Notion Database System – Conceptual and Relational Model

DBMS Hand-in Assignment

D1 + TD1 Slot

Name: Shaik Chishti
Reg No: 24BCA7027

Software Chosen: Notion

1 Brief Overview of Notion

Notion is a productivity and knowledge-management software used to plan, organize, and create structured information. It allows users to define highly customizable data structures and aims to replace multiple applications within a single unified workspace.

Everything in Notion is built on the concept of **blocks**. A block may:

- Exist independently
- Contain child blocks
- Be inherited or nested
- Act as both content and structured data

This block-first approach forms the foundation of Notion's database architecture.

2 High-Level Architecture

Notion's database system is composed of the following core entities:

- Databases
- Pages
- Blocks
- Database Properties
- Properties & Values

- Relations

Each entity is stored independently and connected via explicit relationships, allowing scalability and schema flexibility.

3 Entity Descriptions

3.1 Databases

A Database defines the schema and structure. It does not store rows directly; instead, it owns pages.

Key Fields:

- object (string)
- id (UUIDv4)
- created_time
- created_by
- last_edited_time
- title
- description
- icon, cover
- parent
- url, public_url
- archived, in_trash, is_inline

3.2 Pages

Pages act as rows within databases and store property values and content blocks.

3.3 Blocks

Blocks are the atomic units of Notion. Pages themselves are blocks, and blocks may contain child blocks, enabling deep nesting.

3.4 Database Properties

These define the schema of a database (title, select, relation, rollup, formula, etc.).

3.5 Properties & Values

These store actual data values for each page and reference database properties.

3.6 Relations

Relations define many-to-many links between pages and behave as graph edges rather than traditional foreign keys.

4 ER Diagram (Conceptual Model)

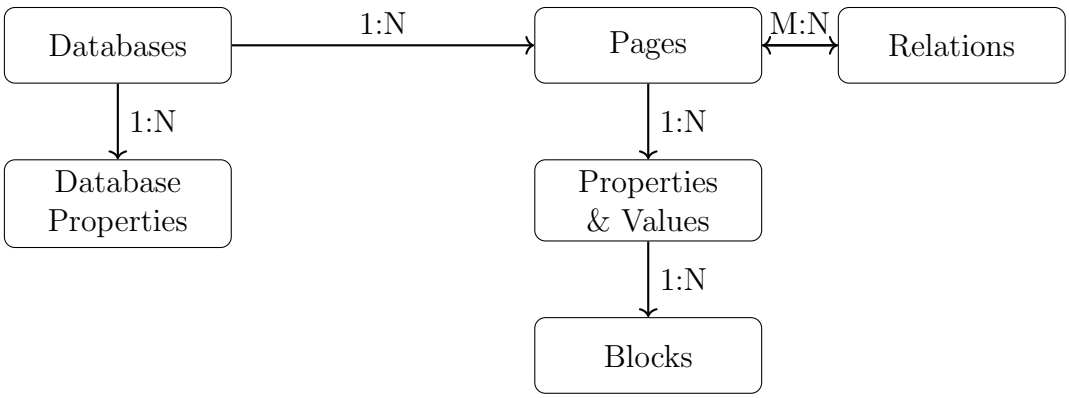

Figure 1: ER Diagram of Notion Database Architecture

5 Formal Relational Schema

Table	Attributes
Databases	`database_id (PK)`, object, title, description, created_time, created_by, last_edited_time, parent, url, archived
Pages	`page_id (PK)`, database_id (FK), created_by, last_edited_time, archived, icon, cover, url
Blocks	`block_id (PK)`, parent_id (FK), page_id (FK), type, created_time, last_edited_time, has_children
Database_Properties	`property_id (PK)`, database_id (FK), name, description, type

Property_Values	`value_id (PK)`, page_id (FK), property_id (FK), value_json
Relations	`relation_id (PK)`, from_page_id (FK), to_page_id (FK), relation_type

6 Design Insights

- Everything is represented as blocks
- Database rows are pages, not tuples
- Schema is metadata-driven
- Relations form a graph structure
- Rollups and formulas are computed dynamically

This design enables Notion to combine structured databases with rich content seamlessly.

7 References

- https://developers.notion.com
- https://developers.notion.com/reference
- https://www.notion.com/help/database-properties
- https://www.notion.com/help/relations-and-rollups
- Personal Documentation