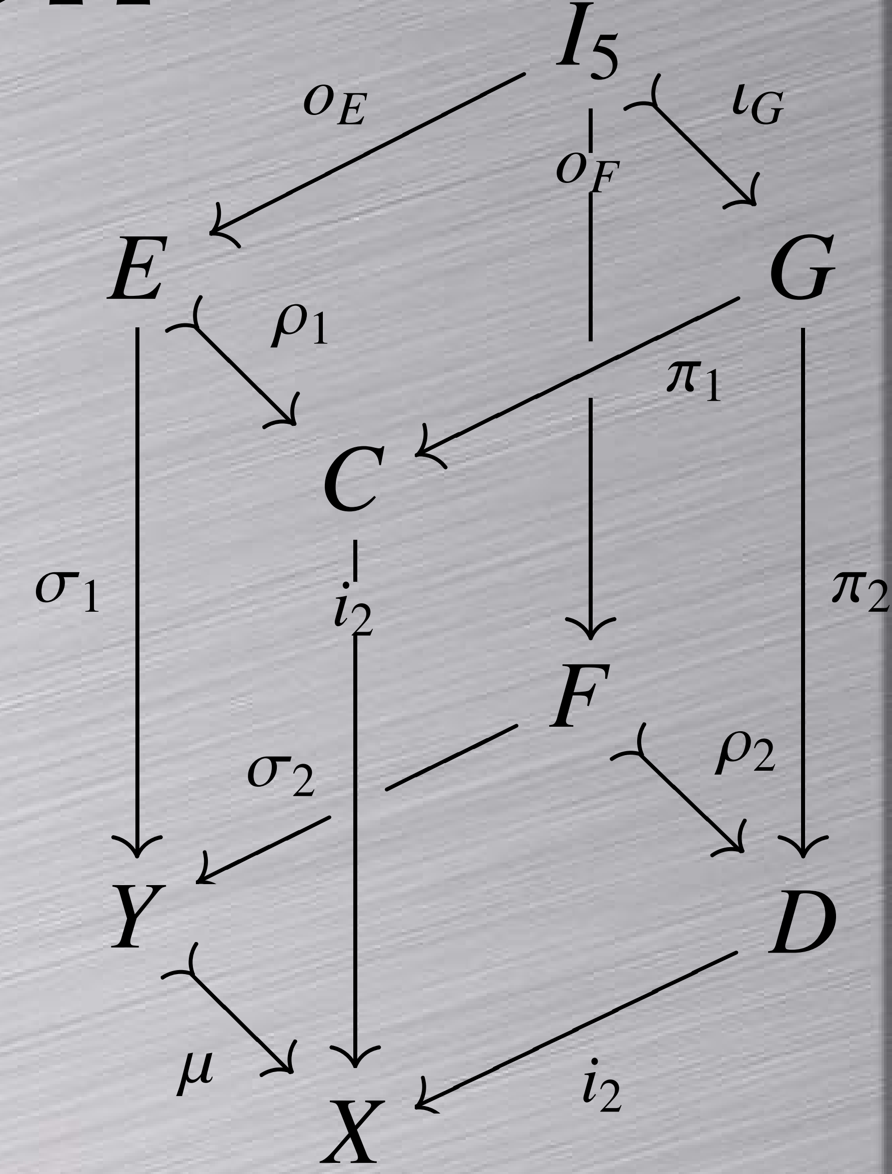




Construction



$$Y = A \cup_X B$$



Graph Rewriting as Reactive System

- For every span $L \xleftarrow{l} K \xrightarrow{r} R$
  let $\left\langle 0 \to L \xleftarrow{l} K, 0 \to R \xleftarrow{r} K \right\rangle \in \mathcal{R}$
- ***Lemma***:

  $\longrightarrow$ double-pushout rewrite

  $\longrightarrow\!\!\triangleright$ reaction relation in reactive system

$$C \longrightarrow D \qquad \text{iff} \qquad C_0^0 \longrightarrow\!\!\triangleright D_0^0$$

LTS for graph rewriting

- The resulting LTS has:
  - **Nodes**: graphs (up-to-iso) with output interface (possibly non-mono)
  - **Labels**: smallest graph contexts (up-to-iso) which allow reaction
- ***Theorem***: Bisimulation, trace equivalence, failures equivalence are congruences

Advantages of LTS

- Transfer of concepts from process algebra to graph rewriting
- Labelled, compositional semantics
  - the class of adhesive categories covers many categories with "*graph-like*" objects

And what's this for?

- What's missing here??

Special Cases

- Rewriting with borrowed contexts [*Ehrig and Koenig (2004)*]
    - LTS for graph rewriting, up-to-iso not taken into account, all interfaces mono
    - ***Theorem***: when restricting our approach to linear cospans we derive *the same* LTS
    - ***Corollary***: their congruence theorem
- Bigraphs...

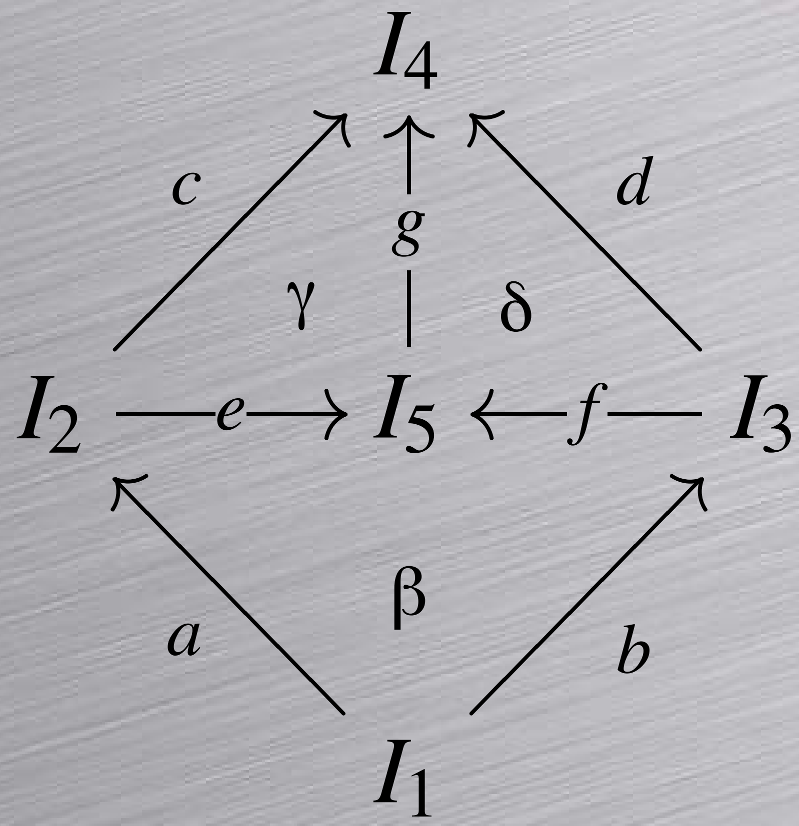
The case of Bigraphs

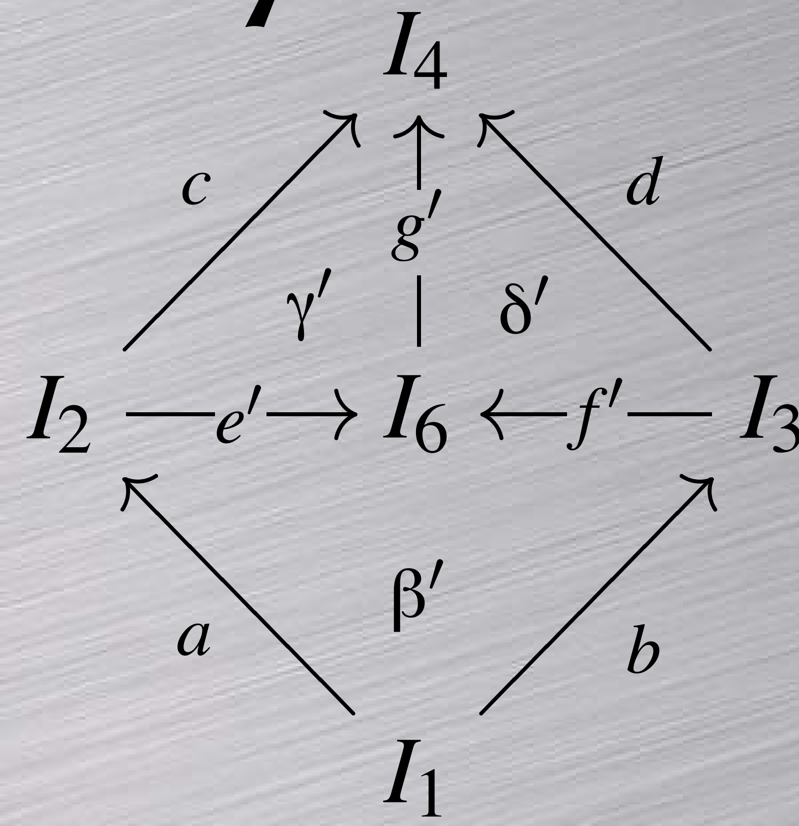
- **Bigraphs** can be seen as **LLC**(**dpl-grph**).
- It follows from the theorem that **Bigraphs** has **GRPOs**.
- Main difference with Milner's original bigraphs: input-lineary and name aliasing.
- The case of **Trigraphs** ... as above
- ...

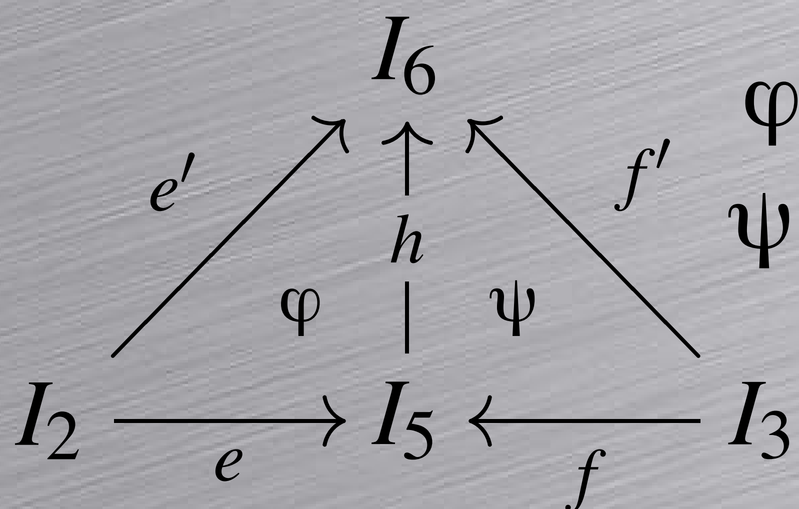
Conclusion

- Construction of labels for an interesting class of reactive systems
- Two applications so far, more in the future?

Minimality

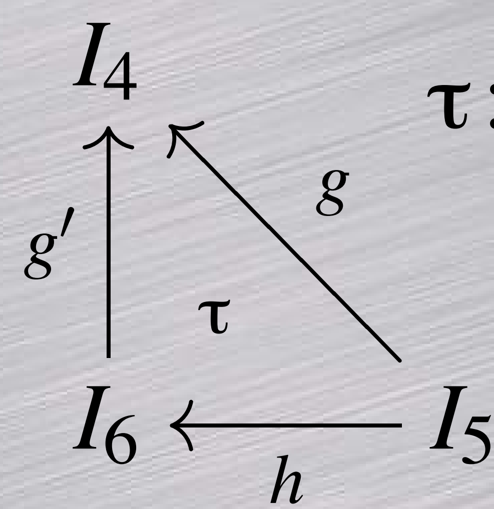






$$\varphi\colon e' \Rightarrow he$$

$$\psi\colon hf \Rightarrow f'$$



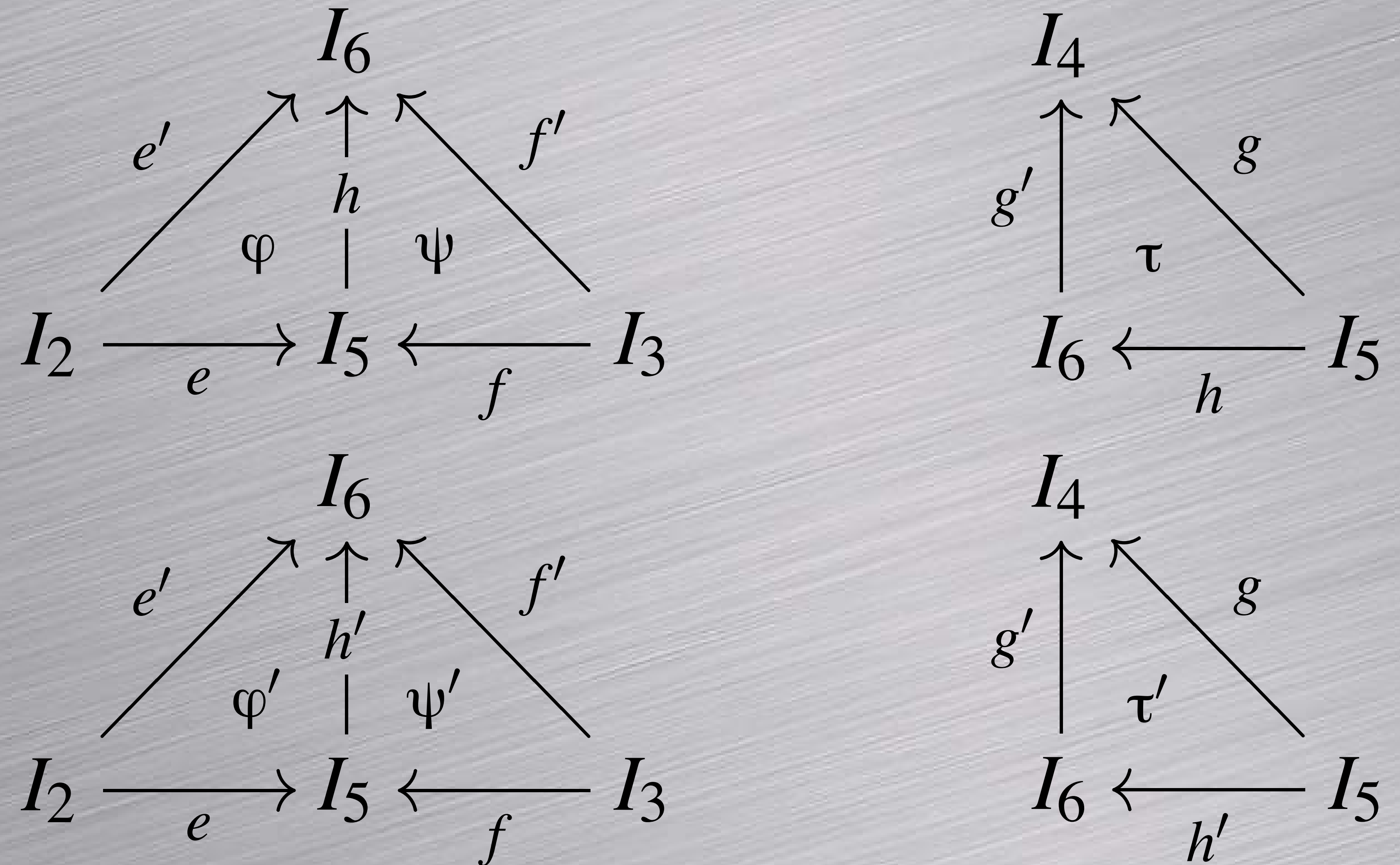
$$\tau\colon g'h \Rightarrow g$$

$$\tau e \bullet g'\varphi \bullet \gamma' = \gamma$$

$$\delta' \bullet g'\psi \bullet \tau^{-1} f = \delta$$

$$\psi b \bullet h\beta \bullet \varphi a = \beta'$$

Essential Uniqueness



$$\exists!\, \xi : h \to h'$$

$$\xi e \bullet \varphi = \varphi'$$
$$\psi \bullet \xi^{-1} f = \psi'$$
$$\tau' \bullet g' \xi = \tau$$