Foreign Keys

- Pieces of data in different relations are correlated by means of values of primary keys.

- Referential integrity constraints are imposed in order to guarantee that the values refer to existing tuples in the referenced relation.

- A *foreign key* requires that the values on a set X of attributes of a relation R_1 must appear as values for the primary key of another relation R_2.

  - In other words, set of attributes in one relation that is used to `refer' to a tuple in another relation. (Must correspond to primary key of the second relation.) Like a `logical pointer'.

Referential Integrity

☐ E.g. *sid* is a foreign key referring to Students:

   ☐ Enrolled(*sid:* string, *cid:* string, *grade:* string)

   ☐ If all foreign key constraints are enforced, *referential integrity* is achieved, i.e., no dangling references.
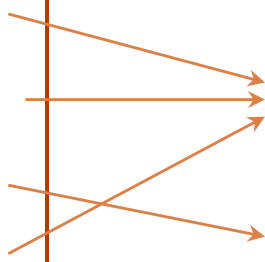
Referential Integrity (cont'd)

☐ Only students listed in the Students relation should be allowed to enroll for courses.

Enrolled

sid	cid	grade
53666	Carnatic101	C
53666	Reggae203	B
53650	Topology112	A
53666	History105	B

Students

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eecs	18	3.2
53650	Smith	smith@math	19	3.8

Enforcing Referential Integrity

- Consider Students and Enrolled; *sid* in Enrolled is a foreign key that references Students.

- What should be done if an Enrolled tuple with a non-existent student id is inserted?   *Reject it!*

- What should be done if a Students tuple is deleted?
  - Also delete all Enrolled tuples that refer to it.
  - Disallow deletion of a Students tuple that is referred to.
  - Set sid in Enrolled tuples that refer to it to a *default sid*.
  - Set sid in Enrolled tuples that refer to it to NULL.

- Similar if primary key of Students tuple is updated.

Where do ICs Come From?

□ ICs are based upon the semantics of the real-world enterprise that is being described in the database relations.

□ We can check a database instance to see if an IC is violated, but we cannot infer that an IC is true by looking at an instance.

  ■ An IC is a statement about *all possible* instances

□ Key and foreign key ICs are the most common; more general ICs supported too.

One More Example

Offences

Code	Date	Officer	Dept	Registration
143256	25/10/1992	567	75	5694 FR
987554	26/10/1992	456	75	5694 FR
987557	26/10/1992	456	75	6544 XY
630876	15/10/1992	456	47	6544 XY
539856	12/10/1992	567	47	6544 XY

Officers

RegNum	Surname	FirstName
567	Brun	Jean
456	Larue	Henri
638	Larue	Jacques

- Offences[Officer] ⊆ Officers[RegNum]

- Offences[Registration,Dept] ⊆ Cars[Registration,Dept]

Cars

Registration	Dept	Owner
6544 XY	75	Cordon Edouard
7122 HT	75	Cordon Edouard
5694 FR	75	Latour Hortense
6544 XY	47	Mimault Bernard

Violation of Foreign keys

Offences

Code	Date	Officer	Dept	Registration
987554	26/10/1992	456	75	5694 FR
630876	15/10/1992	456	47	6544 XY

Officers

RegNum	Surname	FirstName
567	Brun	Jean
638	Larue	Jacques

Cars

Registration	Dept	Owner	…
7122 HT	75	Cordon Edouard	…
5694 FR	93	Latour Hortense	…
6544 XY	47	Mimault Bernard	…

Referential Constraints: Comments

- Referential constraints play an important role in making the relational model value-based.

- Care is needed in case of referential constraints that involve two or more attributes.

Example

Accidents

Code	Dept1	Registration1	Dept2	Registration2
6207	75	6544 XY	93	9775 GF
6974	93	5694 FR	93	9775 GF

Cars

Registration	Dept	Owner	…
7122 HT	75	Cordon Edouard	…
5694 FR	93	Latour Hortense	…
9775 GF	93	LeBlanc Pierre	
6544 XY	75	Mimault Bernard	…

Here we have two referential constraints for **Accidents**:

Registration1, Dept1 to **Cars**
Registration2, Dept2 to **Cars**.