Functorial Data Integration

Categorical Informatics, Inc.

info@catinf.com

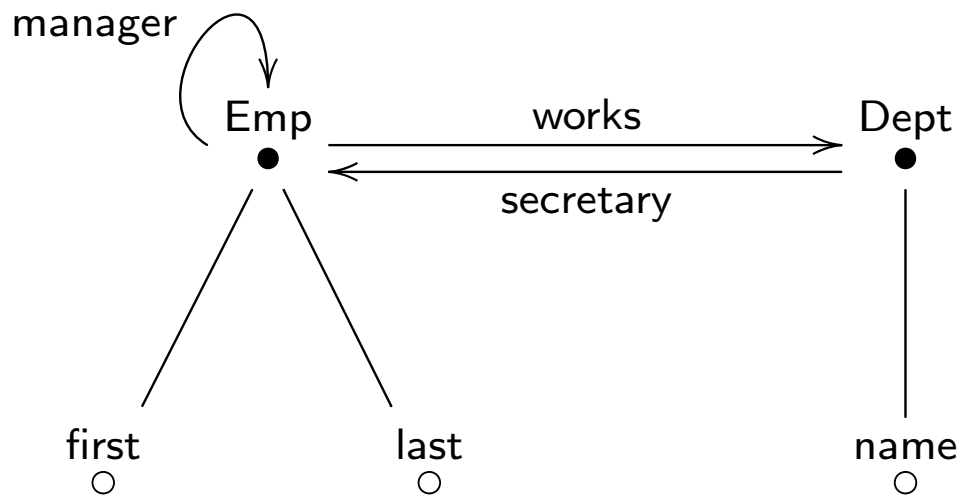
May 1, 2016

$$\mathcal{C}_i$$

Outline

- Review of Functorial Data Migration
  - Associated with a schema mapping $F : S \to T$ are three adjoint data migration operations Δ_F: T-inst $\to$ S-inst (project), Π_F: S-inst $\to$ T-inst (join), and Σ_F: S-inst $\to$ T-inst (union).
- Functorial Data Integration
  - The unary operations Δ, Σ, Π are sufficient to query data and to migrate data, but we require binary operations to *integrate* data from multiple sources.
  - Integration is expressed category-theoretically using a *pushout* construction.
- This talk:
  - Pushouts
  - How to use pushouts in data warehousing
  - Extended example
- The example is discussed in `http://arxiv.org/abs/1503.03571` and is included in FQL as "O Integration".

Review of the Functorial Data Model



Emp.manager.works = Emp.works Dept.secretary.works = Dept

Emp				
ID	**mgr**	**works**	**first**	**last**
101	103	q10	Al	Akin
102	102	x02	Bob	Bo
103	103	q10	Carl	Cork

Dept		
ID	**sec**	**name**
q10	102	CS
x02	101	Math

Review of Functorial Data Migration

- A functor $F\colon S \to T$ is a constraint-respecting mapping:

$$nodes(S) \to nodes(T) \qquad edges(S) \to paths(T)$$

  and it induces three adjoint data migration functors:

  - $\Delta_F\colon T$-inst $\to S$-inst (like project)

$$S \xrightarrow{F} T \xrightarrow{I} \mathbf{Set}$$
$$\Delta_F(I) := I \circ F$$

  - $\Pi_F\colon S$-inst $\to T$-inst (like join)

$$\Delta_F \dashv \Pi_F$$

  - $\Sigma_F\colon S$-inst $\to T$-inst (like outer disjoint union then quotient)

$$\Sigma_F \dashv \Delta_F$$

Data Integration

Data Integration

Data Integration

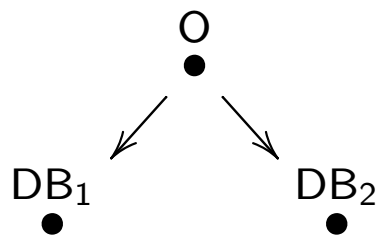
Data Integration

Pushouts

Given two databases:

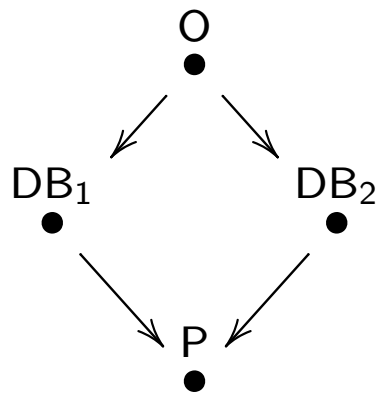
Pushouts

And an overlap between them:



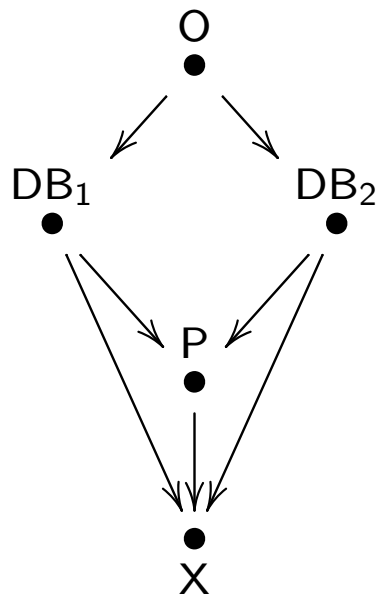
Pushouts

Their pushout is an integrated database:



Pushouts

That is universal among all possible integrated databases:



Pushouts in FQL

- Pushouts can be defined, but may not exist, in any category.
- FQL uses automated theorem proving techniques to compute pushouts.
- Pushouts play a role similar to "the chase" in relational database theory.

Using Pushouts for Data Integration

- Step 1: integrate schemas. Given input schemas S_1, S_2, an overlap schema S, and mappings F_1, F_2:

$$S_1 \overset{F_1}{\leftarrow} S \overset{F_2}{\rightarrow} S_2$$

  we propose to use their pushout T as the integrated schema:

$$S_1 \overset{G_1}{\rightarrow} T \overset{G_2}{\leftarrow} S_2$$

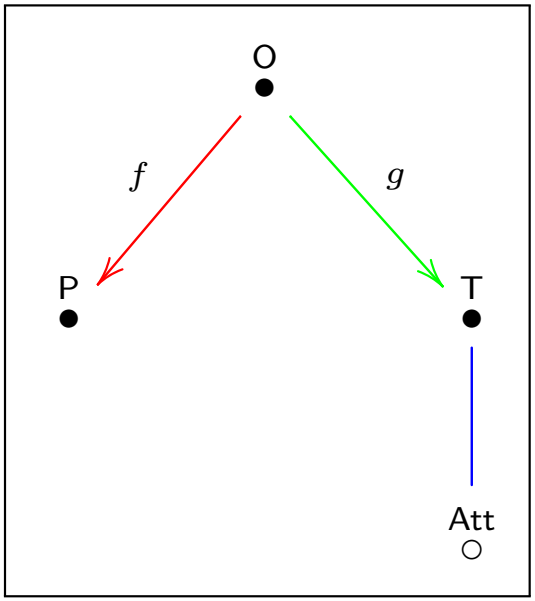
- Step 2: integrate data. Given input S_1-instance I_1, S_2-instance I_2, overlap S-instance I and transforms $h_1 : \Sigma_{F_1}(I) \rightarrow I_1$ and $h_2 : \Sigma_{F_2}(I) \rightarrow I_2$, we propose to use the pushout of:

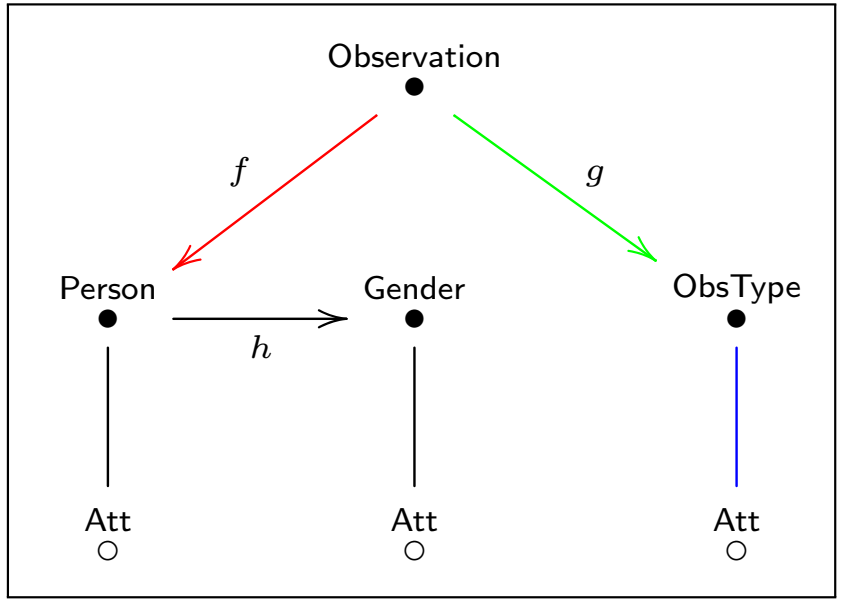
$$\Sigma_{G_1}(I_1) \overset{\Sigma_{G_1}(h_1)}{\leftarrow} \left(\Sigma_{G_1 \circ F_1}(I) = \Sigma_{G_2 \circ F_2}(I)\right) \overset{\Sigma_{G_2}(h_2)}{\rightarrow} \Sigma_{G_2}(I_2)$$

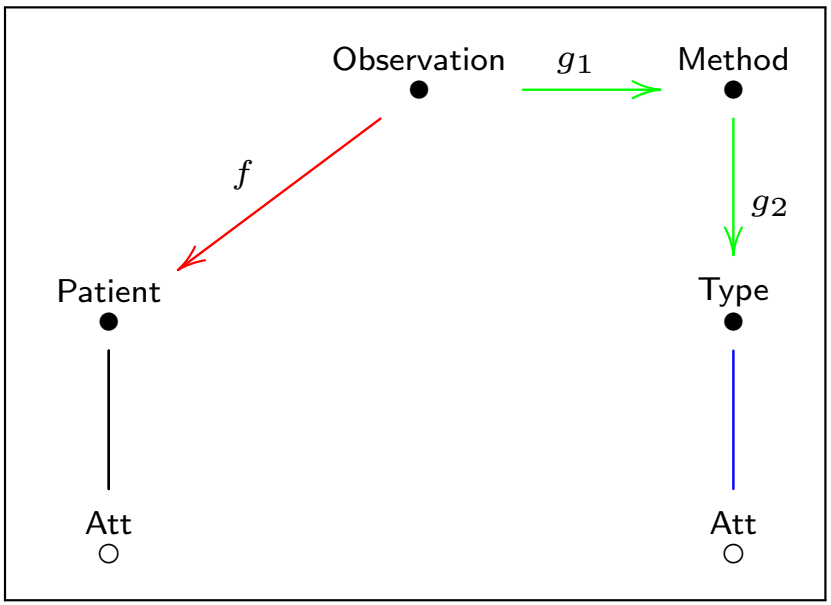
  as the integrated T-instance.

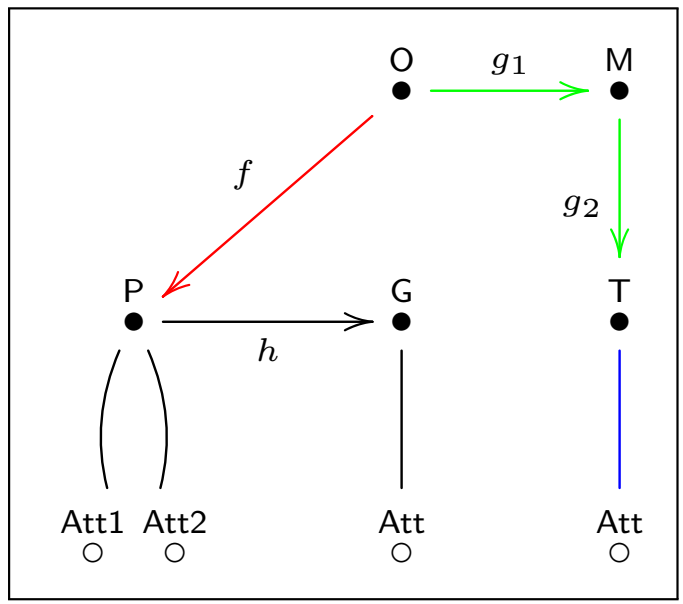
Using Pushouts for Data Integration

- For every two databases to integrate, the strategy just described is a design pattern:
- The input is one schema (S), two schema mappings (F_1, F_2), one database instance (I), and two database mappings (h_1, h_2).
- If desired, (S, F_1, F_2) can be constructed automatically using schema matching techniques.
- If desired, (I, h_1, h_2) can be constructed automatically using data matching techniques.
- The output is an integrated schema T and a "universal" (best possible) integrated T-instance.


O
f
g
P
T
Att


Observation
f
g
Person
h
Gender
ObsType
Att
Att
Att


Observation
g_1
Method
f
g_2
Patient
Type
Att
Att


O
g_1
M
f
g_2
P
h
G
T
Att1
Att2
Att
Att

O

ID	f	g

P

ID
p

T

ID	Att
bp	BP
wt	Weight

→

Gender

ID	Att
f	F
m	M

ObsType

ID	Att
bp	BP
wt	Weight
hr	HR

Observation

ID	f	g
o_5	pe	bp
o_6	pa	hr
o_7	pe	wt

Person

ID	Att	h
pa	Paul	m
p	Peter	m

↓

Method

ID	g2
m_1	bp
m_2	bp
m_3	wt
m_4	wt

Type

ID	Att
bp	BP
wt	Weight

Observation

ID	f	g
o_1	p	m_1
o_2	p	m_2
o_3	j	m_3
o_4	j	m_1

Patient

ID	Att
j	Jane
p	Pete

→

M

ID	g2
g1(o_5)	bp
g1(o_6)	wt
g1(o_7)	hr
m_1	bp
m_2	bp
m_3	wt
m_4	wt

O

ID	f	g1
o_1	p	m_1
o_2	p	m_2
o_3	j	m_3
o_4	j	m_1
o_5	p	g1(o_5)
o_6	pa	g1(o_6)
o_7	p	g1(o_7)

G

ID	Att
f	F
m	M
h(j)	Att(h(j))

T

ID	Att
bp	BP
wt	Weight
hr	HR

P

ID	Att1	Att2	h
j	Att1(j)	Jane	h(j)
pa	Paul	Att2(pa)	m
p	Peter	Pete	m

Conclusion

- Pushouts express “union with overlap, then quotient”.
- We propose a pattern for data integration:
  - Use pushouts to integrate source schemas.
  - Use pushouts to integrate source data.
- The example is discussed in `http://arxiv.org/abs/1503.03571` and is included in FQL as “O Integration”.