COMP 430
Intro. to Database Systems
Denormalization & Dimensional Modeling
Some consequences of normalization

• Data redundancy is reduced or eliminated.
• Relations are broken into smaller, related tables.
• Using all the attributes from the original relation requires joining these smaller tables.
Denormalization

**Deliberately reintroducing** some redundancy, so that we can access data faster.
Example

Technique: Add duplicate fields

Technique: Add computed fields
### Example

<table>
<thead>
<tr>
<th>ProductSK</th>
<th>ProductID</th>
<th>ColorID</th>
<th>Desc</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Mtn Bike #778</td>
<td>Red</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>Mtn Bike #778</td>
<td>Black</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>Road Bike #123</td>
<td>Red</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>Road Bike #123</td>
<td>Black</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>Road Bike #123</td>
<td>Silver</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>1</td>
<td>Touring Bike #222</td>
<td>Red</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>3</td>
<td>Touring Bike #222</td>
<td>Silver</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>4</td>
<td>Touring Bike #222</td>
<td>Mauve</td>
</tr>
</tbody>
</table>
Less common denormalization techniques

- Duplicating a commonly-used subset of table fields
- Splitting some table rows into different tables
  - Frequently- vs. rarely-used data
  - Data for different regions
  - Common subclasses of data
When to denormalize?

Typically used when some or all of the following apply:

• Many queries need to join the data
• Joining the data is expensive – uses scans, rather than indices
• Computing derived data is expensive – complex queries or complex functions
Dealing with redundant data

Still want data consistency, but now it requires work.
Is full consistency required at all times?

Techniques:
• Stored procedures act as API for updating DB. They add the redundant data.
• Triggers check (and fix?) consistency.
• Application code carefully maintains consistency during updates.
• Reconcile data as background process.
• Reconcile data during system maintenance.
Dimensional modeling

In a tiny, brief nutshell
An alternative to ER modeling

Only a brief overview.
So, we’ll view it through our lens of ER modeling + denormalization.

Emphasizes decision making & use of historical data
• Fast retrieval & aggregation of data
• Less concern with updating data & maintaining consistency while updating
DB design often resembles multiple starflakes

Starflake = tree of junction table & child tables in 1-to-many relationships

Typical:
- Few cycles.
- Super/sub classes implemented only with superclass table.
Starflakes can be compressed to *stars*

Star = junction table & one level of child tables in 1-to-many relationships

Denormalize by joining each child’s tree.
**Fact & dimension tables**

**Facts:** The junction tables are the most important data – the *facts*
- E.g.: store purchases, class enrollments, click data
- Generally the largest tables
- Key data often numeric & additive – e.g., quantity bought, cost per unit, advertisement views

**Dimensions:** The child tables in 1-to-many relationship with facts
- E.g., stores, customers, sales people, sales period
Dimensional modeling process

• Centered on identifying the business model
  • Identifying the potential queries
  • Identifying the facts – the data used in such queries
  • Each query should use only one fact table & its dimension tables.

• Possibly start with an ER model
  • Identify which junction tables serve as fact tables
  • Use only surrogate keys
  • Often add time dimension
  • Denormalize into starflake or star schema
View fact table as $n$-dimensional data cube

Facts are the data in the cube.

Each dimension table represents a dimension of the cube.

Facts might be pre-aggregated along each dimension or combination of dimensions.
Sometimes things are still messy

Not all data fit nicely into facts + 1-to-many dimensions.

Leads to exceptions from this simple presentation.