Pointer Swizzling

• As objects are read and written to and from disk, in-memory pointers must be converted to and from on-disk pointers, this is true regardless of whether the DBMS is relational, object-relational, or object-oriented.

• The automatic replacement of a disk pointer by an in-memory pointer by an object-oriented DBMS is referred to as *swizzling*.
**Persistent and Transitory Pointers**

- In a relational *database*, references, or rather pointers between entities are represented by foreign keys, which are maintained primarily by the user or the DBA*.

  - Referred to as *identity by value*.

  - Such references are traditionally *persistent*.

- In an object-oriented *program*, references, or rather pointers between objects are represented by in-memory pointers, which are maintained primarily by the run-time environment*.

  - Referred to as *built-in identity*.

  - Such references are traditionally *transitory*. 

* DBA: Database Administrator
*
Object-Oriented and Relational Representations

- Traditionally:
  - Relational databases are *persistent*; they are maintained on disk.
  - Object-oriented structures are *transitory*; they are maintained in RAM, or rather, virtual memory.

- More Recently:
  - A need has developed for object-oriented structures to be persistent, as well as accessible in virtual memory.
  - This creates the additional need to convert between in-memory object-oriented structures, and structures on disk, be they relational or otherwise.
Object Oriented Terminology

- object => row
- variables/attributes => attributes
- class => entity/table
- messages/methods => functions/procedures
- inheritance
- multiple inheritance
- containment

*Note: Object-oriented terminology and use varies!!!