CS 440 Final Project Report

Project Title: International Satellite Database

Group Members

Alexander Hull (hullale@oregonstate.edu)
Ryan Crane (cranery@oregonstate.edu)
John Magenheimer (magenhej@oregonstate.edu)
Cooper Hutchens (hutchico@oregonstate.edu)

Project Description

Merge tables containing data on satellites with varying purposes and origins to facilitate easily aggregating information about them and illustrate trends across time periods and over different countries. We used N2YO as our dataset. Although the website allows for selecting particular categories or countries, it is very difficult to view trends or more refined queries. We used the Beautiful Soup Python library to scrape the data from n2yo.com. N2YO maintained satellites in two tables organizing them by categories and by the country or agency that operated them. We decomposed these tables into several relations with some foreign keys defined to represent important relationships between data elements.

Data Acquisition

The N2YO website was our source for satellite data. N2YO's database contains data on over 43000 satellites organized predominantly by agency and by category. N2YO provides a public API to access their database however it does not provide a method that returns all the information we need for a given satellite. Additionally it imposes a rate limit on API calls which would have bottlenecked our progress, so we decided to scrape the data we needed directly from the website. We initially tried simple get requests for the raw html documents, however we found that their website used scripts that executed on the client side to populate their tables with data. To solve this we used Beautiful Soup. Once it collects a chunk of data, the python program pulls the relevant information and adds it to a csv file. Next, this csv file can be imported through the PHPMyAdmin website. Two python programs were used to pull data, one collects satellites based on agency, and the other by category.

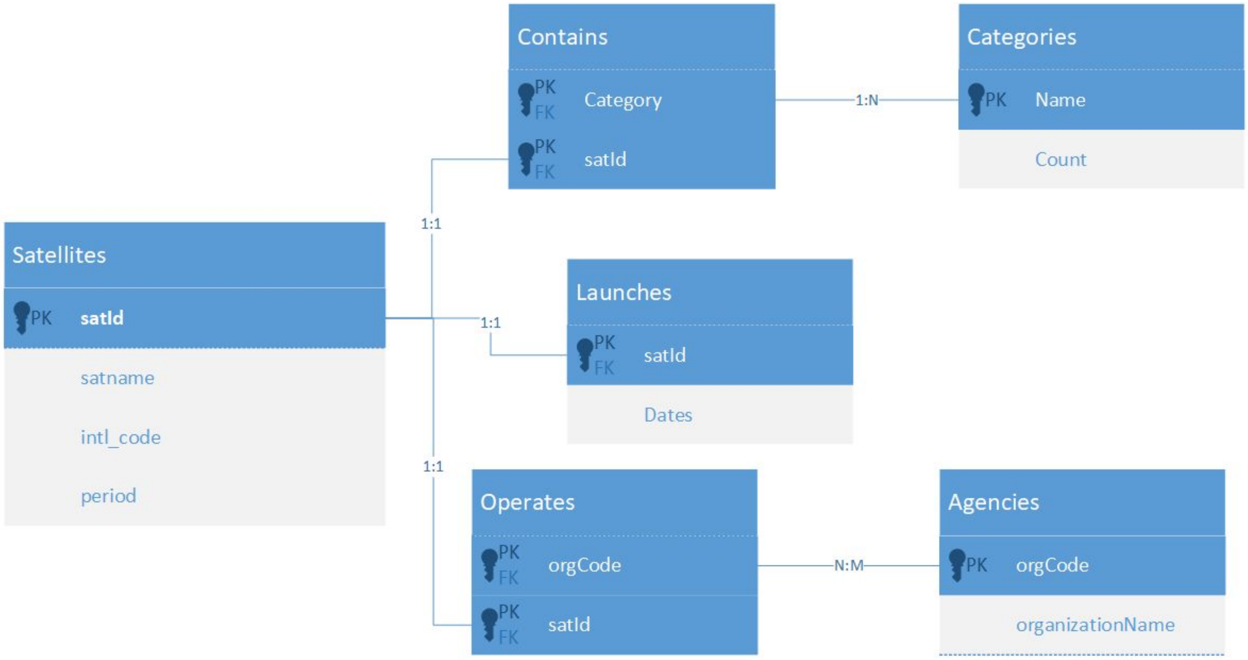
Data Cleaning

The data gathered by the Python programs into csv files needed to be reformatted. The satellites in the two datasets derived from the csv files--satellites_by_category and satellites_by_agency--needed to be merged into a single table, Satellites(satId, satname, intl_code, period), whereas the categories and agencies needed to be isolated into their own tables.
Columns in the csv files were, by default, of type varchar; the launch dates and period attributes needed to converted into date types and decimals respectively.

When adding primary and foreign keys to the Satellites table (in order to use the satId in the Contains, Launches, and Operates relations), we noticed a constraint violation--there were null satId entries in the Contains and Operates relations; the Python program responsible for gathering satellite data by category had captured satellites with null values. To remedy this violation, satellites were deleted from the Contains relation if they did not contain a satId from the Satellites table.

Database Design

The primary table is the Satellites table, which contains a unique ID pulled from the data source, along with a name, international naming convention, and period. There exists three relational tables: contains, launches, and operates. Lastly, there are two tables, categories and agencies. Below is our schema design.



Database Documentation

To use the international satellites database, import the satelliteDB.sql file into your PHPmyadmin account or access the classmysql database:

1. Connect to the OSU campus server.
2. Go to https://tools.engr.oregonstate.edu/phpMyAdmin/index.php?server=5
3. Enter ‘cs440_magenhej’ in the Username field.
4. Enter ‘6782’ in the Password field.
5. Select ‘classmysql’ from the dropdown menu of the Server Choice field.
6. Press Go.

The queries written for this database are in procedures, located in PHPmyadmin’s ‘Routines’ tab. Pressing a procedure’s execute button will, in most cases, prompt the user to input values for the procedure’s parameters. Below are the procedures and instructions for using them. Refer to the *Agency List* to find the organization codes when needed; refer to the *Categories List* to find the category names when needed.

Procedures

- **AllSatellitesLaunchedBetweenTimes(YEAR t1, YEAR t2):** To output a list of all satellites launched over a time range, input a starting year **t1** and an ending year **t2** in the form YYYY.
- **CategoriesOperatedByAgency(VARCHAR agency):** To list the categories of satellites operated by an agency and the respective total count of satellites within each category, input the organization code **agency** of the agency you wish to observe; this input must be in all caps.
- **CategoryAveragePeriod(VARCHAR cat):** To compute the average period (the time in minutes it takes for a satellite to orbit the Earth) of satellites within a category, input the category name **cat** of the category you wish to observe; this input must be in all caps.
- **CountAllSatellitesLaunchedBetweenTimes(YEAR t1, YEAR t2):** To output the count of all satellites over a time range, input a starting year **t1** and an ending year **t2** in the form YYYY.
- **CountSatellitesOfCategoryLaunchedBetweenTimes(VARCHAR cat, YEAR t1, YEAR t2):** To output the number of satellites belonging to a category, launched over a time range, input the category name **cat** of the category you wish to observe (this input must be in all caps) and the starting year **t1** and ending year **t2** both in the form YYYY.
- **GetSatellitesByCategoryAgencyTime(VARCHAR cat, VARCHAR agency, YEAR t1, YEAR t2):** To output all satellites of a specified category, launched by a specified organization, over a specified period of time, input the category name **cat** of the category you wish to observe (this input must be in all caps), the agency code **agency** of the agency you wish to observe (this input must be in all caps), and the starting year **t1** and ending year **t2** both in the form YYYY.
- **RankAgenciesByCategory(VARCHAR cat):** To output a list of agencies ranked by the total number of satellites they have launched of a specified

satellite category, input the category name **cat** of the category you wish to observe (this input must be in all caps).

- **SatellitesLaunchedByAgencyBetweenTimes(VARCHAR agency, YEAR t1, YEAR t2):** To output a list of satellites launched by an agency over a specified time range, input the agency code **agency** of the agency you wish to observe (this input must be in all caps) and the starting year **t1** and ending year **t2** both in the form YYYY.
- **SatellitesLaunchedByOrganizationPerYear(VARCHAR agency):** To output a list of counts for the number of satellites launched by a specified agency per year, input the agency code **agency** of the agency you wish to observe (this input must be in all caps).
- **SatellitesLaunchedPerYear():** This procedure will output a list of counts for all satellites launched per year from the year of the first satellite launch (1958) to the present.
- **SatellitesOfCategoryLaunchedBetweenTimes(VARCHAR cat, YEAR t1, YEAR t2):** To output a list of satellites belonging to a specified category over a specified time range, input the category name **cat** of the category you wish to observe (this input must be in all caps) and the starting year **t1** and ending year **t2** both in the form YYYY.
- **INSERT_SATELLITE(INT satId, VARCHAR satname, VARCHAR intl_code, DECIMAL period, VARCHAR category, DATE dates, VARCHAR orgcode):** To insert a new satellite into the database, provide inputs for the satId, satname, intl_code, period, category, dates, and orgcode. The orgcode and category must already exist in the agencies and categories tables.

Appendix

Agency List

organizationName	orgCode
ARAB SATELLITE COMMUNICATIONS ORGANIZATION	AB
ASIA SATELLITE TELECOMMUNICATIONS COMPANY (ASIASAT)	AC
ALGERIA	ALG
ARGENTINA	ARGN
AUSTRALIA	AUS
AZERBAIJAN	AZER
BELARUS	BELA
BULGARIA	BGR
BOLIVIA	BOL

BRAZIL	BRAZ
BHUTAN	BT
CANADA	CA
CHINA/BRAZIL	CHBZ
CHILE	CHLE
COMMONWEALTH OF INDEPENDENT STATES (FORMER USSR)	CIS
COSTA RICA	CRI
CZECHIA	CZ
CZECH REPUBLIC (FORMER CZECHOSLOVAKIA)	CZCH
DENMARK	DEN
ECUADOR	ECU
EGYPT	EGYP
EUROPEAN SPACE AGENCY	ESA
ESTONIA	EST
EUROPEAN ORGANISATION FOR THE EXPLOITATION OF METEOROLOGICAL SATELLITES	EUME
EUROPEAN TELECOMMUNICATIONS SATELLITE ORGANIZATION	EUTE
FRANCE/GERMANY	FGER
FRANCE	FR
FRANCE/ITALY	FRIT
GERMANY	GER
GLOBALSTAR	GLOB
GREECE	GREC
INTERNATIONAL MOBILE SATELLITE ORGANIZATION (INMARSAT)	IM
INDIA	IND
INDONESIA	INDO
IRAQ	IRAK

IRAN	IRAN
ISRAEL	ISRA
INTERNATIONAL SPACE STATION	ISS
ITALY	IT
INTERNATIONAL TELECOMMUNICATIONS SATELLITE ORGANIZATION	ITSO
JAPAN	JPN
KAZAKHSTAN	KAZ
KENYA	KEN
LAOS	LAOS
LITHUANIA	LTU
LATVIA	LTV
LUXEMBOURG	LUXE
MOROCCO	MA
MALAYSIA	MALA
MEXICO	MEX
NORTH ATLANTIC TREATY ORGANIZATION	NATO
NETHERLANDS	NETH
NEW ICO	NICO
NIGERIA	NIG
NORTH KOREA	NKOR
NORWAY	NOR
O3B NETWORKS	O3B
ORBCOMM	ORB
PAKISTAN	PAKI
PERU	PER
POLAND	POL
PORTUGAL	POR

PEOPLE'S REPUBLIC OF CHINA	PRC
REGIONAL AFRICAN SATELLITE COMMUNICATIONS ORGANIZATION	RASC
TAIWAN (REPUBLIC OF CHINA)	ROC
PHILIPPINES (REPUBLIC OF THE PHILIPPINES)	RP
SOUTH AFRICA	SAFR
SAUDI ARABIA	SAUD
SEA LAUNCH	SEAL
SOCIETE EUROPEENNE DES SATELLITES	SES
SINGAPORE	SING
SLOVAKIA	SK
SOUTH KOREA	SKOR
SPAIN	SPN
SINGAPORE/TAIWAN	STCT
SWEDEN	SWED
THAILAND	THAI
TURKMENISTAN/MONACO	TMMC
TURKEY	TURK
UNITED ARAB EMIRATES	UAE
UNITED KINGDOM	UK
URUGUAY	URY
UNITED STATES	US
UNITED STATES/BRAZIL	USBZ
VENEZUELA	VENZ
VIETNAM	VTNM

Category List

- AMATEUR RADIO
- BEIDOU NAVIGATION SYSTEM
- BRIGHTEST
- CELESTIS
- CUBESATS
- DISASTER MONITORING
- EARTH RESOURCES
- EDUCATION
- ENGINEERING
- EXPERIMENTAL
- FLOCK
- GALILEO
- GEODETIC
- GEOSTATIONARY
- GLOBAL POSITIONING SYSTEM (GPS) CONSTELLATION
- GLOBAL POSITIONING SYSTEM (GPS) OPERATIONAL
- GLOBALSTAR
- GLONASS CONSTELLATION
- GLONASS OPERATIONAL
- GOES
- GONETS
- GORIZONT
- INTELSAT
- IRIDIUM
- IRNSS
- ISS
- LEMUR
- MILITARY
- MOLNIYA
- NAVY NAVIGATION SATELLITE SYSTEM
- NOAA
- O3B NETWORKS
- ORBCOMM
- PARUS
- QZSS
- RADAR CALIBRATION
- RADUGA
- RUSSIAN LEO NAVIGATION
- SATELLITE-BASED AUGMENTATION SYSTEM
- SEARCH & RESCUE
- SPACE & EARTH SCIENCE
- STRELA
- TRACKING AND DATA RELAY SATELLITE SYSTEM
- TSELINA
- TSIKADA
- TSIKLON
- TV
- WEATHER
- WESTFORD NEEDLES
- XM AND SIRIUS
- YAOGAN