CSC 370 - Midterm 01 - Conceptual Database Design: Study Sheet

Data Modeling Concepts

Information System Life Cycle Feasibility Study → Requirements → Design → Prototype and Implement → Testing → Operation.

Data-Driven Approach to Information System Design Requirements → Conceptual Design → Conceptual Schema → Logical Design → Logical Schema → Physical Design → Physical Schema.

Classification Represents one concept as a class. Dashed arrow pointing from the instance to the concept. Instance **IS_MEMBER_OF** class.

Aggregation Represents the component parts. Double or Thick arrow pointing from the parts to the concept. (N-Ary Aggregation). Component **IS_PART_OF** concept.

Generalisation Represents inheritance. Split arrow pointing from sub-class to parent class. Sub **IS_A** parent class.

Coverage Properties Exclusive (e), Overlapping (o), Total (t), and Partial (p). Ex. (t, e) = (1, 1), (t, o) = (1, n), (p, e) = (0, 1), (p, o) = (0, n). Where (t, e) is the default.

Cardinality: min-card(C, A) Aggregation of A between class C_1 and C_2. Where if it equals 0 then it's optional and if it is more than 0 then it's mandatory.

Cardinality: max-card(C, A) Aggregation of A between class C_1 and C_2. Where one-to-one is (1, 1), one-to-many (n, 1), many-to-one (1, n), and many-to-many (m, n).

Cardinality: card(C, A) = (m, M) where min-card(C, A) = m and max-card(C, A) = M.

Entity-Relationship Model (ERD)

- *Entity* is a class represented by a rectangle.
- *Attribute* is a property represented by an arrow with a hollow or filled circle.
- *Relationship* is an aggregation of entities (N-Ary) with optional roles (loop) represented by a diamond.

Many-to-Many: Straight arrow, One-to-Many: One sided arrow, and One-to-One: Double sided arrow.

Identifiers, Weak-Entities, and Sub-Classes

Generalisation Hierarchy Is unidirectional and represented by a split arrow pointing to parent class. A *Sub-class* is referred to specialisation. A *Subset* has only one and (p,e) is the default.

Composite Attributes Is an aggregation abstraction. Represented by an oval with properties where attributes are (1,1) by default but can be (0,n).

Identifiers Internal, External, Simple, and Complex. Represented by a line with a filled in circle. Inheritance is applied to sub-classes.

Weak Entity Sets ID attributes are from a strong entity. {some_entity_attributes, strong_entity_identifier_attributes}

Abstraction Mechanisms

Classification Abstraction 1. Entity: Class and properties. 2. Relationship: Class relates Entities. 3. Attribute: Class properties of Entities or Relationship.

Aggregation Abstraction 1. Entity: Aggregation of attributes. 2. Relationship: Aggregation of Entities and Attributes. 3. Composite Attribute: Aggregation of Attributes. 4. Identifier: Aggregation of Identifier Attributes.

Generalisation Abstraction Hierarchy (or Subset) of one Entity set from collection of Entity sets.

Design Methods

Design Process Rigorous and flexible. Transformations: Starting Schema → Resulting Schema. Mapping between concepts. Inheritance of all established logical connections.

Top-Down Design Abstraction and apply primitives for refinement. T_1: Entity → Related Entities. T_2: Entity → Generalisation. T_3: Entity → Uncorrelated Entities. T_4: Relationship → Parallel Relationship. T_5: Relationship → Entity with Relationship. T_6: Attribute Development. T_7: Composite Attribute Development. T_8: Attribute Refinement.

Bottom-Up Design Compile a collection and then aggregate with primitives. B_1: Entity Generation. B_2: Relationship Generation. B_3: Generalisation Generation (Subset Generation). B_4: Attribute Aggregation. B_5: Composite Attribute Aggregation.

Mixed Strategy Divide and Conquer. Top-Down Design focused.

Design Qualities

Completeness: All features. **Correctness**: ER-Model proper and syntactically correct. **Minimality**: Concepts represented once. **Expressiveness**: No explanation. **Readability**: Aesthetics. **Self-Explanation**: Clear schema. **Extensibility**: Modifiable. **Normality**: BCNF, 3NF, and 4NF.

Achieving Minimality An important quality / strength of the relational data model. Helps avoid anomalies such as update, insertion, and deletion.

Cycles: Redundancy in Relationship. *Derived Attributes*: Totals. *Implicit Subsets*: Subset derived from other subsets. *Dangling Sub-Entities*: Sub-Entity not distinguishable by identifier. *Dangling Entities*: Entity with (0,1) Attribute and one connection to another Entity.

Schema Transformation New schema S_2 from input schema S_1.

Information Preserving: Info content is not changed. (Improvement).

Information Changing: *Augmenting*: Information greater than input information. *Reducing*: Information less than input information. *Incompatible*: Otherwise. Correction of conflicting information.