

Assertions and Triggers

29 December 2009
Lecture 11

Topics for Today

- Complex integrity constraints
- Triggers and Active Databases

Source: RG 5.7-5.9

December 29, 2009

ISE 322: Database Systems

2

Complex Integrity Constraints

- More powerful than the constraints we have seen until now
- Use the CHECK statement in the schema
 - Anything we can write in SQL can go in

December 29, 2009

ISE 322: Database Systems

3

Example: Range Constraint

- Ensure the rating of a sailor is between 1 and 10

```
CREATE TABLE Sailors (  
  sid INTEGER,  
  sname CHAR(10),  
  rating INTEGER,  
  age REAL,  
  PRIMARY KEY (sid),  
  CHECK (rating >= 1 AND rating <= 10))
```

The same as checking:
0 = (SELECT COUNT(*)
FROM Sailors
WHERE NOT (rating >= 1
AND rating <= 10))

Works in MS SQL.

December 29, 2009

ISE 322: Database Systems

4

Complex Integrity Constraints

- Ensure no boat named Interlake is reserved
- Not supported in MS SQL.

```
CREATE TABLE Reserves (  
  sid INTEGER,  
  bid INTEGER,  
  day DATETIME,  
  FOREIGN KEY (sid) REFERENCES Sailors,  
  FOREIGN KEY (bid) REFERENCES Boats,  
  CONSTRAINT noInterlakeRes CHECK ('Interlake' <>  
  (SELECT B.bname  
  FROM Boats B  
  WHERE B.bid = Reserves.bid)))
```

December 29, 2009

ISE 322: Database Systems

5

Complex Integrity Constraints

- Whenever rows are added or modified in Reserves, this is checked
 - If we change the name of a boat in the Boats table, this won't raise an error
- If it's **false**, the update is **rejected**
- MS SQL Server only supports scalar checks in the CHECK conditions, no subqueries
 - Vendor specific implementations
 - MySQL doesn't support CHECKs at all to my knowledge

December 29, 2009

ISE 322: Database Systems

6

Domain Constraints

- Create a custom type with specific integrity or range constraints
 - CREATE DOMAIN
- Creating a special type for rating value of a sailor
 - rating ratingval
- You can then define fields
 - rating ratingval

```
CREATE DOMAIN ratingval INTEGER DEFAULT 1 CHECK
(VALUE >= 1 AND VALUE <= 10)
```

December 29, 2009

ISE 322: Database Systems

7

Domain Constraints: MS SQL

```
CREATE TYPE type_name FROM base_type [( precision [, scale ] )][
NULL | NOT NULL ]
```

- Type_name: the new alias
- Base_type: the system base type
- Precision: (numeric) the number of digits it holds (total)
- Scale: (numeric) the number of digits past the decimal (<= precision)
- NULL/NOT NULL: whether the type can hold NULL (default NULL)

- Examples:
 - CREATE TYPE ratingval FROM INTEGER
 - CREATE TYPE TZ FROM varchar(11) NOT NULL
- You may compare the new types against the original base type

December 29, 2009

ISE 322: Database Systems

8

Distinct Types

- With Domain Constraints, you can compare any two fields with the same base type
- To divide them: CREATE TYPE
 - CREATE TYPE ratingtype AS INTEGER
 - Only in SQL:1999
 - You then must define all operators on the new type (23.4.1 if you are interested)

Not supported in MS SQL. However, you can achieve the same thing using User-Defined Types (next semester).

December 29, 2009

ISE 322: Database Systems

9

Assertions

- Integrity Constraints over several tables
- Constraints are useful, but
 - Only defined on one table
 - Can reference other ones
 - If the table is empty, it's defined true
- Leads us to write ASSERTIONS
 - DBMS always checks them
- The assertion is just a CHECK query:
 - If it's a boolean result – TRUE is Ok and FALSE raises an error
 - If it's a query – if the result has 0 rows it's Ok if it has more than 0 rows, it raises an error

Not supported in MS SQL.

December 29, 2009

ISE 322: Database Systems

10

Example

- The number of boats plus the number of sailors is less than 100

```
CREATE TABLE Sailors (
  sid INTEGER PRIMARY KEY,
  sname CHAR(10),
  rating INTEGER,
  age REAL,
  CHECK (rating >= 1 AND rating <= 10),
  CHECK (
    (SELECT COUNT(S.sid) FROM Sailors S
     +
     SELECT (COUNT(B.bid) FROM Boats b)
    < 100)
  )
```

1. If Sailors is empty, it's always true.

2. Updating Boats won't check the constraint

December 29, 2009

ISE 322: Database Systems

11

Example

- The number of boats plus the number of sailors is less than 100

```
CREATE ASSERTION smallClub CHECK (
  (SELECT COUNT(S.sid) FROM Sailors S)
  + (SELECT COUNT(B.bid) FROM Boats B)
  < 100)
```

December 29, 2009

ISE 322: Database Systems

12

So Far

- Complex integrity constraints
- Triggers and Active Databases

December 29, 2009

ISE 322: Database Systems

13

Triggers

- Trigger: a procedure that is invoked automatically in the data base based on some predefined conditions
- Three parts:
 - Event – The change which activates
 - Condition – Test to perform when the event
 - Action – What to do if the condition is true
- A database with triggers is an **active database**

December 29, 2009

ISE 322: Database Systems

14

Triggers: What they do

- The condition is evaluated
 - If it's true or a non-empty query, it's true
 - If it's false or an empty query, it's false
- The action can
 - Examine the updates sent
 - Examine the old/new data rows
 - Execute new queries
 - Define new tables, schemas, columns
 - Change the statement which activated the trigger
 - Do commits/aborts
 - Perform host language operations

December 29, 2009

ISE 322: Database Systems

15

Triggers: Running

- Triggers can be run before or after the event occurs.
- They can even run *instead of* the event.
- What to do then depends on the goals
- Can be triggered without the user knowing
 - Logging
 - Auditing
 - Integrity
- A trigger does things behind your back
 - Can be bad for usability
 - Can be good for automation or mandatory enforcement of rules

December 29, 2009

ISE 322: Database Systems

16

Trigger Types

- **Statement level triggers** which run for each statement (command) which matches the condition for the event
- **Row-level triggers** that run once for each row which is new or modified

Example:

```
UPDATE Sailors SET rating = rating + 1 WHERE age = 20
```

If five (5) sailors match the condition:

- A statement level trigger will run once
- A row level trigger will run five times

December 29, 2009

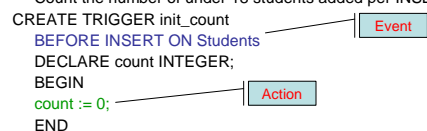
ISE 322: Database Systems

17

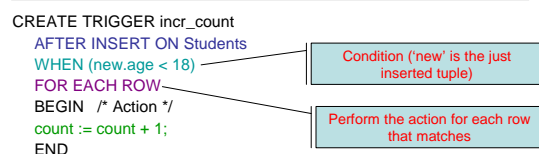
Trigger: PL/SQL Examples

- Count the number of under 18 students added per INSERT statement

```
CREATE TRIGGER init_count
BEFORE INSERT ON Students
DECLARE count INTEGER;
BEGIN
count := 0;
END
```



```
CREATE TRIGGER incr_count
AFTER INSERT ON Students
WHEN (new.age < 18)
FOR EACH ROW
BEGIN /* Action */
count := count + 1;
END
```



December 29, 2009

ISE 322: Database Systems

18

Trigger: SQL Example

- For each INSERT statement, record the number of under 18 students entered in the StatisticsTable

```
CREATE TRIGGER set_count
AFTER INSERT ON Students
REFERENCING NEW TABLE AS InsertedTuples
FOR EACH STATEMENT
INSERT INTO StatisticsTable (ModifiedTable,
ModificationType, Count)
SELECT 'Students', 'Insert', COUNT(*)
FROM InsertedTuples I
WHERE I.age < 18
```

Creates a virtual table with just the newly entered rows

Default

WHEN would go here

December 29, 2009

ISE 322: Database Systems

19

Trigger: SQL Example

- When a bank account goes below zero, we allow "minus" by making an overdraft into a loan

```
CREATE TRIGGER overdraft_trigger
AFTER UPDATE ON Account
REFERENCING NEW ROW AS nrow
FOR EACH ROW
WHEN nrow.balance < 0
BEGIN ATOMIC
INSERT INTO Borrower VALUES
(SELECT customer_name, account_number
FROM Depositor D
WHERE nrow.account_number = D.account_number);
INSERT INTO Loan VALUES (nrow.account_number,
nrow.branch_name, 0 - nrow.balance);
UPDATE Account A SET A.balance = 0
WHERE A.account_number = nrow.account_number;
END
```

December 29, 2009

ISE 322: Database Systems

20

Trigger: SQL Server Example

- For each INSERT statement, record the number of under 18 sailors entered in the StatisticsTable

```
CREATE TRIGGER set_count
ON Sailors AFTER INSERT
AS
INSERT INTO StatisticsTable (modifiedTable,
modificationType, count)
SELECT 'Sailors', 'Insert', COUNT(*)
FROM inserted I
WHERE I.age < 18
```

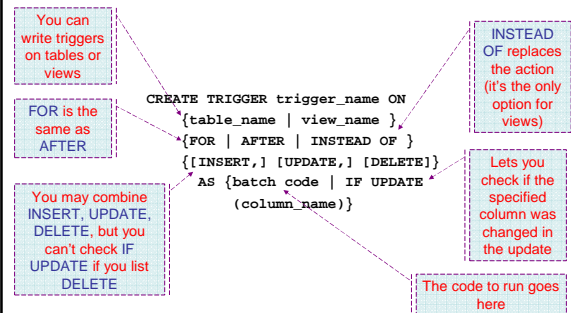
Special table with the new rows

December 29, 2009

ISE 322: Database Systems

21

Trigger Format: SQL Server



December 29, 2009

ISE 322: Database Systems

22

Trigger: SQL Server Example

- Record in the statistics table whenever we raise the ratings of sailors

```
CREATE TRIGGER update_sailors
ON Sailors AFTER UPDATE
AS IF UPDATE(rating)
INSERT INTO StatisticsTable (modifiedTable,
modificationType, count)
SELECT 'Sailors', 'Update', COUNT(*)
FROM inserted I, deleted D
WHERE D.sid = I.sid and I.rating > D.rating
```

What happens if we run UPDATE Sailors SET rating = rating - 1 ?

Only if rating is modified

Records the old rows

December 29, 2009

ISE 322: Database Systems

23

Using Triggers

- One trigger can lead to another
 - Chains
 - Can trigger itself
- The order that they run in is may not be fixed
- MS SQL allows you to set the order of execution for triggers on the same table
 - By default it's the order they were created
 - You can set the actual order using the built in function:

```
EXEC sp_settriggerorder
@triggername='trigger_name',
@order='first', @stmttype='update'
```

LAST | NONE

Action

December 29, 2009

ISE 322: Database Systems

24

Constraints vs Triggers

- Many things with triggers can be done with foreign keys and constraints
- When to use triggers?
 - If a constraint can be fully done with foreign keys cascading/update, unique, use them
 - If the constraint is only on one table, CHECK
 - If the constraint is too complex and inter-table, use ASSERTIONS
- Use triggers if you need to *do something*, not just allow/forbid/update
 - Complex calculations
 - Editing operations
 - Logging

December 29, 2009

ISE 322: Database Systems

25

Conclusion

- Complex integrity constraints
- Triggers and Active Databases

December 29, 2009

ISE 322: Database Systems

26