Table of Contents

Phase I: Problem Definition and Database Design

1. Problem Statement

There is a contract out from NASA to help facilitate efficient information generation for university researchers and amateur student astrologists. The Space Database will fulfill this goal, acting as an encyclopedia for objects in space.

Users of the database will have two roles. Researchers will be able to update the database information and view the current instance of the database, including other researcher's changes. This way new discoveries can be cataloged by relevant professionals in the field.

Students will be able to view the current instance of the database. Therefore, the Space Database will provide them with an easy way to query some of the most recent discoveries made by researches as well as reference long-known space objects.

The Space Database will contain the entities: planets, stars, and satellites.

Stars will have the attributes:

- The solar system that houses it
- Its temperature (in whole Kelvin)

- Its date of discovery (in the middle-endian numerical format MMDDYYYY)
- Who discovered it
- Its distance from Earth (in light years)
- Its classification, or spectral type, based on the MK system. This follows the format character, integer, character* (ie. A0IV where A is a character, 0 is an integer, I is a character, V is a character)

Planets will have the attributes:

- Name
- The solar system that houses it
- Its distance from the star it orbits(in whole kilometers)
- Its size (represented by its radius in whole kilometers)
- Who it was discovered by

Satellites will have the attributes:

- A unique identifier
- What planet it orbits
- Its distance from the planet it orbits (in whole kilometers)
- A type (Natural/Artificial)

Additionally, satellites will be either natural or artificial. Ergo:

Moons (natural) will have the attributes:

- a unique identifier
- its size (represented by its radius in kilometers)

Artificial satellites will have the attributes:

- a unique identifier
- who launched it
- when it was launched (in the middle-endian numerical format MMDDYYYY)
- its cost (in whole US dollar amount)

The database will include the above attributes for each entity. The summation will act as a fact sheet representing space’s general information. Furthermore, the relationships between the entities reflect spatial relationships.

The following are the relationships and shared attributes between stars and planets that will be reflected:

- Planets revolve around a star
  - A star may have none or many planets revolving it
  - A planet will have only one star that it revolves around. This implies a planet is only part of one solar system

between satellites (natural or artificial) and planets that will be reflected:

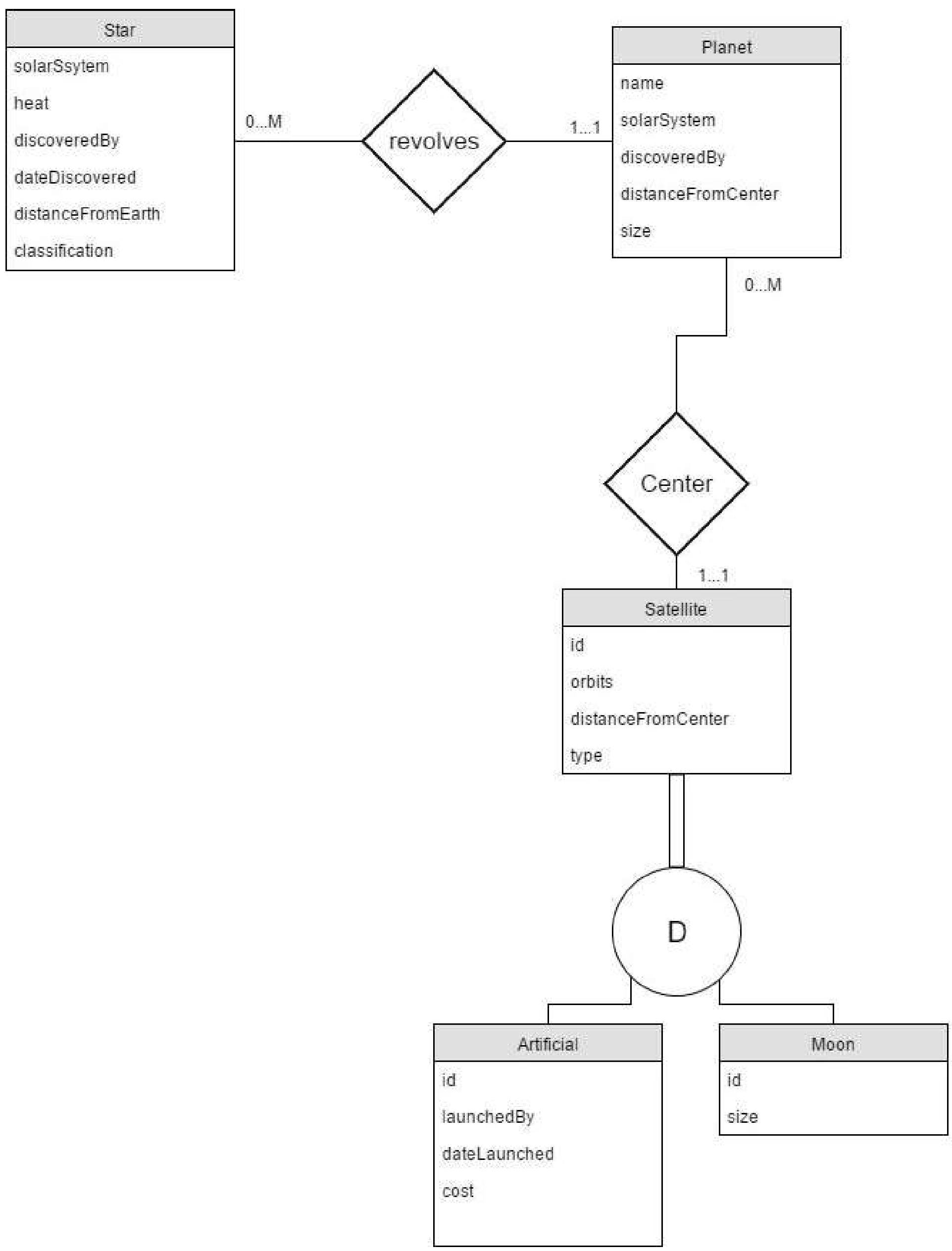
- Satellites revolve around a planet
  - A planet may have none or many satellites
  - A satellite will have one planet

The organizations that will utilize this database will need to have access to various space facts about the majority of stellar anomalies. In addition that the space facts, each space fact has to be indivisible so that information can be generated from the raw data.

Ultimately, Space Database will be used as a resource for students to expand their knowledge of the space surrounding

Earth and the objects it contains. It will increase the students' spatial awareness by making these object relevant to their home planet and make the universe's most distant stars and planets be but one *query* away.

2. Entity-Relationship Diagram



3. Relational Model

Star	
PK	solarSystem
	heat
	dateDiscovered
	distanceFromEarth
	classification

StarDisc	
PK1	solarSystem
PK2	discoveredBy

Revolves	
PK1	planetName
PK2	solarSystem

Planet	
PK	name
FK	solarSystem
	distanceFromCenter
	size

PlanetDisc	
PK1	planetName
PK2	discoveredBy

Center	
PK1	planetName
PK2	satID

Satellite	
PK	id
FK	orbits
	distanceFromCenter
	type

Artificial	
PK,FK	id
	launchedBy
	dateLaunched
	cost

Moon	
PK,FK	id
	size

Relational Model Cont. (Attributes with types, domain, and constraints PKs and FKs (dashed underline):

STAR
Star.solarSystem VARCHAR2(20)
Star.heat VARCHAR2(10)
Star.dateDiscovered
Star.distanceFromEarth
Star.classification

STARDISC (MultiVar)
StarDisc.solarSystem
StarDisc.discoveredBy

PLANET
Planet.name
Planet.solarSystem

Planet.distanceFromCenter
Planet.size

PLANETDISC (MultiVar)
PlanetDisc.planetName
PlanetDisc.discoveredBy

REVOLVES
Revolves.planetName
Revolves.solarSystem

SATELLITE
Satellite.id
Satellite.orbits

Satellite.distanceFromCenter
Satellite.type

ARTIFICIAL
Artificial.id

Artificial.launchedBy
Artificial.dateLaunched
Artificial.cost

MOON
Moon.id

Moon.size

CENTER
Center.planetName
Center.satID

4. Functional Dependencies

Functional dependencies include

(1)Star.solarSystem →
    Star.heat
    Star.dateDiscovered
    Star.distanceFromEarth
    Star.classification

(2)StarDisc.planetName →→
    StarDisc.discoveredBy

(3)Planet.name →
    Planet.solarSystem
    Planet.distanceFromCenter
    Planet.size

(4)PlanetDisc.planetName →→
    PlanetDisc.discoveredBy

(5)Satellite.id →
    Satellite.orbits
    Satellite.distanceFromCenter
    Satellite.type

(6)Artificial.id →
    Artificial.launchedBy
    Artificial.dateLaunched
    Artificial.cost

(7)Moon.id →
    Moon.size

5. Normalization

The canonical cover of all attributes is the concatenation of all primary keys in each relation. If you provide the attributes:

    Star.solarSystem
    StarDisc.discoveredBy
    Planet.name
    PlanetDisc.discoveredBy
    Satellite.id

You will receive a unique tuple if the whole diagram was merged into one relation. However, due to the dependencies we can minimize redundancies and create multiple tables utilizing normalization.

All relations satisfy 1st normal form because all attributes are atomic. Each attribute maintains a domain specific to that attribute.

All relations satisfy 2nd normal form because the candidate key and original single relation:

    Star.solarSystem
    StarDisc.discoveredBy
    Planet.name

PlanetDisc.discoveredBy
Satellite.id

was broken up into multiple relations as seen in the relational model above. This decomposition created individual candidate keys for each relation. Now all relations satisfy all FDs and there are no partial dependencies in the functional dependencies. This generated a total of five relations.

All relations satisfy 3NF--BCNF. This is because there are no transitive functional dependencies.

All relations did not satisfy 4NF because there could have been more than one discoverer of a particular planet and or star. Due to this functional dependencies, the STAR and PLANET relation was decomposed further to satisfy 4NF. The resulting relations include PLANET (with omitted “discoveredBy” attribute), STAR (with omitted discoveredBy attribute), PlanetDisc, and StarDisc. The resulting candidate keys for each decomposed relation is PLANET.name for PLANET, STAR.solarSystem for STAR, PLANETDISC.name&&PLANETDISC.discoveredByfor PLANETDISC, and lastly STARDISC.solarSystem&&STARDISC.discoveredByfor STARDISC. The final decomposition and set of relations included nine relations. These nine relations satisfy the five single variable functional dependencies and two multivariable functional dependencies.

6. Example Data

Star

solarSystem	heat	distFromEarth	classification
Home	5778	0.00001581	G2V
Carina	6998	310	F0II
Aldebaran	3910	65	K5III
Piscis	8590	25	M4V
Hamal	4480	65.8	K2III

Planet

name	solarsystem	size	distanceFromCenter
Earth	Home	6371	1496000
Bespin	Anoat	118000	1250000
Coruscant	Carina	12240	1482000000
Geonosis	Arkanis	11370	1320000000
Saturn	Home	58232.503	1429000000

Satellite

id	orbits	distanceFromCenter	type
Moon	Earth	384400	natural
Europa	Jupiter	780000000	natural
Ganymede	Jupiter	2631	natural
Enceladus	Saturn	238000	natural
Rishi	J1407b	3250	natural
Titan	Saturn	1221870	natural
DS-1 Orbital Battle Station	Geonosis	1737	artificial

Moon

id	size
Moon	1737
Europa	1500
Ganymede	2631
Callisto	2410
Io	1821
Titan	2576
Enceladus	252
Rishi	1500

Artificial

id	launchedBy	dateLaunched	cost
DS-1 Orbital Battle Station	Darth Vader	07012222	1000000000000
Cassini	Jet Prop. Laboratory	10151997	3260000000

planetDisc

planetName	discoveredBy
Bespin	Lando Calrissian
Earth	Lucy
Coruscant	Sheev Palpatine
Geonosis	Darth Vader
Saturn	Cronus

starDisc

solarSystem	discoveredBy
Home	Lucy
Carina	King Menelaus
Aldebaran	Edmund Halley
Piscis	Wilhuff Tarkin
Hamal	Ben Kenobi

Center

planetName	satID
Earth	Moon
Jupiter	Europa
Jupiter	Ganymede
Saturn	Enceladus
Saturn	Titan
J1407B	Rishi
Geonosis	DS-1 Orbital Battle Station

Revolves

planetName	solarSystem
Earth	Home
Bespin	Anoat
Coruscant	Carina
Geonosis	Arkanis
Saturn	Home