

Views and View Management

Grant Weddell

Cheriton School of Computer Science
University of Waterloo

CS 348
Introduction to Database Management
Winter 2017

Definition (View)

A *view* is a relation whose instance is determined by the instances of other relations.

A view has many of the same properties as a table.

- its schema information appears in the database schema
- access controls can be applied to it
- other views can be defined in terms of it

Types of Views

- **Virtual:** Views are used only for querying; they are not stored in the database
- **Materialized:** The query that makes up the view is executed, the view constructed and stored in the database.

- General form:

```
create [materialized] view <name>  
  as <query>
```

- Example

```
create view ManufacturingProjects as  
  ( select projno, projname, firstnme, lastname  
    from project, employee  
    where respemp = empno and deptno = 'D21' )
```

Accessing a View

Query a view as if it were a base relation.

```
select projname  
from ManufacturingProjects
```

Query a view as if it were a base relation.

```
select projname  
from ManufacturingProjects
```

What happens when you query a **virtual** view?

- At compile time, the view definition is found
- The query over the view is modified with the query definition
- The resulting query is optimized and executed

Updating Views

- Modifications to a view's instance must be propagated back to instances of relations in conceptual schema.

Updating Views

- Modifications to a view's instance must be propagated back to instances of relations in conceptual schema.
- Some views cannot be updated unambiguously.

Updating Views

- Modifications to a view's instance must be propagated back to instances of relations in conceptual schema.
- Some views cannot be updated unambiguously.

Conceptual Schema

Persons

NAME	CITIZENSHIP
Ed	Canadian
Dave	Canadian
Wes	American

NationalPastimes

CITIZENSHIP	PASTIME
Canadian	Hockey
Canadian	Curling
American	Hockey
American	Baseball



External Schema

PersonalPastimes

NAME	PASTIME
Ed	Hockey
Ed	Curling
Dave	Hockey
Dave	Curling
Wes	Hockey
Wes	Baseball

Updating Views

- Modifications to a view's instance must be propagated back to instances of relations in conceptual schema.
- Some views cannot be updated unambiguously.

Conceptual Schema

Persons

NAME	CITIZENSHIP
Ed	Canadian
Dave	Canadian
Wes	American

NationalPastimes

CITIZENSHIP	PASTIME
Canadian	Hockey
Canadian	Curling
American	Hockey
American	Baseball



External Schema

PersonalPastimes

NAME	PASTIME
Ed	Hockey
Ed	Curling
Dave	Hockey
Dave	Curling
Wes	Hockey
Wes	Baseball

- 1 What does it mean to insert (Darryl, Hockey)?

Updating Views

- Modifications to a view's instance must be propagated back to instances of relations in conceptual schema.
- Some views cannot be updated unambiguously.

Conceptual Schema

Persons

NAME	CITIZENSHIP
Ed	Canadian
Dave	Canadian
Wes	American

NationalPastimes

CITIZENSHIP	PASTIME
Canadian	Hockey
Canadian	Curling
American	Hockey
American	Baseball



External Schema

PersonalPastimes

NAME	PASTIME
Ed	Hockey
Ed	Curling
Dave	Hockey
Dave	Curling
Wes	Hockey
Wes	Baseball

- 1 What does it mean to insert (Darryl, Hockey)?
- 2 What does it mean to delete (Dave, Curling)?

View Updates in SQL

According to SQL-92, a view is updatable only if its definition satisfies a variety of conditions:

- The query references exactly one table
- The query only outputs simple attributes (no expressions)
- There is no grouping/aggregation/**distinct**
- There are no nested queries
- There are no set operations

These rules are more restrictive than necessary.

Problem

When a base table changes, the materialized view may also change.

Problem

When a base table changes, the materialized view may also change.

- Solution?
 - Periodically reconstruct the materialized view.
 - Incrementally update the materialized view.

Problem

When a base table changes, the materialized view may also change.

- Solution?
 - Periodically reconstruct the materialized view.
 - Incrementally update the materialized view.
- Example: Data warehouses