Database Schema

The questions are based on the following application about a company's operation. Its ER data model is shown below with the following constraints.

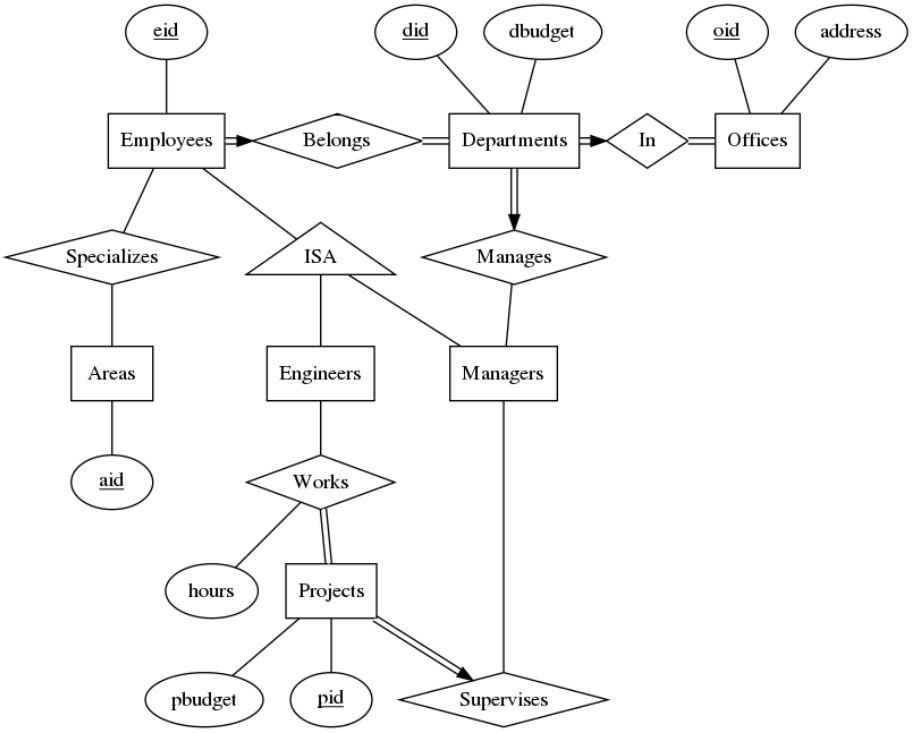

Figure 1: ER Model

The company has at least one office. Each office (identified by oid with location specified by address) consists of one or more departments. Each department (identified by did with a budget dbudget) is located in one office and has one or more employees. Each employee (identified by eid) must belong to a department. The application focuses on two subclasses of employees: engineers and managers. An employee can be neither an engineer nor a manager, and no employee can be both an engineer and a manager. Each employee can specialize in 0 or more areas (identified by aid). Each department must be managed by exactly one manager, and each manager can manage 0 or more departments. Each engineer can work in 0 or more projects, and there must be at least one engineer working in each project. For each project P that an engineer E works on, the number of hours per week that E spends on P is given by hours. Each project (identified by pid with a budget pbudget) must be supervised by exactly one manager. A manager can supervise 0 or more projects. Attributes hours, dbudget and pbudget have non-null values.

Relational Schema

The following is the relational schema for this application.

```
CREATE TABLE Offices (
    oid             INTEGER,
    address         VARCHAR(60),
    PRIMARY KEY (oid)
);



/* eid = eid of department's manager */
CREATE TABLE Departments (
    did             INTEGER,
    dbudget         INTEGER NOT NULL,
    oid             INTEGER NOT NULL,
    eid             INTEGER NOT NULL,
    PRIMARY KEY (did),
    FOREIGN KEY (oid) REFERENCES Offices
);

CREATE TABLE Employees (
    eid             INTEGER,
    did             INTEGER NOT NULL,
    PRIMARY KEY (eid),
    FOREIGN KEY (did) REFERENCES Departments
);

CREATE TABLE Engineers (
    eid             INTEGER,
    PRIMARY KEY (eid),
    FOREIGN KEY (eid) REFERENCES Employees
);

CREATE TABLE Managers (
    eid             INTEGER,
    PRIMARY KEY (eid),
    FOREIGN KEY (eid) REFERENCES Employees
);
```

```
/* eid = eid of project's supervisor  */
CREATE TABLE Projects (
    pid             INTEGER,
    pbudget         INTEGER NOT NULL,
    eid             INTEGER NOT NULL,
    PRIMARY KEY (pid),
    FOREIGN KEY (eid) REFERENCES Managers
);


CREATE TABLE Works (
    pid             INTEGER,
    eid             INTEGER,
    hours           INTEGER NOT NULL,
    PRIMARY KEY (pid,eid),
    FOREIGN KEY (eid) REFERENCES Engineers,
    FOREIGN KEY (pid) REFERENCES Projects
);

CREATE TABLE Areas (
    aid             VARCHAR(5),
    PRIMARY KEY (aid)
);

CREATE TABLE Specializes (
    eid             INTEGER,
    aid             VARCHAR(5),
    PRIMARY KEY (eid,aid),
    FOREIGN KEY (eid) REFERENCES Employees,
    FOREIGN KEY (aid) REFERENCES Areas
);
```