

Prototyping

By: Andrew Snyder, Alex Polston, Alek Norris,
Eamon Collins, James Byrd, Svyatoslav Varnitskyy

sdmay25-20

HADES: Holistic Astronomical
Database Exploration System

Project Overview

- **Objective:** Develop a system to manage and analyze simulated binary star data
- **Key Features:**
 - Import multivariate time-series simulation data into relational database
 - Provide sample SQL queries
 - Enable custom queries through natural language processing
- **Deliverables:**
 - Relational database
 - User Interface for writing and viewing SQL queries
 - Sample SQL queries



UI Prototyping

🔍 Query

↶ History

🔧 Utilities

⚙️ Settings

ℹ️ About Us

HADES

How can I help?

Welcome to the Binary Star Query Bot! This interactive assistant is designed to help you explore and retrieve data on binary star systems with ease. Whether you're an astronomer, student, or space enthusiast, you can quickly access detailed information on various binary systems, including orbital periods, mass, luminosity, and more. Additionally, you can upload your own binary star data, and the bot will parse and integrate it for seamless querying. From answering general questions about binary stars to providing insights into specific systems, this bot is here to enhance your understanding of the fascinating world of stellar pairs. Let's explore the stars together!

Query



UI Reflection

Why We Need This:

- Users will utilize this page as a home/query page

Goals in Creating a Prototype:

- Demonstrate overall styling and design pattern
- Determine if UI is intuitive

Outcomes:

- Styling looks professional and intuitive
- We need to expand on the navigation bar to include more tools
- Overall experience is positive

NLP SQL ProtoTyping

Still awaiting funding from Iowa State To use API so “free” version of GPT with prompt training was used to demonstrate functionality.

Please help me make a query that can do the following. Find the initial mass, age of star S1, and values X and Y for all records where the initial mass of star S1 is between X and Y.

Here is the SQL query for your requirement

sql

Copy code

```
SELECT "S1_star_mass", "S1_star_age", "X", "Y" FROM "grid_0_0142_grid_initial_values" WHERE
```

```
hades=# SELECT "S1_star_mass", "S1_star_age", "X", "Y" FROM "grid_0_0142_grid_initial_values" WHERE "S1_star_mass" BETWEEN "X" AND "Y";
```

S1_star_mass	S1_star_age	X	Y
0.6485974130048953	204761.21349503414	0	0.9858
0.9583208826042157	70351.5454282795	0	0.9858
0.4999776686421848	440976.44075786445	0	0.9858
0.5694623297313895	272367.8957760109	0	0.9858
0.8413909711359655	103439.14060812922	0	0.9858
0.738730130709694	148952.36247570606	0	0.9858

(6 rows)

hades=# |

Prompt: Please help me make a query that can do the following. Find the initial mass, age of star S1, and values X and Y for all records where the initial mass of star S1 is between X and Y.

Query: `SELECT "S1_star_mass", "S1_star_age", "X", "Y"
FROM "grid_O_0142_grid_initial_values"
WHERE "S1_star_mass" BETWEEN "X" AND "Y";`

Generated query correctly and returns relevant data.

NLP SQL ProtoTyping

Prompting

Im wanting to query
"public.grid_0_0142_grid_run0_final_profile1" to
look at the energy, where the mass is greater
than 5.

To query the data for research purposes based on your requirement,
you can use the following SQL command:

```
sql
SELECT energy
FROM "grid_0_0142_grid_run0_final_profile1"
WHERE mass > 5;
```

nothing showed up, could you show me one,
including "logRho" where mass is greater than
1.5?

You can use the following SQL query to look at `logRho`, `mass`, and
`energy` values where `mass` is greater than 1.5 from the
"public.grid_0_0142_grid_run0_final_profile1" table:

```
sql
SELECT mass, logRho, energy
FROM "grid_0_0142_grid_run0_final_profile1"
WHERE mass > 1.5;
```

Queries and Results

```
hades=# SELECT mass, energy FROM "grid_0_0142_grid_run0_final_profile1" where mass > 5;
 mass | energy
-----+-----
(0 rows)
```

```
hades=# SELECT mass, "logRho", energy FROM "grid_0_0142_grid_run0_final_profile1" where mass > 1.5;
 mass | logRho | energy
```

```
-----+-----+-----
2.428273796159032 | -9.136390283830577 | 1918837430976.8416
2.4282737961569523 | -9.136385738294923 | 1918846682484.2263
2.4282737961536687 | -9.136376285548819 | 1918865921319.566
2.428273796145799 | -9.136353507185277 | 1918912279573.6323
2.428273796126791 | -9.136309798045636 | 1919001227357.688
2.428273796094218 | -9.136225588910927 | 1919172564990.2378
2.4282737960273955 | -9.13605283777916 | 1919523938069.9016
2.4282737958902247 | -9.135671753942669 | 1920298508511.1755
2.428273795576972 | -9.134977642395501 | 1921706794345.8872
2.4282737950683777 | -9.133810604257508 | 1924067006502.471
2.4282737941908974 | -9.131446150132005 | 1928817056717.969
2.428273792243294 | -9.1271758655197 | 1937239745625.9265
2.428273789032749 | -9.121944620825284 | 1947186937349.4182
2.428273785828184 | -9.115405319729237 | 1959027315942.9255
2.4282737808556893 | -9.105310190363538 | 1976041574932.489
2.428273772867065 | -9.092210443503726 | 1997346728660.3992
2.4282737634462848 | -9.078021173265473 | 2021613784892.8313
2.428273753204602 | -9.065011749167072 | 2045749040781.1973
2.42827374459265 | -9.053217905603713 | 2069786570419.9033
2.428273735339396 | -9.040464185947844 | 2098166821978.6426
```

Native Data

grid > run0 > final_profile1									
Compound									
Cell width: 120									
Notation Auto Scientific Fixed-point Freeze indices Export									
grid.0.0142.h5									
~ grid									
@ final_values									
@ initial_values									
~ run0									
@ final_profile1									
@ history1									
> run1									
> run10									
> run11									
> run12									
> run13									
> run14									
> run15									
> run16									
> run17									
> run18									
> run19									
> run2									
> run20									
> run21									
> run22									
> run23									
> run24									
> run25									
radius mass logRho omega energy x.mass fracty.mass fracty									
z 1923									
x [D0]									
0	36.602	2.428	-9.136	0.000	18837430976.8416	0.000	0.986		
1	36.602	2.428	-9.136	0.000	18846682484.2263	0.000	0.986		
2	36.602	2.428	-9.136	0.000	18865921319.566	0.000	0.986		
3	36.602	2.428	-9.136	0.000	18912279573.6323	0.000	0.986		
4	36.602	2.428	-9.136	0.000	19001227357.688	0.000	0.986		
5	36.602	2.428	-9.136	0.000	1912564990.2378	0.000	0.986		
6	36.602	2.428	-9.136	0.000	19523938069.9016	0.000	0.986		
7	36.602	2.428	-9.136	0.000	20298508511.1755	0.000	0.986		
8	36.602	2.428	-9.135	0.000	21706794345.8872	0.000	0.986		
9	36.601	2.428	-9.134	0.000	24867006502.471	0.000	0.986		
10	36.601	2.428	-9.131	0.000	28817056717.969	0.000	0.986		
11	36.600	2.428	-9.127	0.000	37239745625.9265	0.000	0.986		
12	36.599	2.428	-9.122	0.000	47186937349.4182	0.000	0.986		
13	36.597	2.428	-9.115	0.000	59027315942.9255	0.000	0.986		
14	36.595	2.428	-9.105	0.000	70841574932.489	0.000	0.986		
15	36.591	2.428	-9.092	0.000	7346728660.3992	0.000	0.986		
16	36.587	2.428	-9.078	0.000	21613784892.8313	0.000	0.986		
17	36.583	2.428	-9.065	0.000	15749040781.1973	0.000	0.986		
18	36.579	2.428	-9.053	0.000	19786570419.9033	0.000	0.986		
19	36.576	2.428	-9.040	0.000	28166821978.6426	0.000	0.986		
20	36.571	2.428	-9.028	0.000	2749374485.1	0.000	0.986		
21	36.568	2.428	-9.014	0.000	56862064225.4	0.000	0.986		
22	36.562	2.428	-8.994	0.000	21189212434.1	0.000	0.986		

1. Initial generated Query that returned no matching data
2. Updated Query and results

NLP SQL Reflection

Why We Need This:

- Not all users know how to write proper SQL queries

Goals in Creating a Prototype:

- To demonstrate how Machine Learning can be used to translate natural language to SQL queries

Outcomes:

- Natural language to SQL statement translation is possible
- Prompt training helped improve translation accuracy
- Further, stricter prompt training is required

Implications and Next Steps

- We have confidence that the NLP SQL generation is feasible and workable
- We have confidence that we can meld the AI SQL generation with the PostgreSQL database
- Expand on UI to include other pages and begin integration with Natural Language Processing
- Begin incorporating Database connection into UI
- There is still further work to do on the prompt training of the AI SQL generation module, however after tinkering expectations are high
- The integration of the Data into the database seems to be working, further work needs to be done to facilitate the normalization of the data.