Information Modeling of Student Awards and Prizes

Amol Govekar
Hrishikesh Potdar
Jun Liu

Agenda for Today

- Overview of the Application Domain.
- Challenges and Controversies that we faced in various milestones.
- Best tools for Information Modeling based on our learning.
- Conclusion.

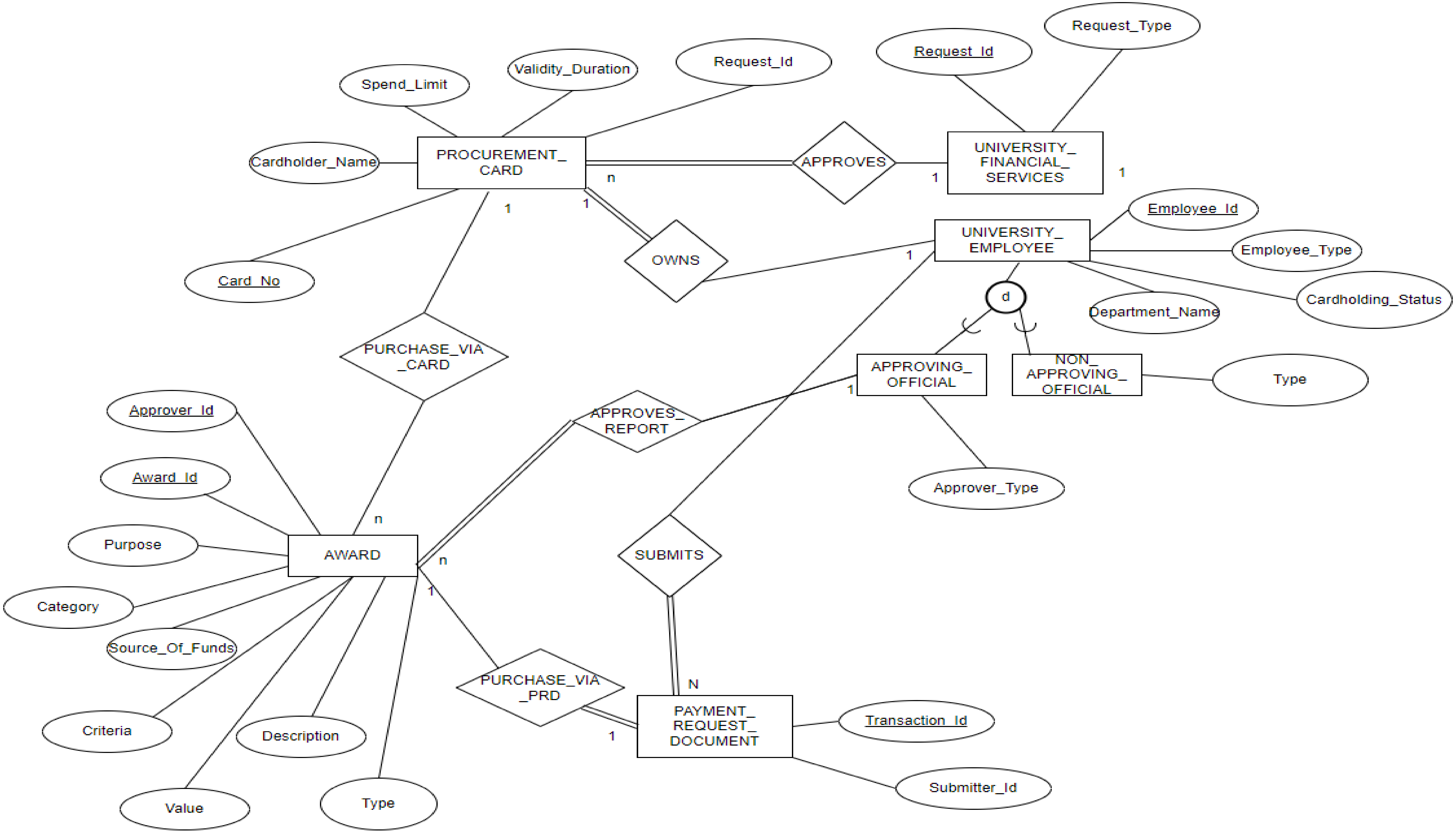
Overview of Application Domain

Establishment of Student Recognition Awards and Prizes System for the University of Kentucky

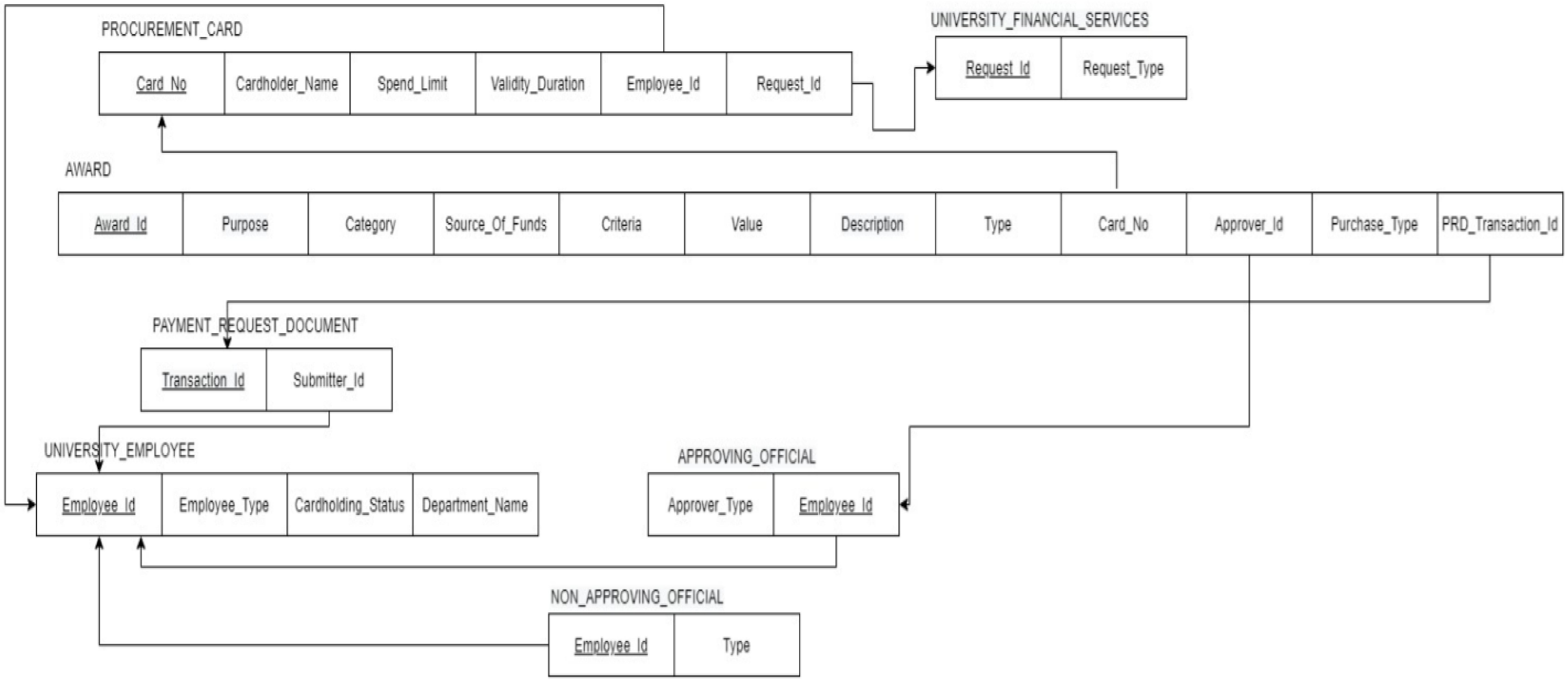
Goal- To develop a guidelines and solutions for a system for awards administration subjected to polices of the University of Kentucky.

Key Entities - University Employees, Procurement Card, Award, University Financial Services, Purchase Request Document.

Overview of the EER Diagram of the System



Our Relational Schema



PROCUREMENT_CARD

Card_No	Cardholder_Name	Spend_Limit	Validity_Duration	Employee_Id	Request_Id

UNIVERSITY_FINANCIAL_SERVICES

Request_Id	Request_Type

AWARD

Award_Id	Purpose	Category	Source_Of_Funds	Criteria	Value	Description	Type	Card_No	Approver_Id	Purchase_Type	PRD_Transaction_Id

PAYMENT_REQUEST_DOCUMENT

Transaction_Id	Submitter_Id

UNIVERSITY_EMPLOYEE

Employee_Id	Employee_Type	Cardholding_Status	Department_Name

APPROVING_OFFICIAL

Approver_Type	Employee_Id

NON_APPROVING_OFFICIAL

Employee_Id	Type

Difficulties and Challenges

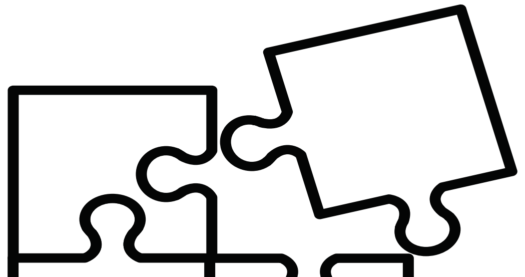
Milestone 1

Challenges:

1. Whether to implement nested specializations of University Employee into Approving and Non-Approving Official and further specialization of Approving Official and Non-Approving Official.
2. Deciding participation of two entities, the University Financial Services (UFS) and the Office of Student Financial Aid (OSFA).

Solutions:

1. Went through study material and decided on using multilevel specializations.
2. Carefully studies all the material and decided that both entities should have total participation.



Continued…

Milestone 2

Challenges:

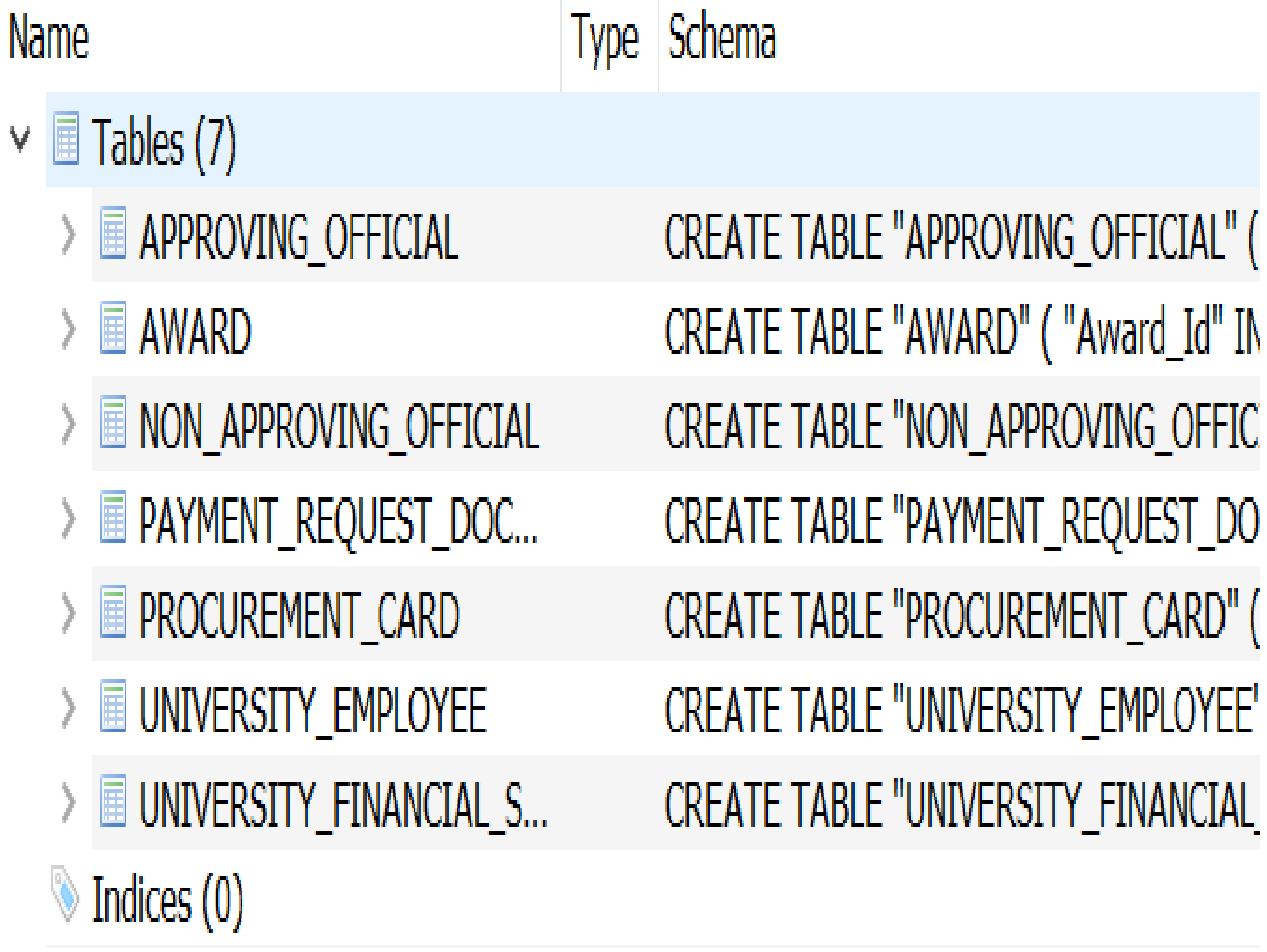
1. Whether to completely remove multilevel specializations in our EER diagram or add attributes to entities (multivalued attributes).

2. Whether to use “ON DELETE CASCADE” or “ON DELETE UPDATE” rules on foreign keys, as it was not clear which rule would be optimal with respect to our design.

Solutions:

1. We removed multi-level specialization and replaced it with adding multivalued attributes to required entities.

2. We decided to proceed with ON DELETE CASCADE as after deletion in the parent table there was no purpose for a tuple to exist in the child table.



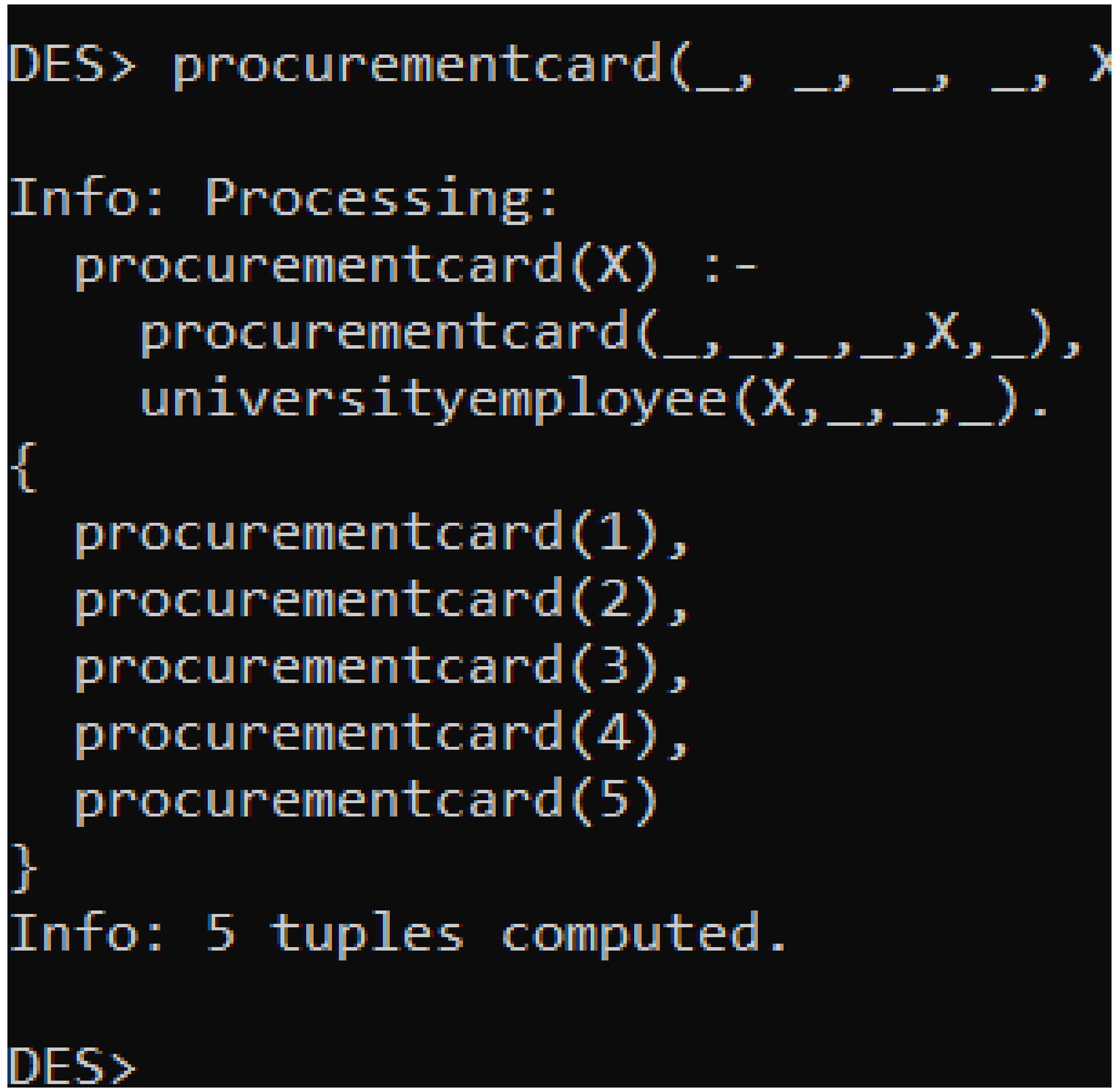
Name	Type	Schema
Tables (7)		
APPROVING_OFFICIAL		CREATE TABLE "APPROVING_OFFICIAL" (
AWARD		CREATE TABLE "AWARD" ("Award_Id" IN
NON_APPROVING_OFFICIAL		CREATE TABLE "NON_APPROVING_OFFIC
PAYMENT_REQUEST_DOC...		CREATE TABLE "PAYMENT_REQUEST_DO
PROCUREMENT_CARD		CREATE TABLE "PROCUREMENT_CARD" (
UNIVERSITY_EMPLOYEE		CREATE TABLE "UNIVERSITY_EMPLOYEE"
UNIVERSITY_FINANCIAL_S...		CREATE TABLE "UNIVERSITY_FINANCIAL_
Indices (0)		

SQLite

Milestone 3:

Challenges:

1. While designing new inference rules a controversy arose on whether to include any AND/OR clauses in queries.
2. Implementing nested queries from our previous milestone.



```
DES> procurementcard(_, _, _, _, X

Info: Processing:
  procurementcard(X) :-
    procurementcard(_,_,_,_,X,_),
    universityemployee(X,_,_,_).
{
  procurementcard(1),
  procurementcard(2),
  procurementcard(3),
  procurementcard(4),
  procurementcard(5)
}
Info: 5 tuples computed.

DES>
```

Solutions:

1. We came to the consensus that we can add little complexity to our new inference rules, thus we incorporated multiple clauses (AND/OR) in our rules.
2. We went ahead with simplifying the query from previous milestone and getting the output.

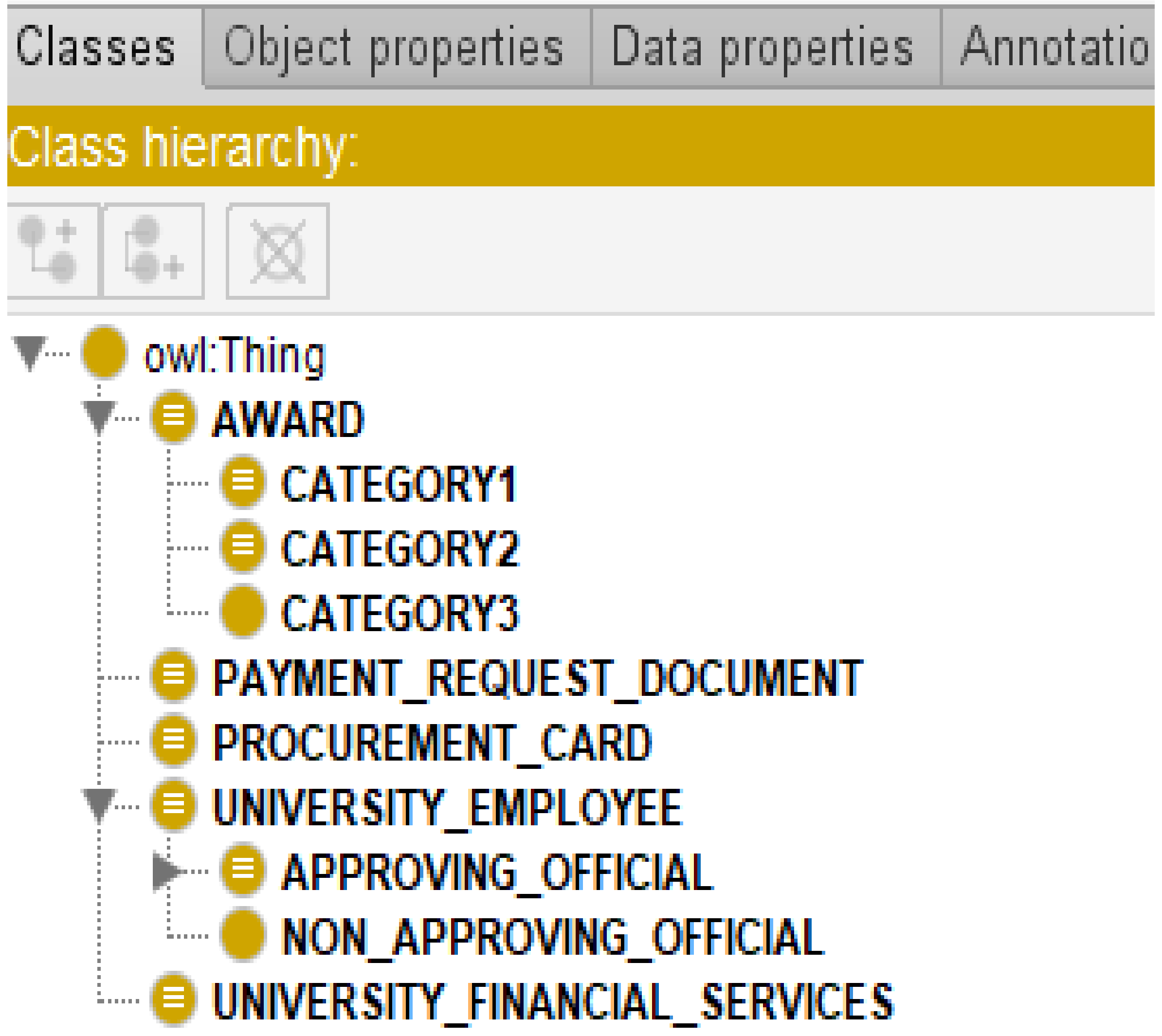
Milestone 4:

Challenges:

While designing data properties, we were not sure whether to add foreign keys present in table (“class” in context of ontology) as an independent data property.

Solutions:

We resolved this confusion by removing redundant data properties added for every class.



protégé

Best tools for Information Modeling

We utilized following information modelling tools for designing our application domain:

1. SQLite/DB Browser, MySQL.
2. SWI-Prolog, Datalog Educational System (DES).
3. Ontology, Web Ontology Language (OWL), and Protégé

We came to conclusion that **SQLite** will be the best tool for information modeling with respect to our domain.

1. It was easy to incorporate our conceptual modelling ideas into SQLite, designing database, inserting data and formulating queries to retrieve useful information.
2. It was comparatively easier to execute complex queries in SQLite as compared to DES. We found that it's easier to debug errors in SQLite as compared to DES.
3. SQLite has more widespread community support as compared to modelling tools like Protégé, thus it was easier to resolve issues in SQLite.

Conclusion

1. SQLite will be the best tool for information modeling with respect to our domain.

2. Information system can improve the management and administrative efficiency of higher education institutions.

3. A variety of effective information modeling tools can be used in combination to achieve the best results.

4. Team collaboration is an efficient way to work in information modeling, and external technical support/Q&A is an effective means to solve programming bugs.