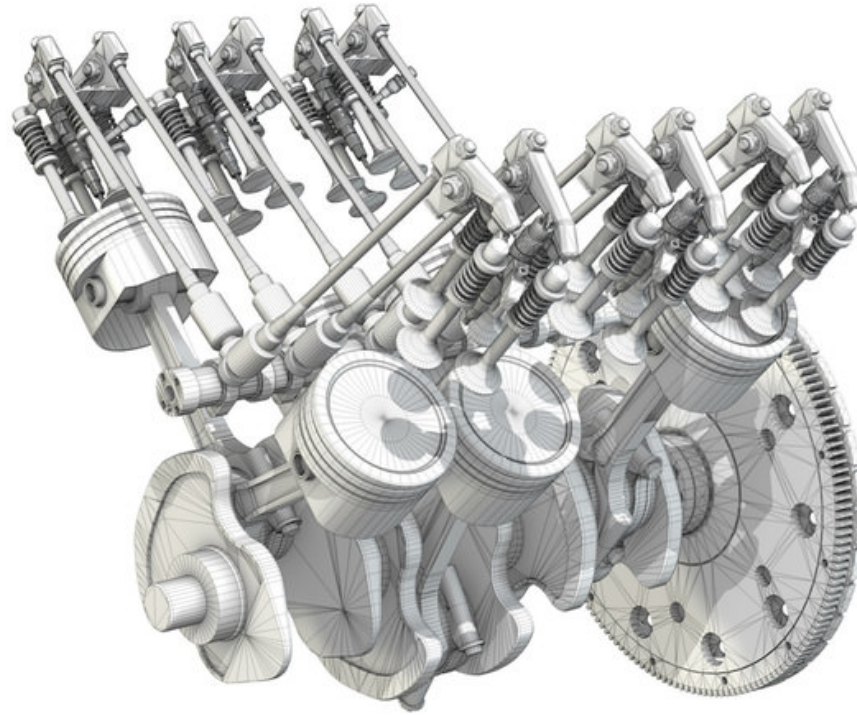


Neo4j Internals



Disk Storage

- Neo4j stores graph data in a number of different store files
- Each file contains the data for a specific part of the graph:
 - nodes
 - relationships
 - properties
 - relationship types etc
- Record-based files per data structure (nodes, relationships, properties etc)

Disk Storage

- The blocks have different sizes according to the type of record they correspond to:
 - **Node records:** labels, pointer to first relationship, pointer to first property block.
 - **Relationship records:** relationship-type, pointers to: start-node, end-node, first property block, next-relationship and previous relationship for both start and end-nodes
 - **Property records:** up to four property entries, each with a property-key id, and an encoded / compressed property value or pointer to string or array storage list

Disk Storage

- Each node and relationship record block is of fixed size

Node record block: 9 bytes

A	B	C
---	---	---

A: In use flag

B: ID of first relationship

C: ID of first property of node

Relationship record block: 33 bytes

A	B	C	D	E	F	G	H	I
---	---	---	---	---	---	---	---	---

A: In use flag

B: ID of start node of relationship

C: ID of end node of relationship

D: pointer to relationship type

E: ID of next relationship of start node

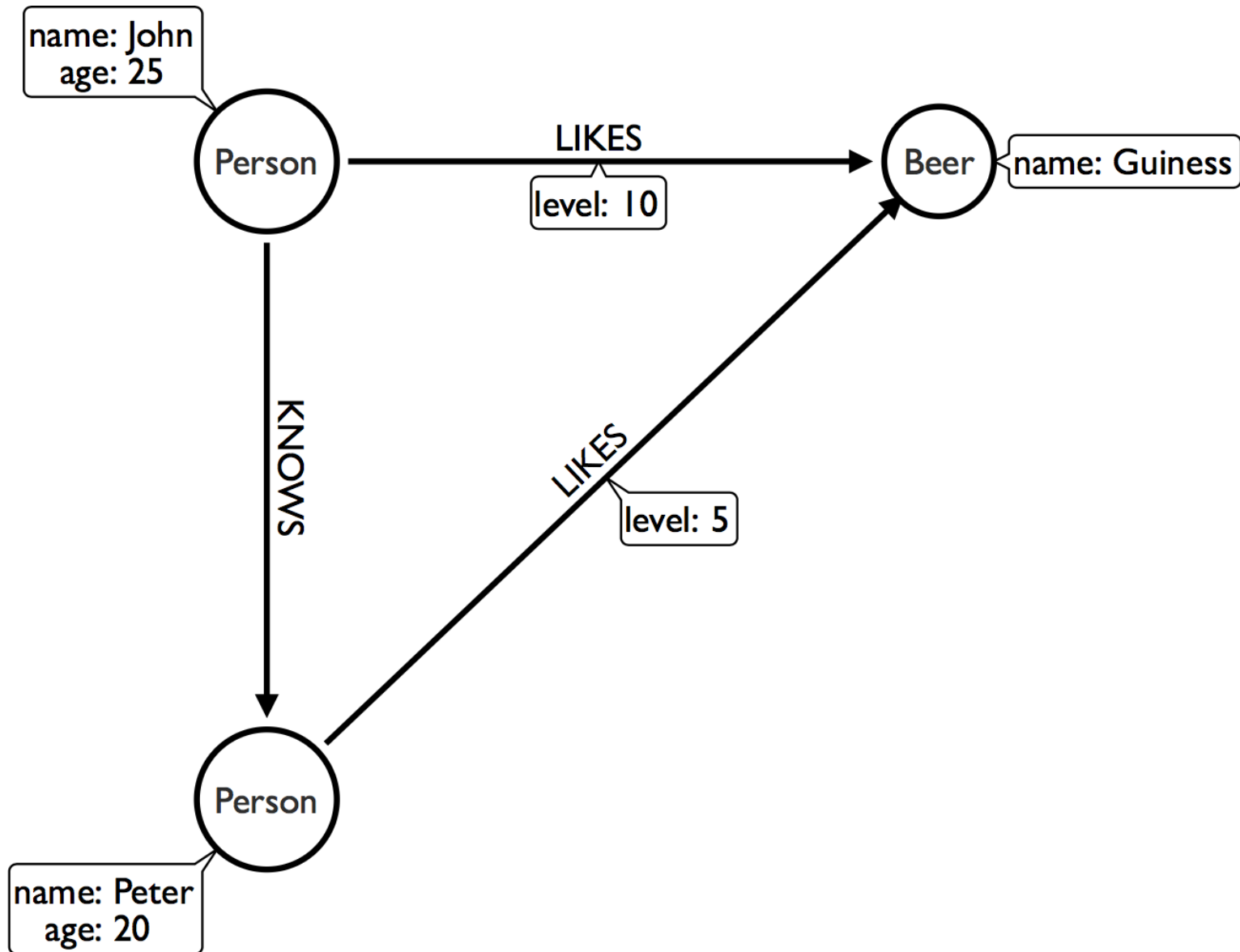
F: ID of prev relationship of start node

G: ID of next relationship of end node

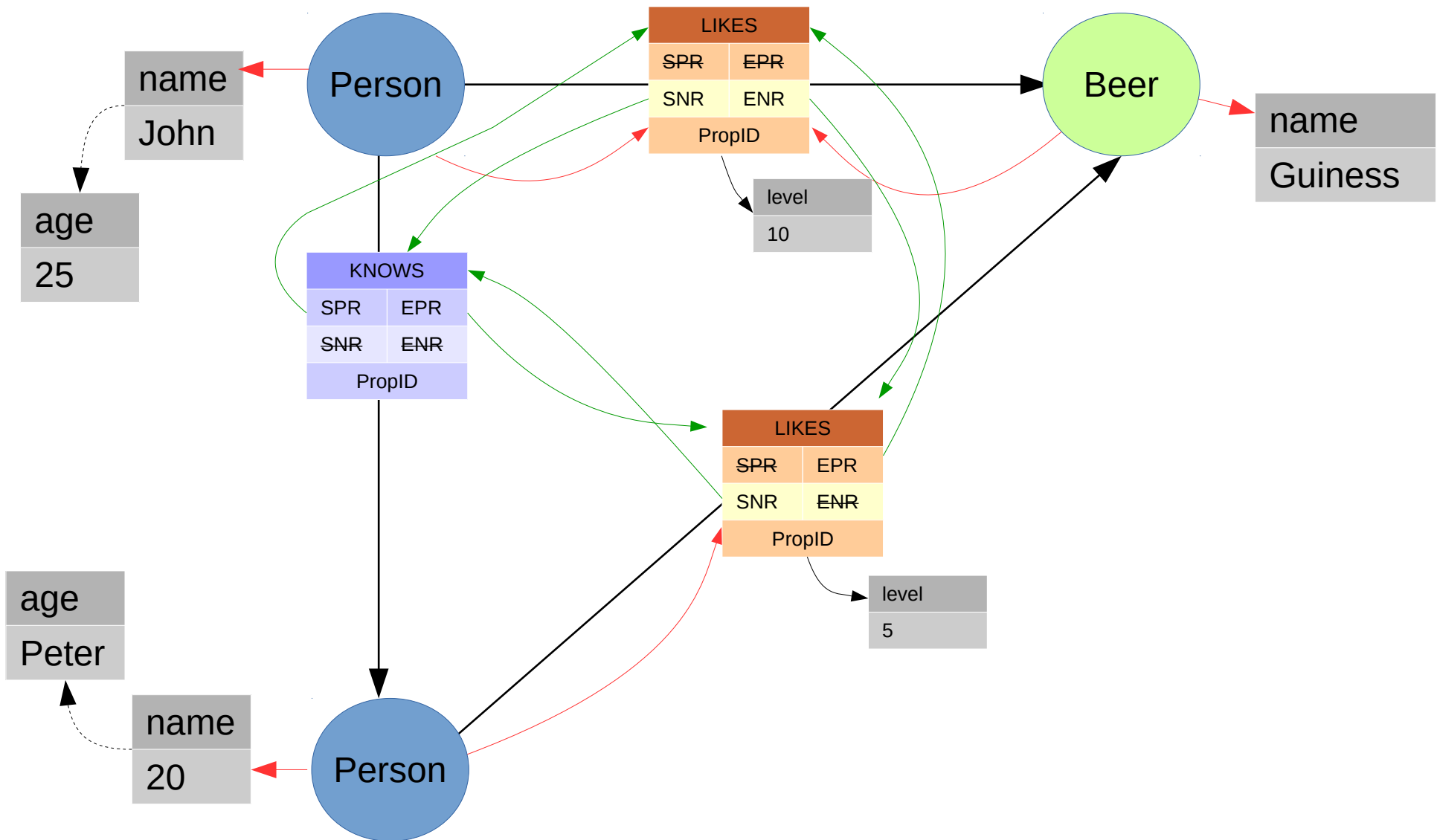
H: ID of prev relationship of end node

I : ID of first property

Example



Example



Disk Storage

- Each node and relationship record block is directly addressable by its id
- Fixed size records enable fast lookup
 - Node/relationship id * record size → location in the corresponding file
 - ID's are recycled to avoid fragmentation
- Neo4j employs additional caches for loaded Node and Relationship-Records for faster retrieval