Rockbuster Stealth

Data Dictionary & Appendix

Table of Contents

Fact Tables

payment

Columns	Data Type	Description
payment_id	SERIAL	Surrogate **primary key**, uniquely identifying each payment transaction in sequential order.
customer_id	SMALLINT	**Foreign key**, uniquely identifying an individual customer
staff_id	SMALLINT	**Foreign key**, uniquely identifying an individual store employee
rental_id	INTEGER	**Foreign key**, uniquely identifying each rental transaction (i.e., each instance of a particular movie rental)
amount	NUMERIC(5,2)	Payment amount, stored with up to five digits before the decimal point and two digits after
payment_date	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the payment, stored with a precision of up to six digits of fractional seconds

Links to the following tables: rental, customer, staff.

rental

Columns	Data Type	Description
rental_id	SERIAL	Surrogate **primary key**, uniquely identifying each rental transaction (i.e., each instance of a particular movie rental)
rental_date	TIMESTAMP(6) WITHOUT TIME ZONE	Date of the rental transaction, stored with a precision of up to six digits of fractional seconds
inventory_id	INTEGER	**Foreign key**, uniquely identifying each physical copy of a film in inventory.

customer_id	SMALLINT	**Foreign key**, uniquely identifying an individual customer
return_date	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the movie return by the customer, stored with up to six digits of fractional seconds precision.
staff_id	SMALLINT	**Foreign key**, uniquely identifying an individual store employee
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to the following tables: inventory, customer, staff.

Links from: payment.

Dimension Tables

customer

Columns	**Data Type**	**Description**
customer_id	SERIAL	Surrogate **primary key**, uniquely identifying an individual customer
store_id	SMALLINT	**Foreign key**, uniquely identifying a physical story
first_name	CHARACTER VARYING(45)	First name of the individual customer, with a maximum length of 45 characters.
last_name	CHARACTER VARYING(45)	Last name of the individual customer, with a maximum length of 45 characters.
email	CHARACTER VARYING(50)	Email address, with a maximum of 50 characters.
address_id	SMALLINT	**Foreign key**, uniquely identifying a mailing address
activebool	BOOLEAN	Boolean value indicating whether the address is still in use by the customer (T/F)
create_date	DATE	Date when the customer record was created, i.e. first interaction or account creation date, stored

		in YYYY-MM-DD format.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.
active	INTEGER	Integer flag indicating whether the customer account or record is active.

Links to: address.

Links from the following tables: payment, rental.

address

*note: both Customer and Store reference address_id.

Columns	**Data Type**	**Description**
address_id	SERIAL	Surrogate **primary key**, uniquely identifying a mailing address, which can represent the mailing address for a customer, the business address of the store, the mailing address of an employee.
address	CHARACTER VARYING(50)	The primary street address of the customer or store (e.g., building number and street name); up to 50 characters.
address2	CHARACTER VARYING(50)	Additional address information (ex: apartment number, suite number); up to 50 characters.
district	CHARACTER VARYING(20)	District of address information; up to 20 characters.
city_id	SMALLINT	**Foreign key**, uniquely identifying the city associated with the address, linked to the city in the city table.
postal_code	CHARACTER VARYING(10)	Postal code associated with the address; up to 10 characters.
phone	CHARACTER VARYING(20)	Phone number associated with the address; up to 20 characters.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to: city.
Links from the following tables: staff, store, customer.

staff

Columns	Data Type	Description
staff_ID	SERIAL	Surrogate **primary key,** uniquely identifying an individual store employee
first_name	CHARACTER VARYING(45)	First name of staff member; up to 45 characters
last_name	CHARACTER VARYING(45)	Last name of staff member; up to 45 characters
address_id	SMALLINT	**Foreign key** linking the staff member to their mailing address, stored in the address table.
email	CHARACTER VARYING(50)	Email address of the staff member; up to 50 characters
store_id	SMALLINT	**Foreign key**, uniquely identifying a physical store location, linked to the store table.
active	BOOLEAN	Indicates whether the employee is currently employed by the company (T/F).
username	CHARACTER VARYING(16)	The employee's username for accessing the company's internal platform
password	CHARACTER VARYING(40)	The employee's password for accessing the company's internal platform, limited to 16 characters.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.
picture	BYTEA	Employee's photo stored in binary data form.

Links to: address.
Links from the following tables: store, payment, rental.

store

Columns	Data Type	Description
store_id	SERIAL	Surrogate **primary key**, uniquely identifying a physical store.
manager_staff_id	SMALLINT	Surrogate key, uniquely identifying the manager of the store.
address_id	SMALLINT	**Foreign key**, referring to the address associated with the store.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to the following tables: staff, address.

city

Columns	Data Type	Description
city_id	SERIAL	Surrogate **primary key** uniquely identifying each city, referenced by the address table to associate an address with its city.
city	CHARACTER VARYING(50)	Name of city
country_id	SMALLINT	**Foreign key**, associating the city to the country where it is located.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to: country.

Links from: address.

country

Columns	Data Type	Description
country_id	SERIAL	Surrogate **primary key** uniquely identifying each

		country, referenced by the city table to associate a city with its country.
country	CHARACTER VARYING(50)	Name of country
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links from: city.

inventory

Columns	**Data Type**	**Description**
inventory_id	SERIAL	Surrogate **primary key**, uniquely identifying each physical copy of a movie in inventory.
film_id	SMALLINT	**Foreign key** uniquely identifying a movie title, referenced from the film table.
store_id	SMALLINT	**Foreign key**, uniquely identifying a physical store.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to: film.

Links from: rental.

film

Columns	**Data Type**	**Description**
film_id	SERIAL	Surrogate **primary key**, uniquely identifying a movie title
title	CHARACTER VARYING(255)	Title of movie (up to 255 characters)
description	TEXT	Short summary/overview of film.
release_year	YEAR	Film release year (4 digit format)

language_id	SMALLINT	**Foreign key**, uniquely identifying a language; referenced to associate a film with its language.
rental_duration	SMALLINT	The number of days a film title can be rented, typically used as the standard rental period for a particular title.
rental_rate	NUMERIC(4,2)	The price of renting the film title (stored with up to four digits before the decimal point and two digits after)
length	SMALLINT	The duration of the film in minutes.
replacement_ cost	NUMERIC(5,2)	Price of replacing a physical copy, if original in inventory is lost or damaged.
rating	mpaa_rating	The MPAA (Motion Picture Association of America) rating assigned to the film, indicating content and age-appropriate audience (e.g., G, PG, PG-13, R, NC-17).
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.
special_ features	TEXT[]	A list of extra content beyond the official film, such as behind-the-scenes footage, director's commentary, deleted scenes, etc.
fulltext	TSVECTOR	Terms (words) extracted from a document or text (such as a film's description), along with their positions (numbers after the colon) in the original text; TSVECTOR creates efficient indexes for searching terms, making it easier to find matches in large amounts of text. *(from ChatGPT)*

Links to: language.

Links from the following tables: inventory, film_actor, film_category.

language

Columns	Data Type	Description
language_id	SERIAL	Surrogate **primary key**, uniquely identifying a language
name	CHARACTER VARYING(20)	Name of language
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links from: film.

film_actor

*join table; each record represents a unique pairing of an actor and a film

Columns	Data Type	Description
actor_id	SMALLINT	**Foreign key**, referencing an actor from the actor table.
film_id	SMALLINT	**Foreign key**, referencing a film from the film table.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to the following tables: film, actor.

actor

Columns	Data Type	Description
actor_id	SERIAL	Surrogate **primary key**; uniquely identifying an individual actor
first_name	CHARACTER VARYING(45)	Actor’s first name
last_name	CHARACTER VARYING(45)	Actor’s last name

last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links from: film_actor.

film_category

*join table; each record represents a unique pairing of film and genre

Columns	**Data Type**	**Description**
film_id	SMALLINT	**Foreign key**; referencing unique identifier for a specific film title.
category_id	SMALLINT	**Foreign key**; referencing unique identifier for a film genre.
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links to the following tables: film, category.

category

Columns	**Data Type**	**Description**
category_id	SERIAL	Surrogate **primary key**; uniquely identifying a film genre
name	CHARACTER VARYING(25)	Name of category/genre
last_update	TIMESTAMP(6) WITHOUT TIME ZONE	Timestamp of the last update to the record, stored with up to six digits of fractional seconds precision.

Links from: film_category.

APPENDIX: Rockbuster Dataset - Description, Relationships, Constraints

This document provides an analysis of the Rockbuster dataset, detailing its structure, relationships, constraints, and any data integrity issues encountered during analysis.

Overview: The Rockbuster dataset contains records of 599 individual customers. Grouping all payment_ids by distinct customer_ids (from both the payment table and the revenue_analysis view) gives us the number of payments each customer has made, yielding 599 rows. There are also 599 unique customer_ids in the customer dimension table.

Each movie rental costs the customer one of three standard rental rates: $0.99, $2.99, or $4.99. Late fees are charged at $1 per day overdue.

Payment Table Analysis: By analyzing the payment table, we can count the occurrences of each amount value, producing the following results:

Amount	Count
11.99	8
10.99	99
9.99	233
9.98	1
8.99	438
8.97	1
7.99	621
7.98	5
6.99	1017
5.99	1188
5.98	7
4.99	3424
3.99	988
3.98	8
2.99	3233
1.99	580
1.98	1
0.99	2720
0.00	24

- We determined the late fee to be $1/day by comparing the amount a customer was charged for renting a movie against the movie's rental rate, rental duration, and the number of days between the rental and return dates. This calculation helped identify entries with amounts that did not round to the nearest $x.99 as likely data entry errors.
- Additionally, we found that the 24 instances of $0.00 charges corresponded to records with NULL return dates.

Recommendation – create a `revenue_analysis` view:

- All non-$0.00 amounts are rounded up to the nearest $x.99.
- $0.00 amounts are preserved but flagged.
- Three instances of underpayment are flagged.

Data Integrity and Issues: The original payment table of the Rockbuster dataset has 14,596 records. Each rental_id has one corresponding payment_id (14,596 records). However, there is an outlier with 5 payment_ids for rental_id ‘4591’, resulting in 14,592 records (-4).

The original rental table contains 16,044 records, with each rental_id being unique. When the payment table is LEFT JOINed to the rental table, there are 16,048 records, with 1,452 records containing NULL payment entries.

Upon examination, it was found that all NULL payment records fall between movie rentals from May 25, 2005, to June 30, 2005. These represent the first month of the rental dataset's time period (2005-05-24 to 2006-06-02), and 42.27% of records from that month are missing payment information.

For revenue analysis, we set the condition `rental_date >= 2005-07-01 00:00`. However, for popularity/frequency analysis, the NULL records are preserved.

Data Inconsistencies and Fixes:

- When performing a LEFT JOIN, each payment_id for rental_id 4591 gets joined to the corresponding record in the rental table, creating duplicate rows. Since there are 5 payment_ids for a single rental_id, this results in 4 additional, duplicate rows.

	payment_id integer	customer_id smallint	rental_id integer	customer_id smallint
1	19518	16	4591	182
2	25162	259	4591	182
3	29163	401	4591	182
4	31069	182	4591	182
5	31834	546	4591	182

- Upon further inspection of rental_id 4591, we find discrepancies where the customer_id from the payment table (column 2) and the customer_id from the rental table (column 4) should be the same. We determined that rental_id 4591 should only be attributed to the record in row 4: payment_id 31069, customer_id 182. For the remaining records, priority should be given to the `payment_customer_id`, i.e., the value found in column 2.

Timestamp Discrepancies: The issue with rental_id 4591 stems from a system malfunction that caused inaccurate rental_date and return_date timestamps (last two columns in the payment table). This error results in identical timestamps across all five records, which is suspicious since the rental duration varied between clients.

The payment dates cover the period from February 14, 2007, to May 14, 2007 – a period of three months. Rental records, however, begin with the earliest rental date of May 24, 2005, and the last return date of June 2, 2006 – a period of about a year.

Location Information Updates

The following updates were made to the dataset for accuracy and standardization of country names:

1. **Réunion**: Updated from "runion" to "Réunion" for correct spelling.
2. **Yugoslavia**: Updated from "Yugoslavia" to "Serbia" to reflect the current political entity after the breakup of Yugoslavia.
3. **Kazakhstan**: Updated from "Kazakstan" to "Kazakhstan" for correct spelling.

These changes ensure consistency with the most up-to-date geopolitical standards.