Presentation of Update Semantics of Relational Views

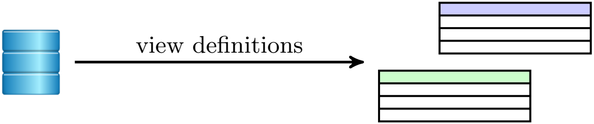
Nhung Ngo & Yu Liu

FCCOD 2014

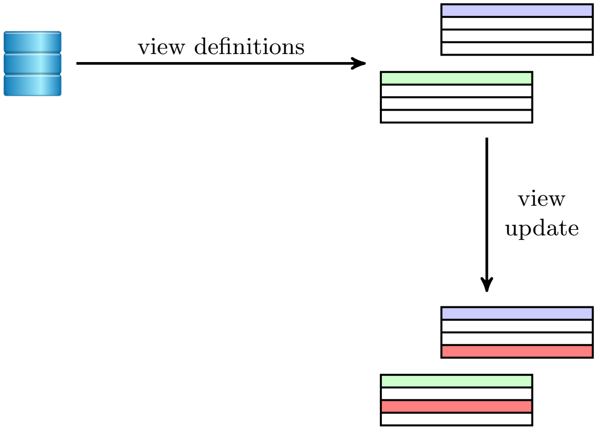
Outline

1. Problem
   - Overview of the problem being addresseds
   - Formal definition of the problem
2. Solution
   - Translation under constant complement
   - Update policy
   - Advantages and disadvantages of solutions

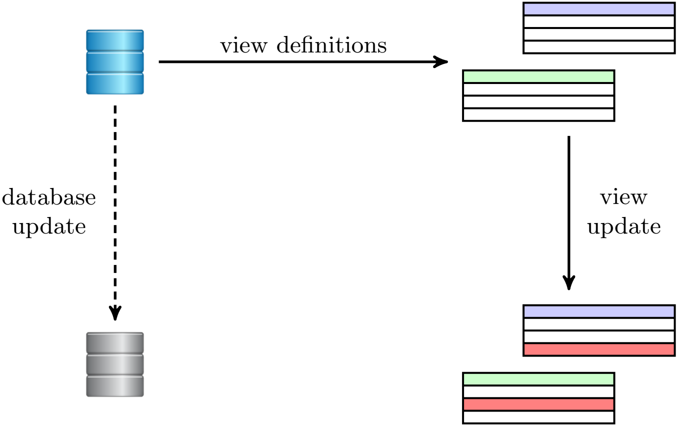
Overview



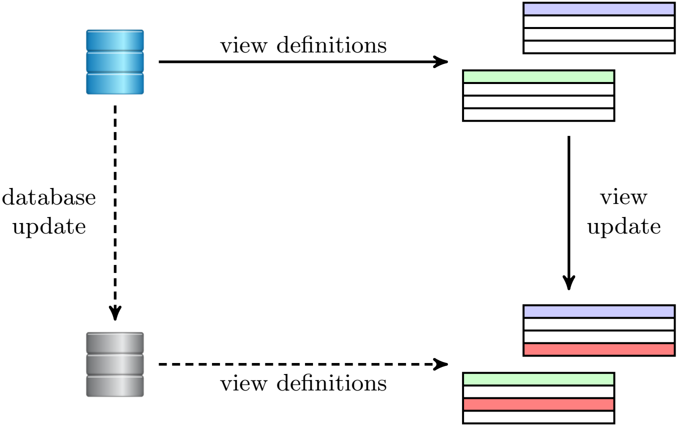
Overview



Overview



Overview


view definitions
database
update
view
update
view definitions

Example 1

Unexpected changes on the view

$$V = E \bowtie D$$

EMP	DEP	MGR
Mike	EEE	Susan

EMP	DEP
Mike	EEE
Mary	CS

DEP	MGR
EEE	Susan

Example 1

Unexpected changes on the view

$$V = E \bowtie D$$

EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex

EMP	DEP
Mike	EEE
Mary	CS

DEP	MGR
EEE	Susan

Example 1

Unexpected changes on the view

$$V = E \bowtie D$$

EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex

EMP	DEP
Mike	EEE
Mary	CS
Jane	CS

DEP	MGR
EEE	Susan
CS	Alex

Example 1

Unexpected changes on the view

$$V = E \bowtie D$$

EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex
Mary	CS	Alex

EMP	DEP
Mike	EEE
Mary	CS
Jane	CS

DEP	MGR
EEE	Susan
CS	Alex

Example 1

Unexpected changes on the view

$$V = E \bowtie D$$

EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex
Mary	CS	Alex

EMP	DEP
Mike	EEE
Mary	CS
Jane	CS

DEP	MGR
EEE	Susan
CS	Alex

The changes on the db must reflect *exactly* the changes on the view

Example 2

Unjustified changes on the database

$V = \pi_{EMP,\ DEP}(R)$

EMP	DEP
Mike	EEE
Jane	CS

R

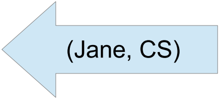
EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex
Jane	CS	Max

Example 2

Unjustified changes on the database

$V = \pi_{EMP,\ DEP}(R)$

EMP	DEP
Mike	EEE
Jane	CS



R

EMP	DEP	MGR
Mike	EEE	Susan
Jane	CS	Alex
Jane	CS	Max

Example 2

Unjustified changes on the database

$$V = \pi_{\text{EMP, DEP}}(R)$$

EMP	DEP
Mike	EEE
Jane	CS

(Jane, CS)

R

EMP	DEP	MGR
Mike	EEE	Susan
~~Jane~~	~~CS~~	~~Alex~~
Jane	CS	Max

Example 2

Unjustified changes on the database

$V = \pi_{EMP,\ DEP}(R)$

EMP	DEP
Mike	EEE
Jane	CS

← (Jane, CS)

R

EMP	DEP	MGR
Mike	EEE	Susan
~~Jane~~	~~CS~~	~~Alex~~
Jane	CS	Max

Modify the database *only if required* to reflect the changes on the view

Basic Notation

S database schema - set of all database instances (*database states*)

T set of all view instances (*view states*)

- view $f\colon S \to T$
- view update $u\colon T \to T$
- database update $d\colon S \to S$

U_1 set of all database updates

U_f set of all view updates

Concepts to be formalized

- Q1: Given a view update u, what are the constraints on the database update that translates u ?

Concepts to be formalized

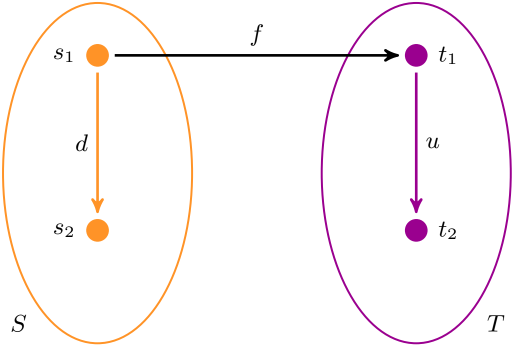
- Q1: Given a view update u, what are the constraints on the database update that translates u ?
- Q2: What sets of view updates do we want to translate, that is, what sets of updates users are to be allowed on the view ?

Concepts to be formalized

- Q1: Given a view update u, what are the constraints on the database update that translates u ?
- Q2: What sets of view updates do we want to translate, that is, what sets of updates users are to be allowed on the view ?
- Q3: How do we associate with each view update a database update that translates it ?

Definitions

Q1: A translation of a view update



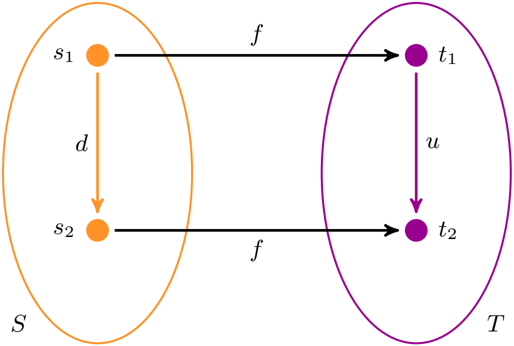
A database update d is a *translation* of a view update u iff
for each database state $s \in S$

(1) $uf(s) = fd(s)$ (*consistent*)

(2) $uf(s) = f(s) \rightarrow d(s) = s$ (*acceptable*)

Definitions

Q1: A translation of a view update



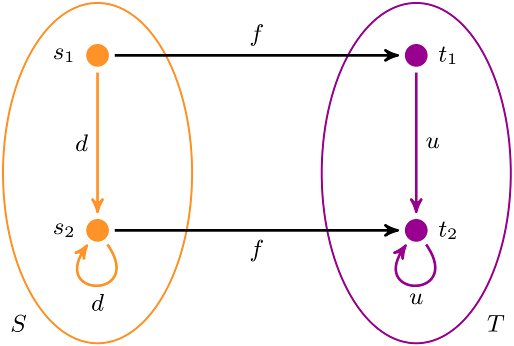
A database update d is a *translation* of a view update u iff for each database state $s \in S$

(1) $uf(s) = fd(s)$ (*consistent*)

(2) $uf(s) = f(s) \rightarrow d(s) = s$ (*acceptable*)

Definitions

Q1: A translation of a view update



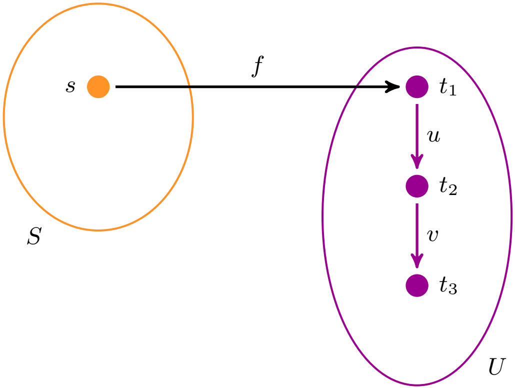
A database update d is a *translation* of a view update u iff for each database state $s \in S$

(1) $uf(s) = fd(s)$ (*consistent*)

(2) $uf(s) = f(s) \rightarrow d(s) = s$ (*acceptable*)

Definitions

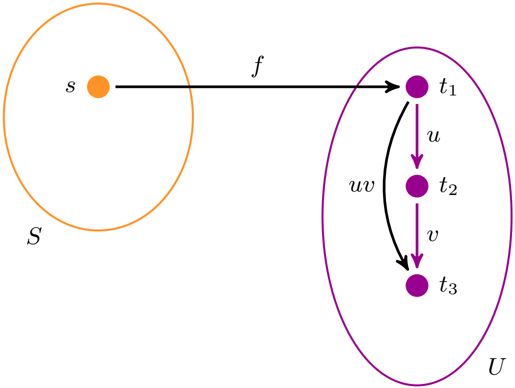
Q2: A complete set of view updates



A set U of view updates is called *complete* iff

Definitions

Q2: A complete set of view updates

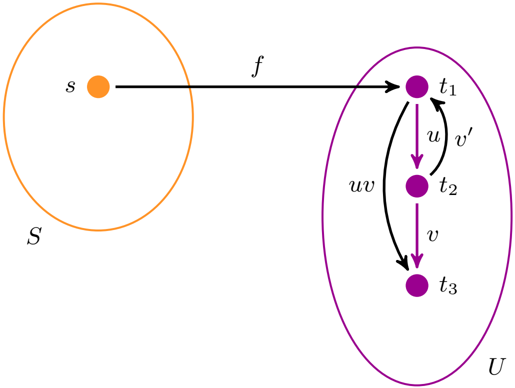


A set U of view updates is called *complete* iff

(1) $\forall u, v \in U,\ uv \in U$

Definitions

Q2: A complete set of view updates



A set U of view updates is called *complete* iff

(1) $\forall u, v \in U,\ uv \in U$

(2) $\forall s \in S,\ \forall u \in U,\ \exists u' \in U\ \ u'uf(s) = f(s)$

Definitions

Q3: A translator

A mapping $T : U \to U_1$ is called a *translator* iff

(1) $\forall u \in U$, T_u is a translation of u

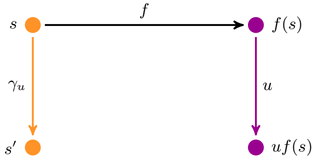
(2) $\forall u, v \in U$, $T_{uv} = T_u T_v$

The view update problem

Given a complete set U of view updates, find a translator of U

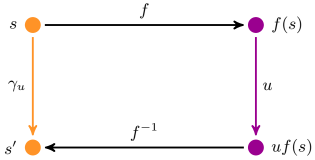
Intuitive idea

- If the view f is injective



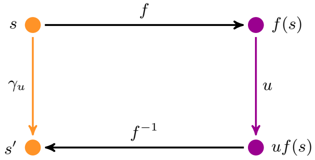
Intuitive idea

- If the view f is injective



Intuitive idea

- If the view f is injective



- If the view f is not injective
  Need a view complement g of f so that $f \times g$ is injective

The view complement

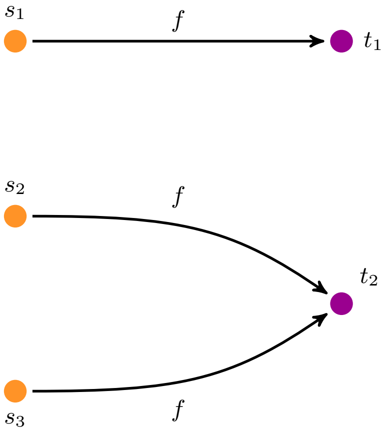
g is a *complement* of f iff $f \times g = 1$

g is a *complement* of f iff $\forall s, s' \in S_\Sigma,\ s \neq s' \wedge f(s) = f(s') \rightarrow g(s) \neq g(s')$

- A complement of f contains "the information not visible within f "
- A complement of f is able to distinguish database states that f maps to the same view state
- A view complement always exists (a renamed copy of the whole db schema in the worst case)
- In general, there is no unique minimal complement

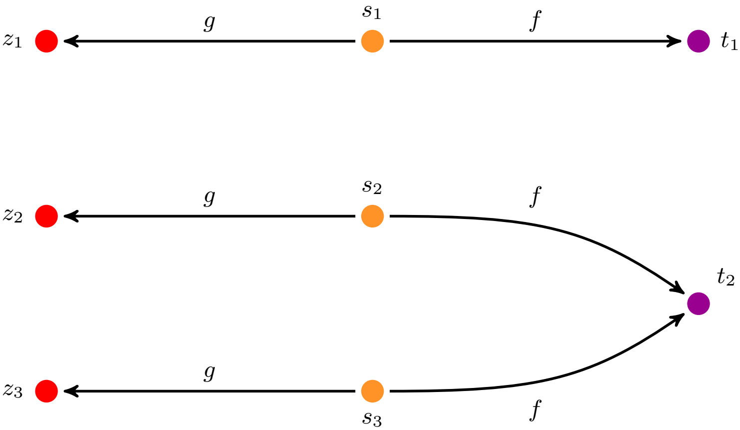
The view complement

An example



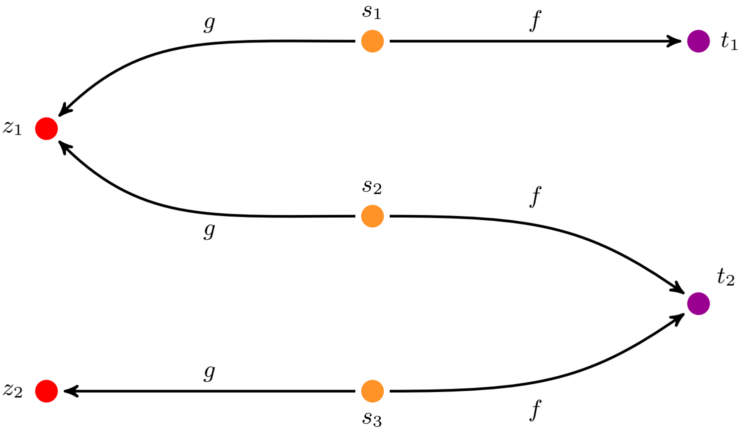
The view complement

An example



The view complement

An example



Constant Complement & Translation under constant complement

Rectangle Rule from Chamberlin et al (1975): *"An insertion, deletion, or update via a view must affect only information visible within the rectangle of the view."*

- A complement g of a view update u should not be changed (i.e. invariant) by a database update.
- A translation γ_u of u should be verified that it makes g invariant.

A given database, a view and a complement view

E(EMP, DEP)
M(DEP, MGR)
C1: EMP $\rightarrow$ DEP
C2: DEP $\leftrightarrow$ MGR
C3: E[DEP] = M[DEP]

$s =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$f(s) = \pi_{DEP,MGR} E \bowtie M$

$f =$

EM	
DEP	MGR
Mary	Alex
Jane	Alex
Mike	Susan

$g(s) = M$

$g =$

M	
DEP	MGR
CS	Alex
EEE	Susan

Example of a translation that leaves a complement invariant

u: Replace employee Mary by employee John.

g: Table M.

γ_u : E $=$ (M*u(EM))[EMP,DEP];M $=$ M.

$s =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\xrightarrow{\gamma_u}$ $s' =$

E	
EMP	DEP
John	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\downarrow$

$f =$

EM	
DEP	MGR
Mary	Alex
Jane	Alex
Mike	Susan

$\xrightarrow{u}$ $uf =$

EM	
DEP	MGR
John	Alex
Jane	Alex
Mike	Susan

u is $g - translatable$

u is $g - translatable$ iff for all s in S, there exists a s' that:

1. $f(s') = uf(s)$
2. $g(s') = g(s)$

The composition of $g - translatable$ updates is also $g - translatable$

- u, v are $g - translatable \Rightarrow uv$ is $g - translatable$
  1. $f(s'') = u(f(s')) = uvf(s)$
  2. $g(s'') = g(s') = g(s)$

Example of a composition of $g - translatable$ updates

u: Replace employee Mary by employee John.
v: Replace employee Jane by employee Lewis.
g: Table M.
γ_{uv} : E = (M*uv(EM))[EMP,DEP];M = M.

$s =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\overset{\gamma_{uv}}{\longmapsto}$ $s'' =$

E	
EMP	DEP
John	CS
Lewis	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\downarrow$

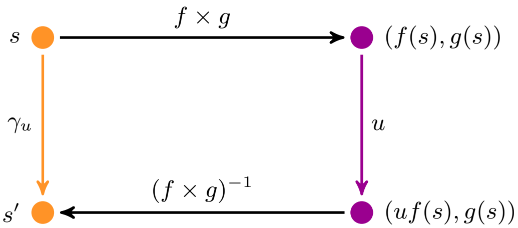
$f =$

EM	
DEP	MGR
Mary	Alex
Jane	Alex
Mike	Susan

$\overset{uv}{\longmapsto}$ $uvf =$

EM	
DEP	MGR
John	Alex
Lewis	Alex
Mike	Susan

$g - translation : \gamma_u$



For a given f, g, u if u is $g - translatable$ then $\gamma_u = (f \times g)^{-1}(uf \times g)$.

γ_u is a translation of u (γ_u is called a $g - translation$ of u):

- $uf = f\gamma_u \rightsquigarrow Consistent$
- $\gamma_u(s) = s \rightsquigarrow Acceptable$

γ_u leaves g invariant

- $g\gamma_u = g$

If u is $g - translatable$, γ_u always exists and is unique.

How to choose a complement of a view?(1)

The choice of g impacts that whether u is $g - translatable$ or not.

u: Replace employee Mary by employee John.
g': Table E.
γ_u : E = (M*u(EM))[EMP,DEP];M = M.

$s =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\overset{\gamma_u}{\longmapsto}$ $s' =$

E	
EMP	DEP
John	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\downarrow$

$f =$

EM	
DEP	MGR
Mary	Alex
Jane	Alex
Mike	Susan

$\overset{u}{\longmapsto}$ $uf =$

EM	
DEP	MGR
John	Alex
Jane	Alex
Mike	Susan

How to choose a complement of a view?(2)

The choice of g impacts that whether u is $g - translatable$ or not.

w: Permute the managers.
g': Table E.
γ_w : E = E;M = (E*w(EM))[EMP,DEP].

$s =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Alex
EEE	Susan

$\xrightarrow{\gamma_w}$ $s' =$

E	
EMP	DEP
Mary	CS
Jane	CS
Mike	EEE

M	
DEP	MGR
CS	Susan
EEE	Alex

$\downarrow$

$f =$

EM	
DEP	MGR
Mary	Alex
Jane	Alex
Mike	Susan

$\xrightarrow{w}$ $wf =$

EM	
DEP	MGR
Mary	Susan
Jane	Susan
Mike	Alex

How to choose a complement of a view?(3)

For a given view u, g, h are both complements of f and h contains less information than g. If u is $g - translatable$ then:

1. u is also $h - translatable$
2. $h - translation = g - translation$

The set of $g - translatable$ updates is maximal when the complement g is minimal.

- We could like to find the minimal complements, so that get maximal update sets (a minimal complement is not unique).

A complement view: an update policy

Universal property of translation under constant complement

Given a complete set $U \subset U_f$, a view f and a complement view g of f:

This paper provided translators T for U if $\forall u \in U$: u is $g - translatable$:

1. Select a complement g of the given view f.
2. Verify that view updates of the given complete set U make g invariant.
3. For each view update $u \in U$, the translation $T_u = (f \times g)^{-1}(uf \times g)$.

For every T of U, there exists a complement g that:

1. $\forall u \in U$, u is $g - translatable$.
2. $\forall u \in U$, $g - translation\ \gamma_u = (f \times g)^{-1}(uf \times g)$

Advantages & Disadvantages

Advantages:

1. This paper provides a formal framework for solving the view update problem.
2. The method is beneficial for solving view update issues in Data Integration.

Disadvantages:

1. Too theoretical, no algorithms for implementation from practical point of view.
2. This paper does not show how to find a minimal complement.

Related works

- Lechtenboerger (2003) gives a characterisation of the constant complement principle in terms of "undo" operations in SQL server.
- Cosmadakis and Papadimitriou (1984) consider a restricted setting that consists of a single database relation and two views defined by projections.
- Gottlob et al (1988) extend to the class of so-called consistent views, which properly contains the views translating under constant complement. The complement is not required to remain invariant in their framework.
- Enrico Franconi and Paolo Guagliardo [2011] provide a general framework for view updating (under constraints) based on the notion of determinacy .