System Analysis – Part 2

Information Systems Modelling and Design
CN4000/CD4000

Arish Siddiqui

Data Modelling Diagrams

Entity Relationship Diagram (ERD)

Information Flow Diagram (IFD)

Context Diagram (CD)

Dataflow Diagram (DFD)

Developing an ER Diagram

- Step 1 Identify Entities
- Step 2 Work out relationships
- Step 3 Identify attributes
- Step 4 Identify cardinality
- Step 5 Identify KEY attributes
- (Resolve assumptions)

Design Guidelines

- Best practices rather than rules
- Entities should have many occurrences
- Avoid unnecessary attributes
- Clearly label all components
- Apply correct cardinality
- Break attributes into lowest level needed
- Labels should reflect common business terms
- Assumptions should be clearly stated

Balancing ERDs with DFDs

All analysis activities are interrelated

Process models contain two data components

Data flows and data stores

The DFD data components need to balance the ERD's data stores (entities) and data elements (attributes)

Many CASE tools provide features to check for imbalance

Check that all data stores and elements correspond between models

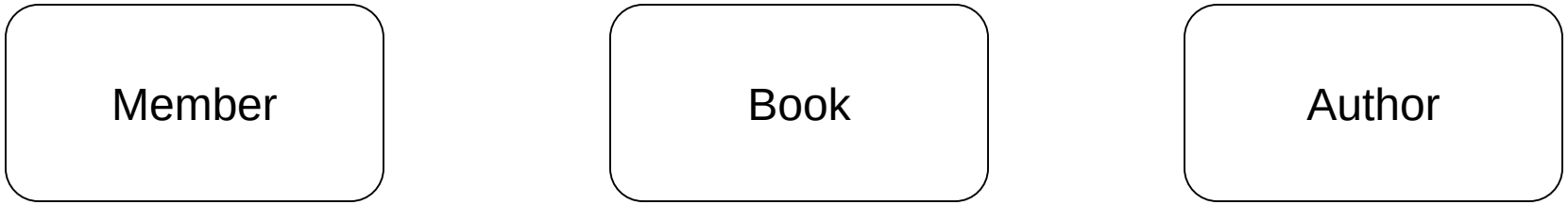
Example Problem

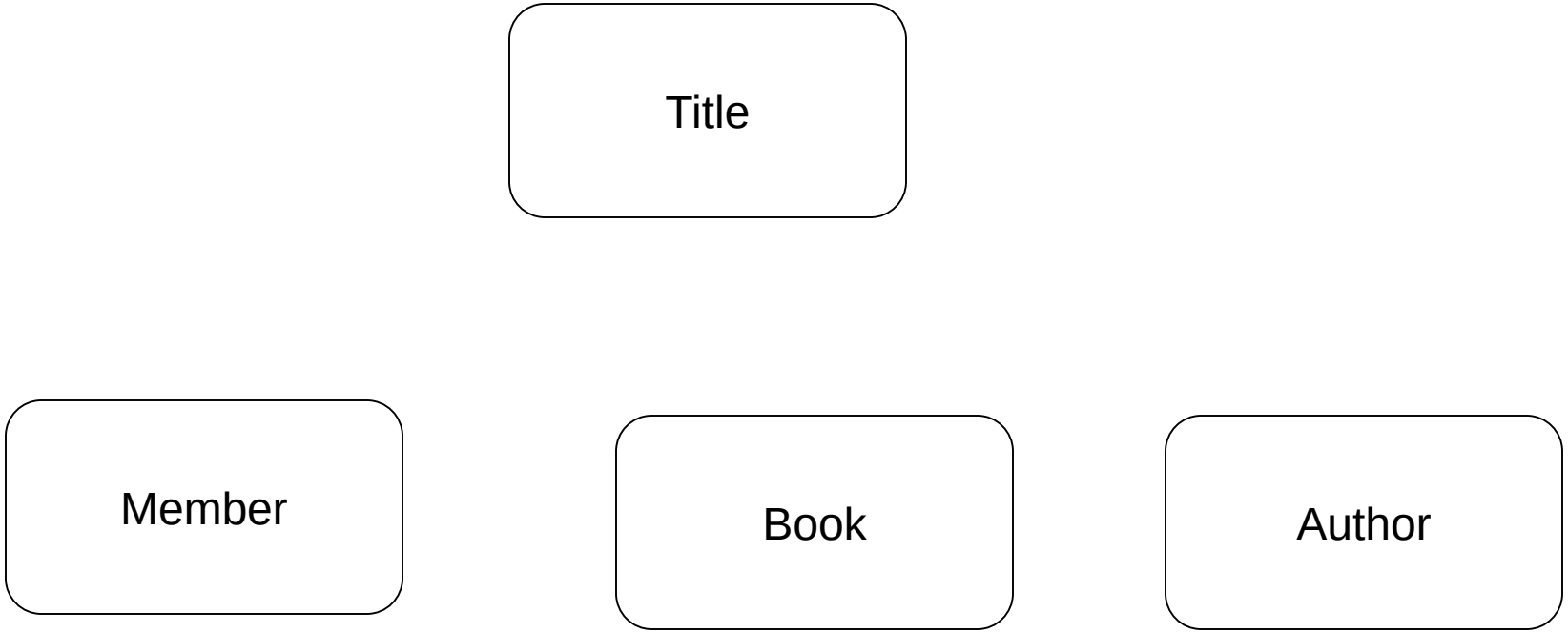
•A college library holds books for its members to borrow.

- Each book may be written by more than one author.
- Any one author may have written several books.

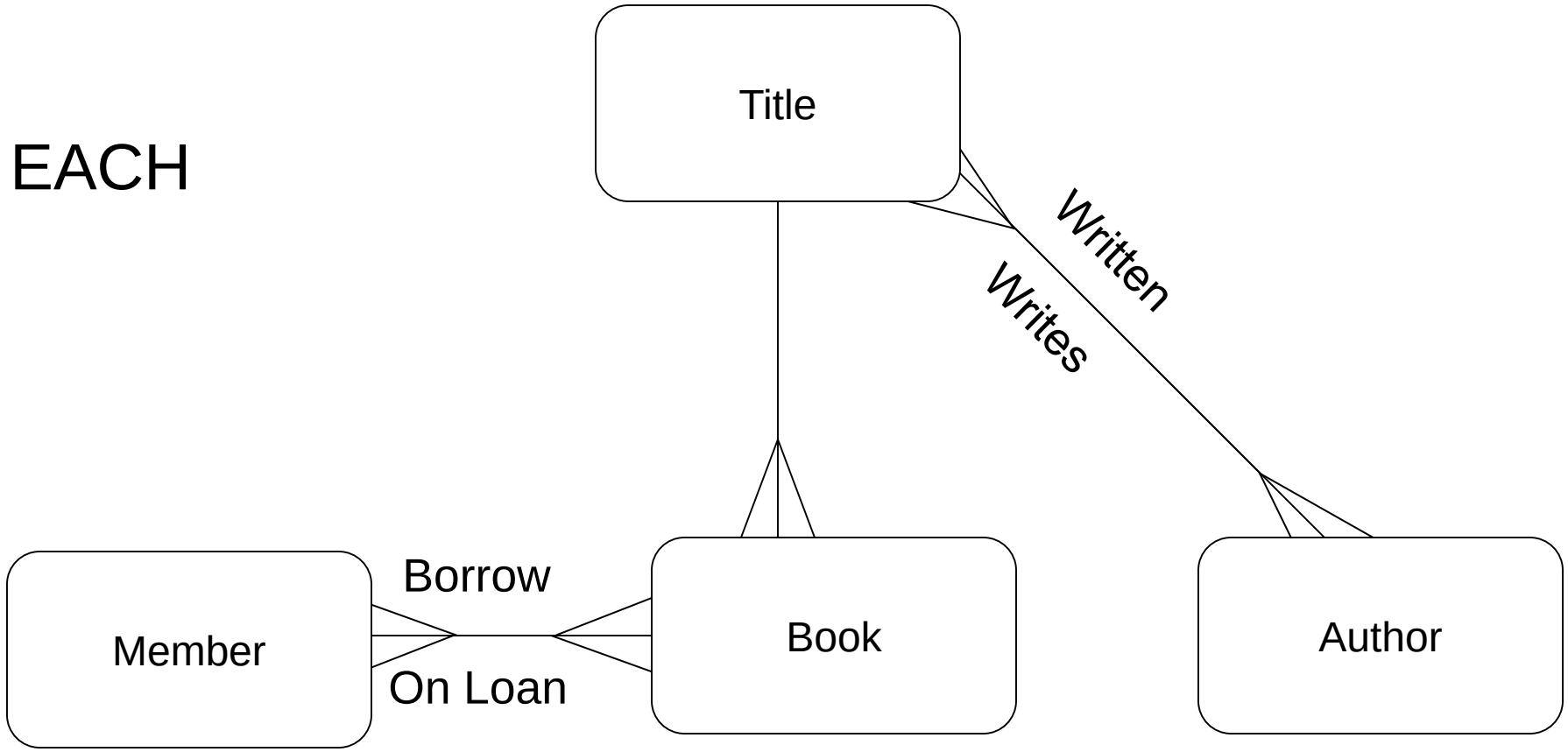
If no copies of a wanted book are currently in stock, a member may a reservation for the title until is available.

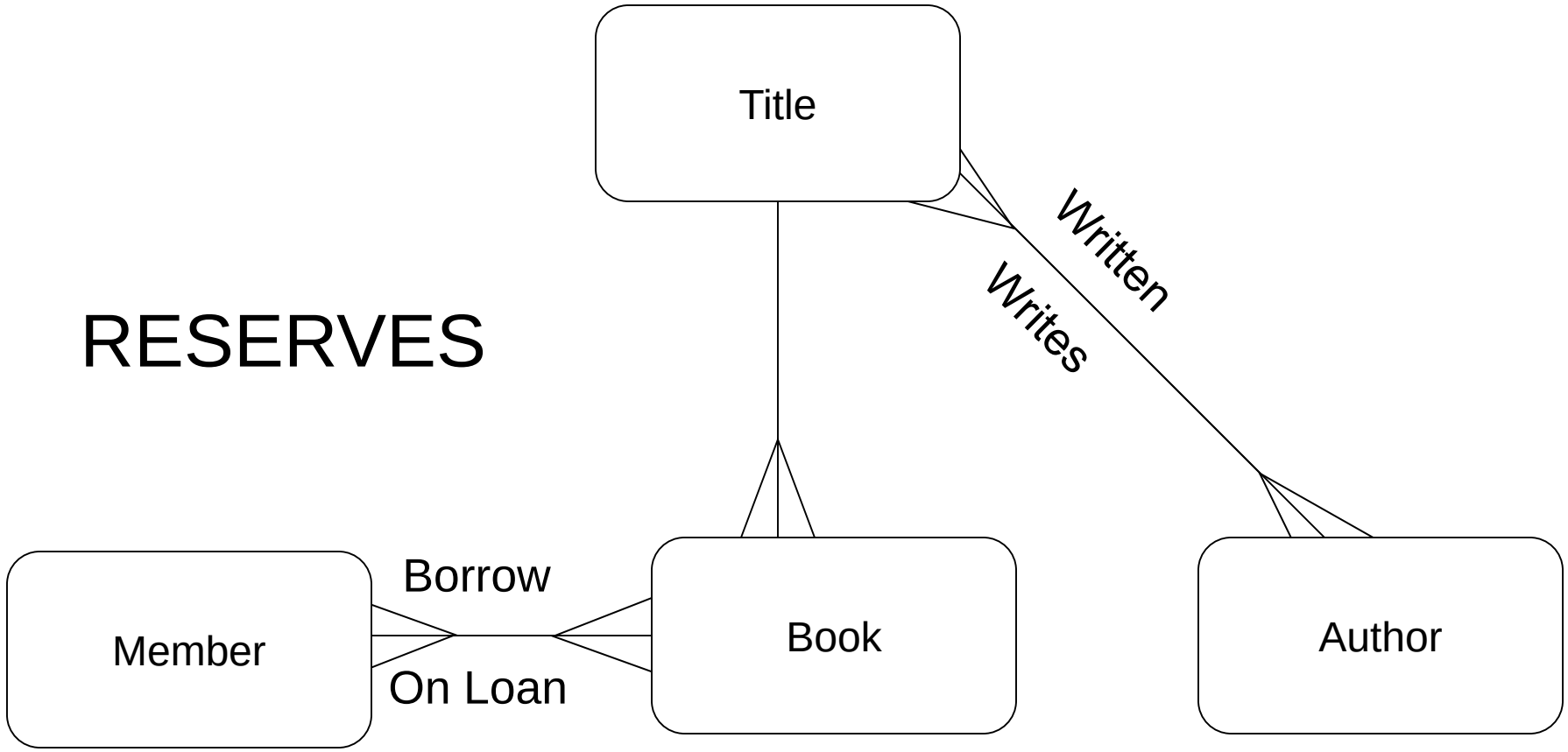
- If books are not returned on time a fine is imposed and if the fine is not paid the member is barred from loaning any other books until the fine is paid.


Member
Book
Author


Title
Member
Book
Author

EACH

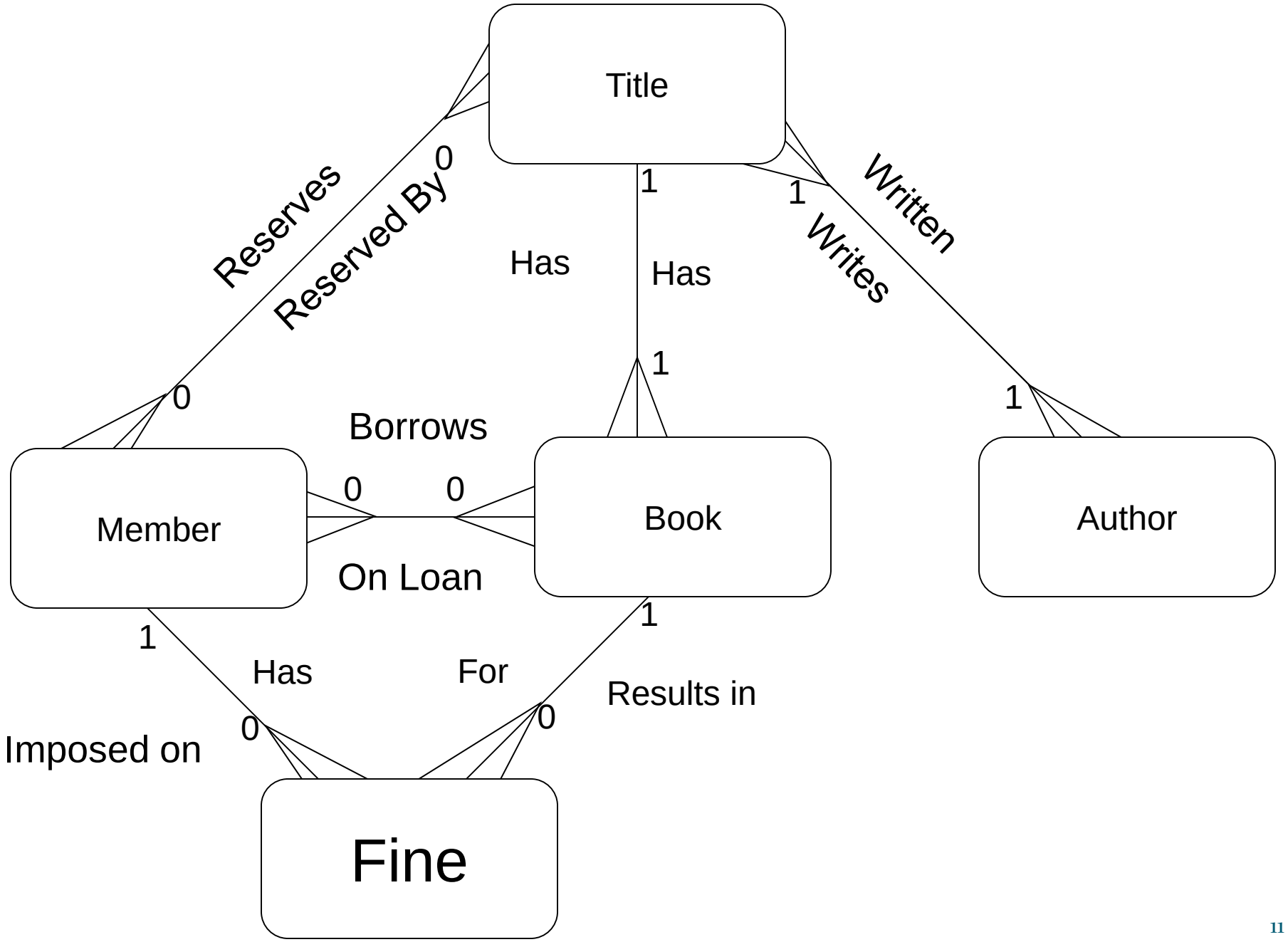


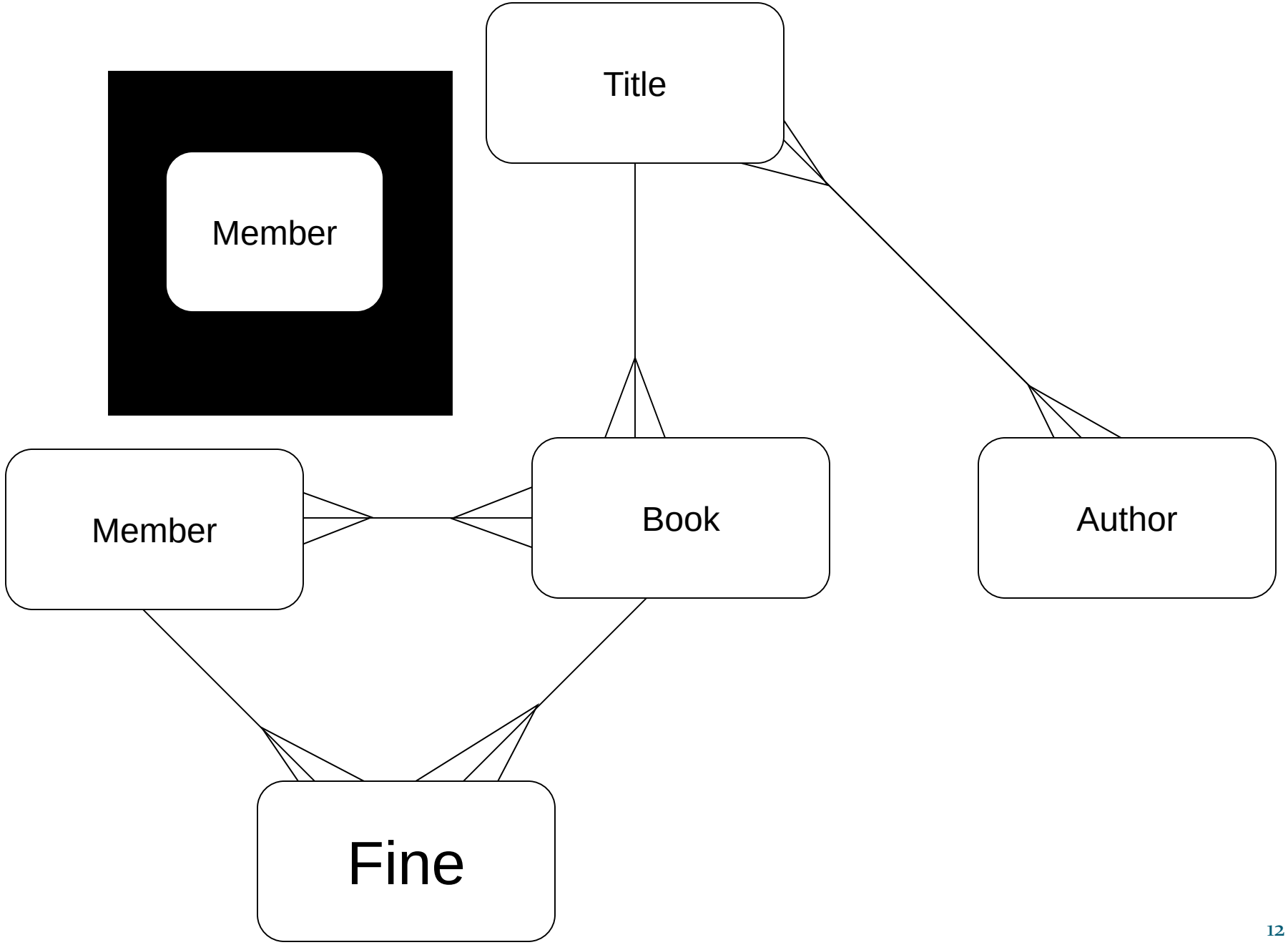


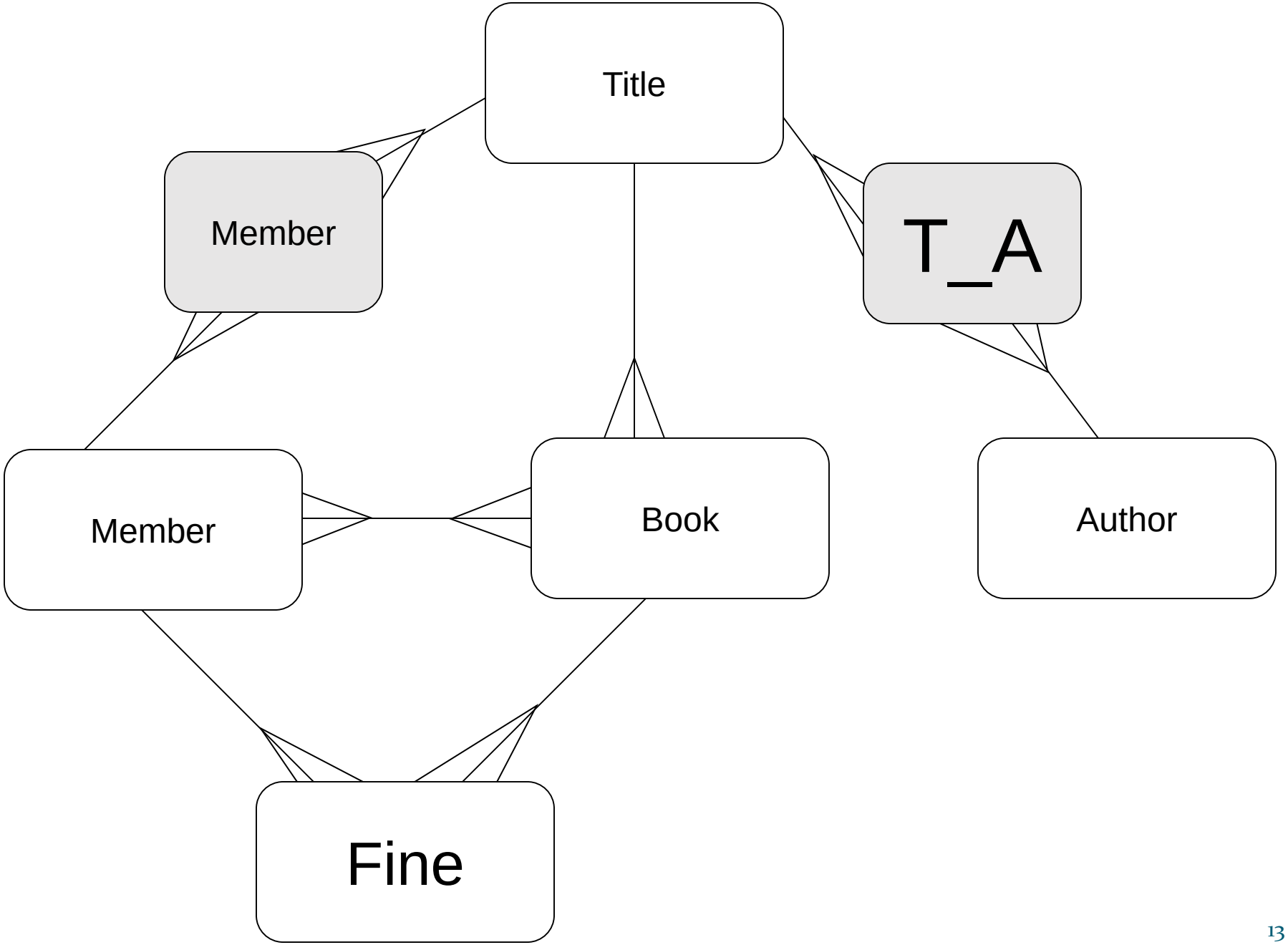
RESERVES

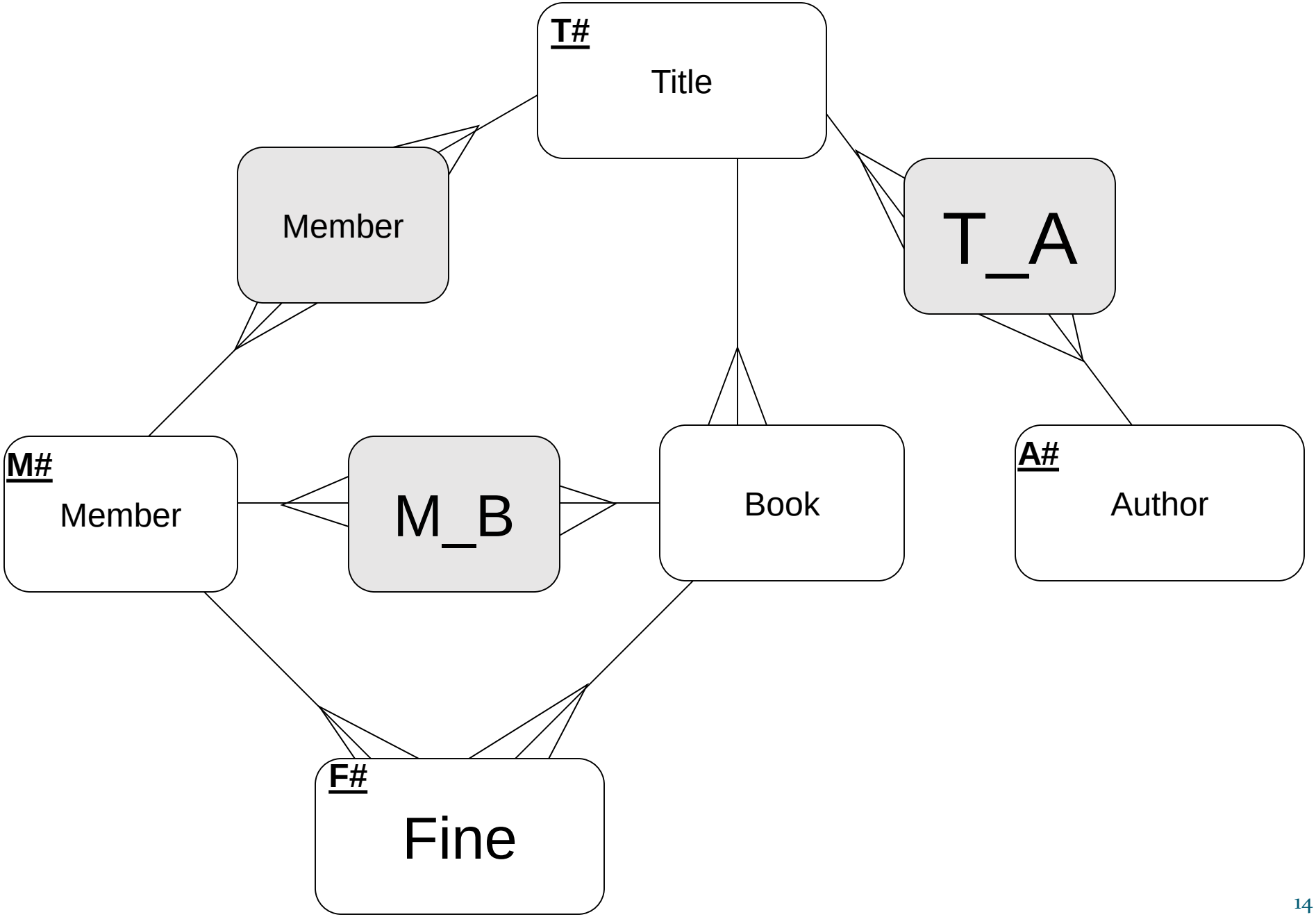
BARRED

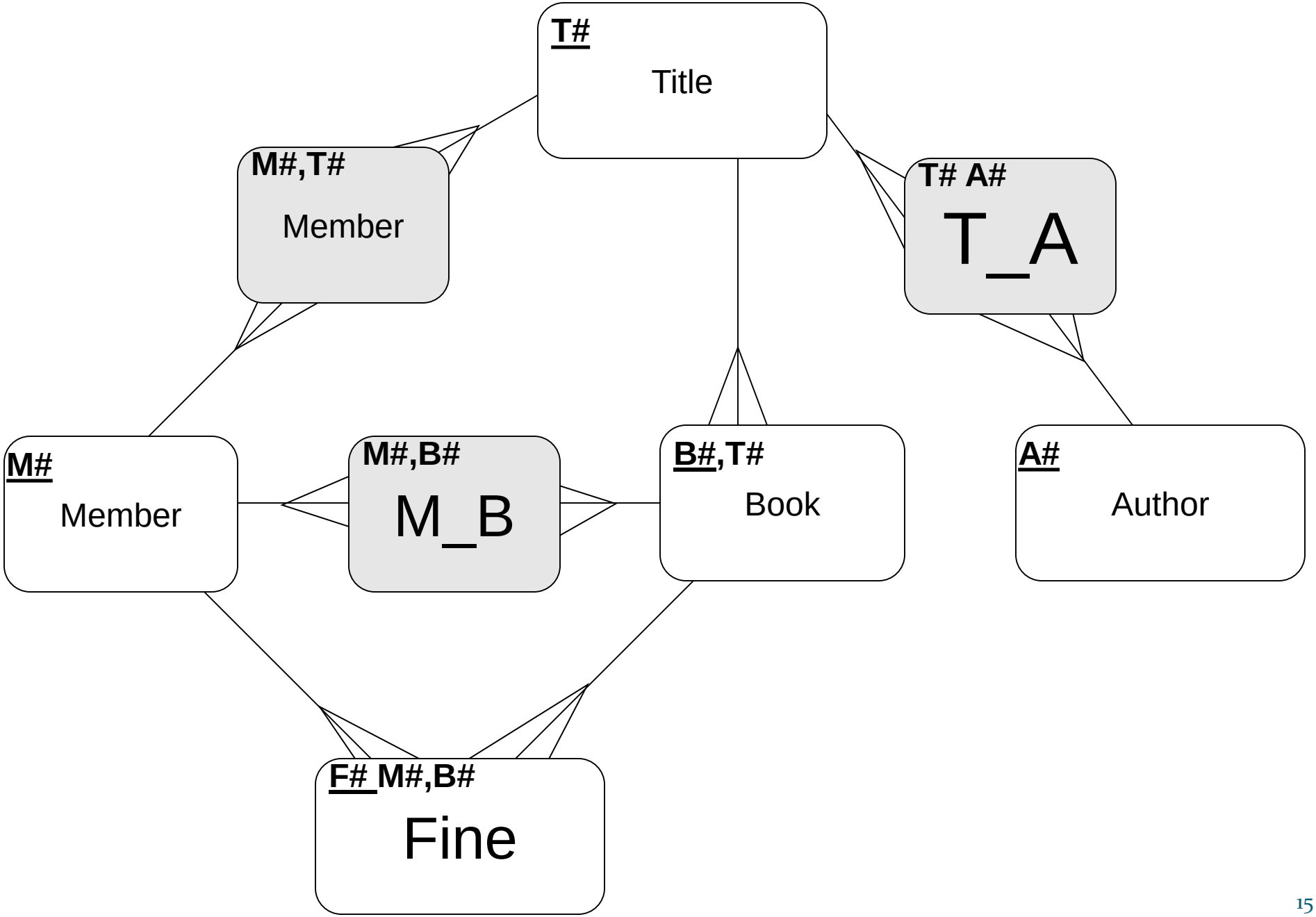
FINE


Title
Reserves
Reserved By
0
0
1
1
Written
Writes
Has
Has
1
1
0
Borrows
Member
0
0
Book
Author
On Loan
1
1
Has
For
Results in
0
0
Imposed on
Fine


Member
Title
Member
Book
Author
Fine


Title
Member
T_A
Member
Book
Author
Fine


T#
Title
Member
T_A
M#
Member
M_B
Book
A#
Author
F#
Fine


T#
Title
M#,T#
Member
T# A#
T_A
M#
Member
M#,B#
M_B
B#,T#
Book
A#
Author
F# M#,B#
Fine

Reserve#
Date reserved
Member #
Title #

Title#
Title
Category

T_A #
Title #
Author #

M_B #
Date issued
Date returned
Member #
Book #

Book / Serial #
Date Purchased
Condition
Title #

Author #
Name

Member #
Name
Address
Date-joined
Barred Y/N

Fine #
Date
Imposed
Member #
Book #

Information flow diagram (IFD)

- **Information flow diagram (IFD)**: A simple diagram showing how information is routed between different parts of an organisation. It has an information focus rather than a process focus.

- Purpose:
  - to document the main flows of information around the organisation;
  - for the analyst to check that they have understood those flows and that none has been omitted;
  - the analyst may use them during the fact-finding process itself as an accurate and efficient way to document findings as they are identified;

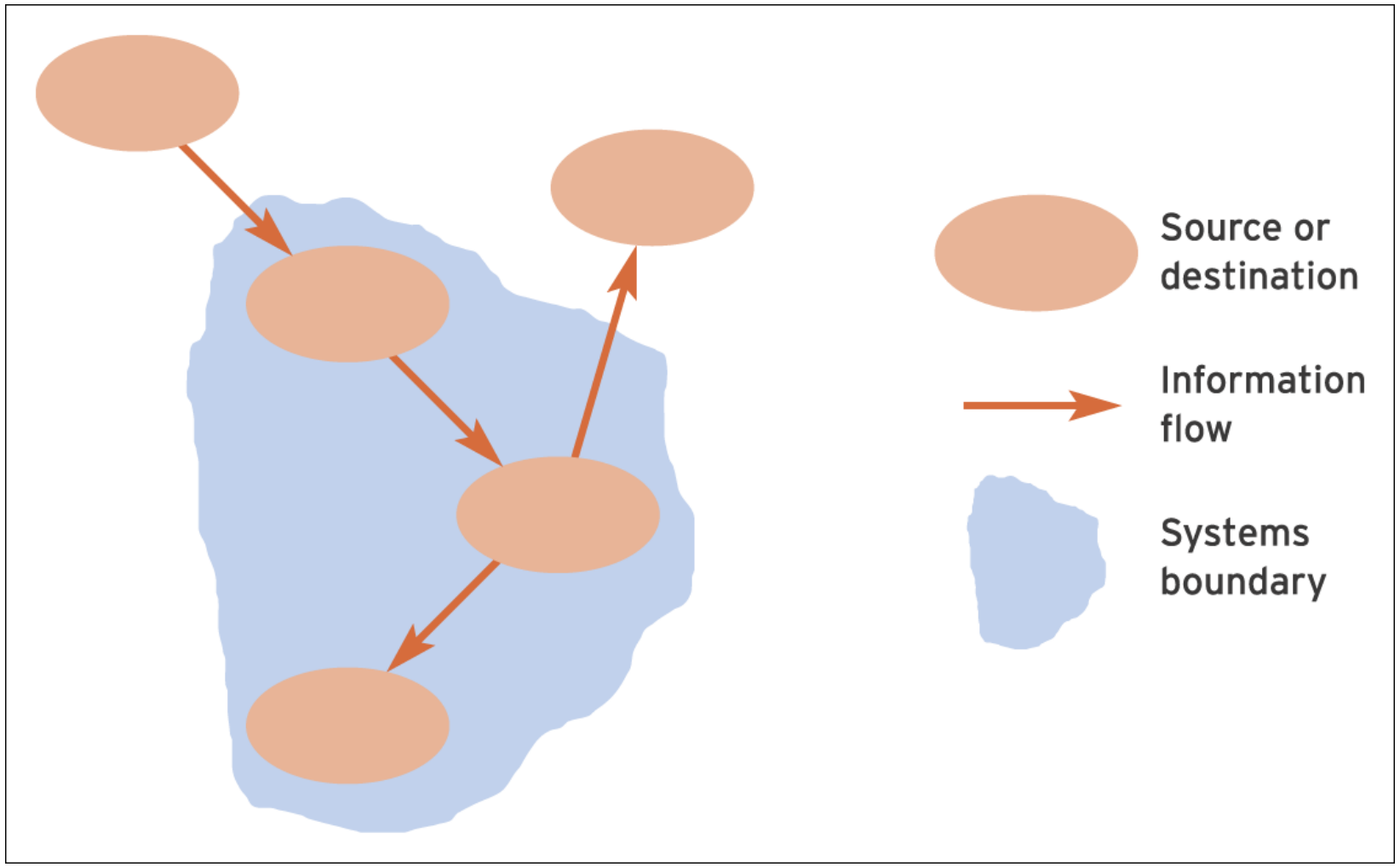


Figure 10.3 Information flow diagrams – the basic building blocks

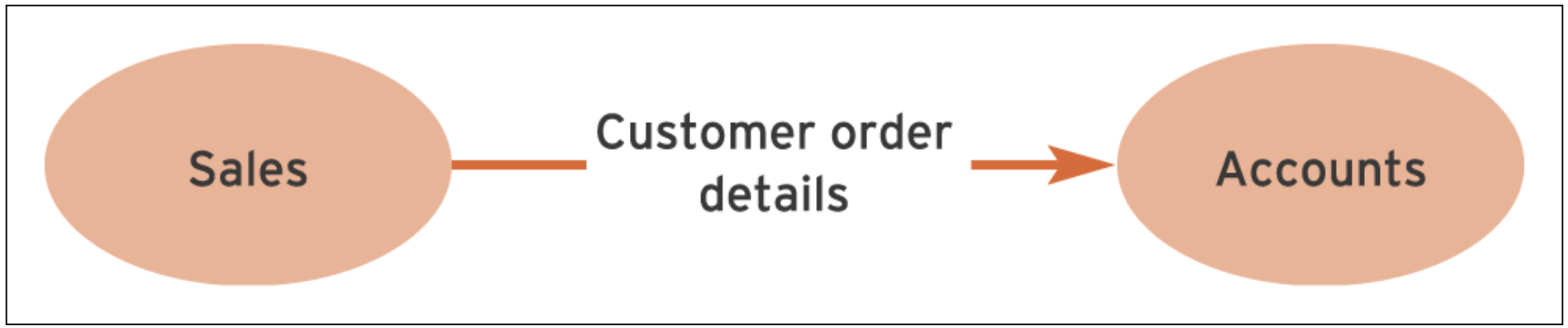


Figure 10.4 An illustration of a simple information flow

Figure 10.5 A simple, high-level IFD, excluding the system boundary

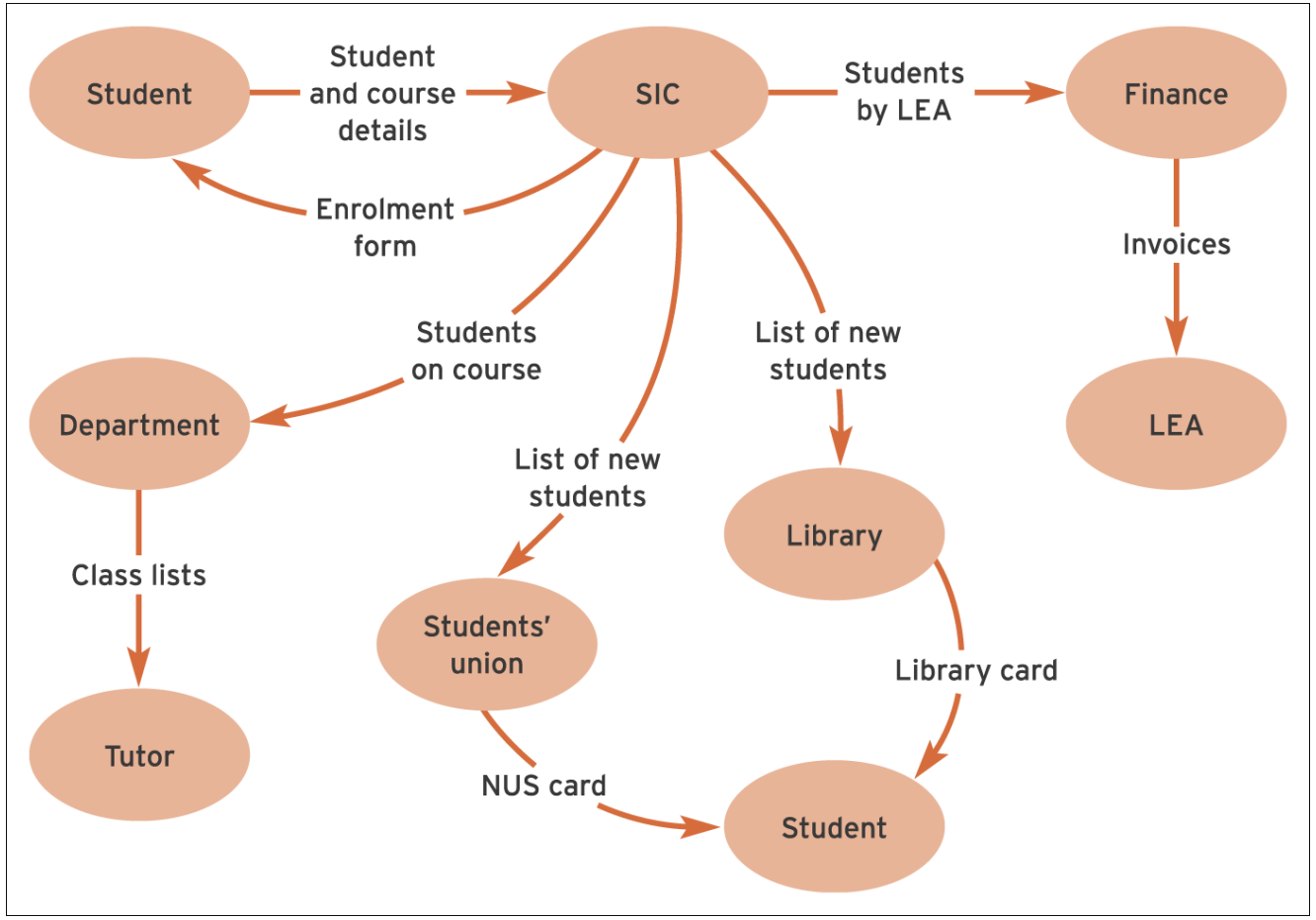
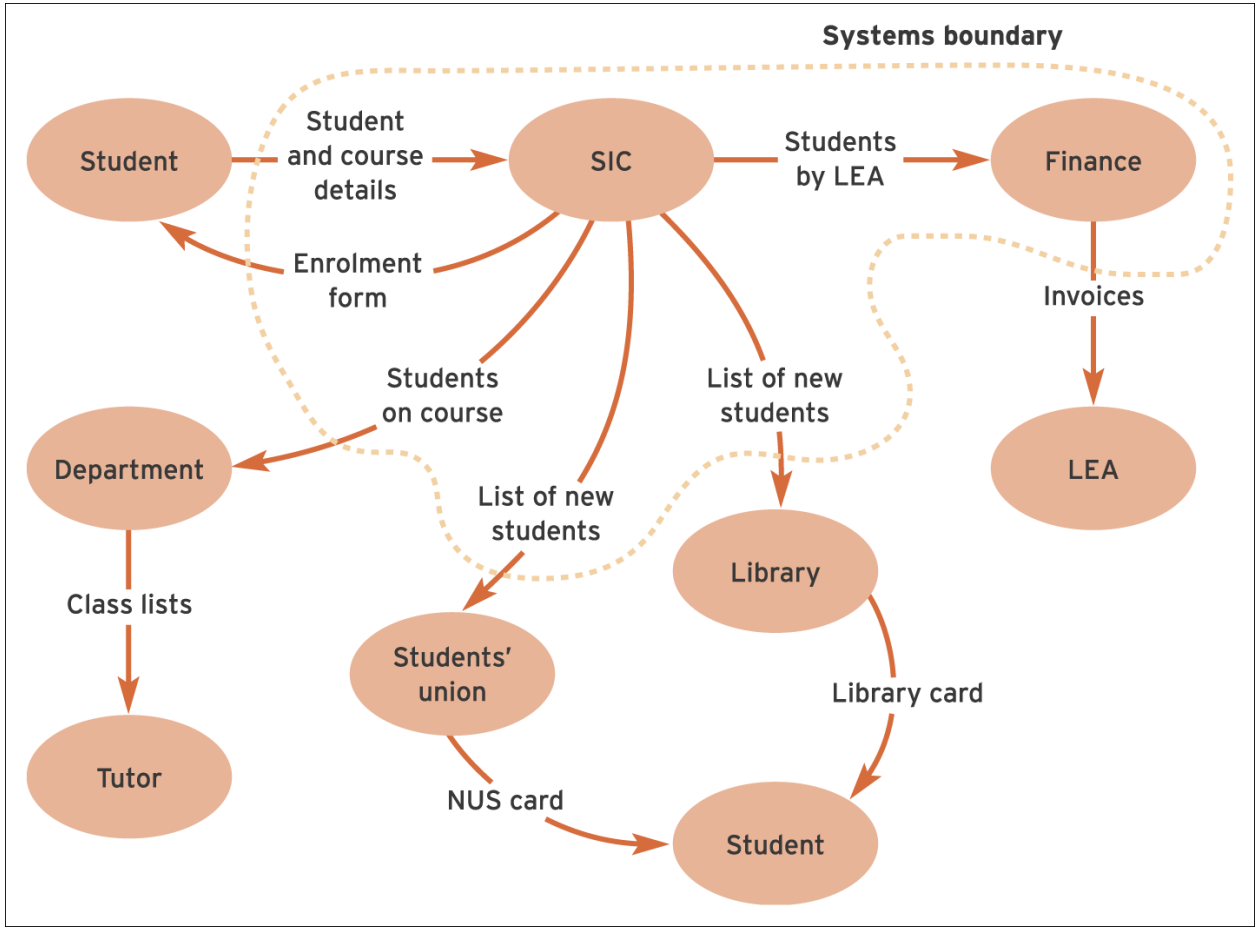


Figure 10.6 The completed IFD, including the system boundary

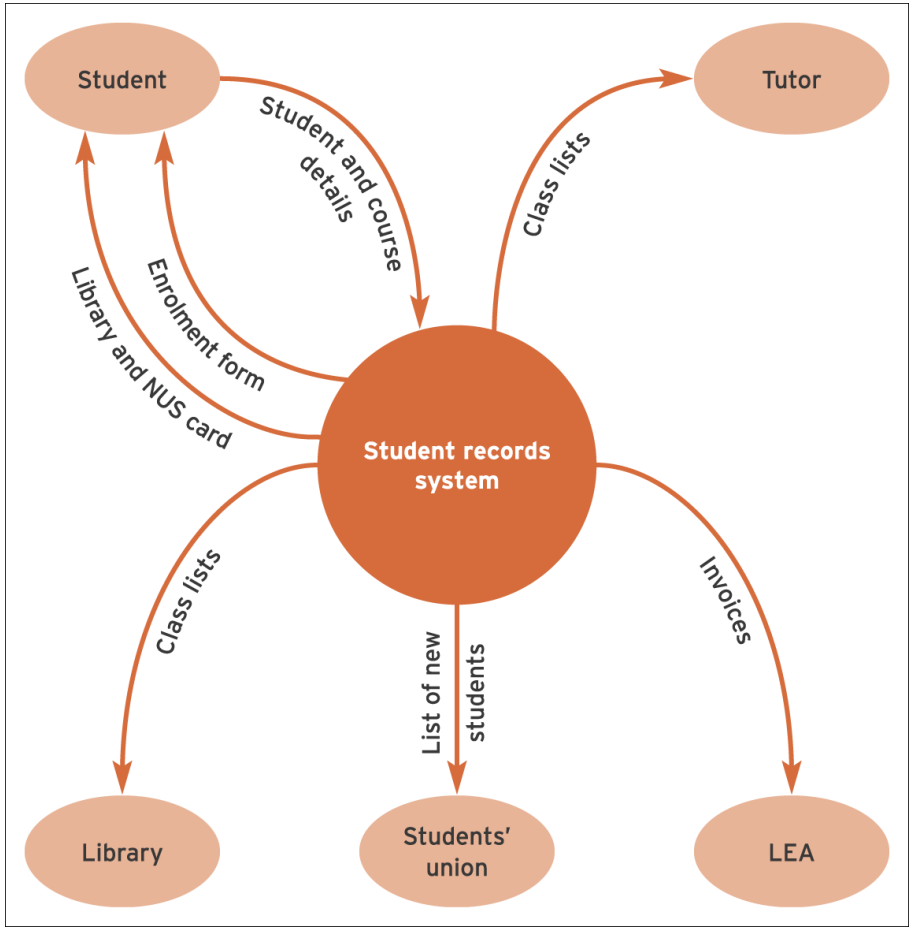


Context diagrams

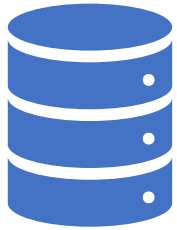
A simplified diagrams that are useful for specifying the boundaries and scope of the system.

They can be readily produced after the information flow diagram since they are a simplified version of the IFD showing the external entities.

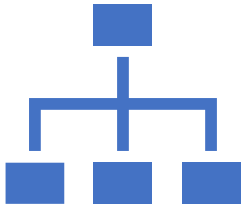
Figure 10.7 Context diagram for the student loan system described in Case Study 10.1



Dataflow diagrams (DFDs)

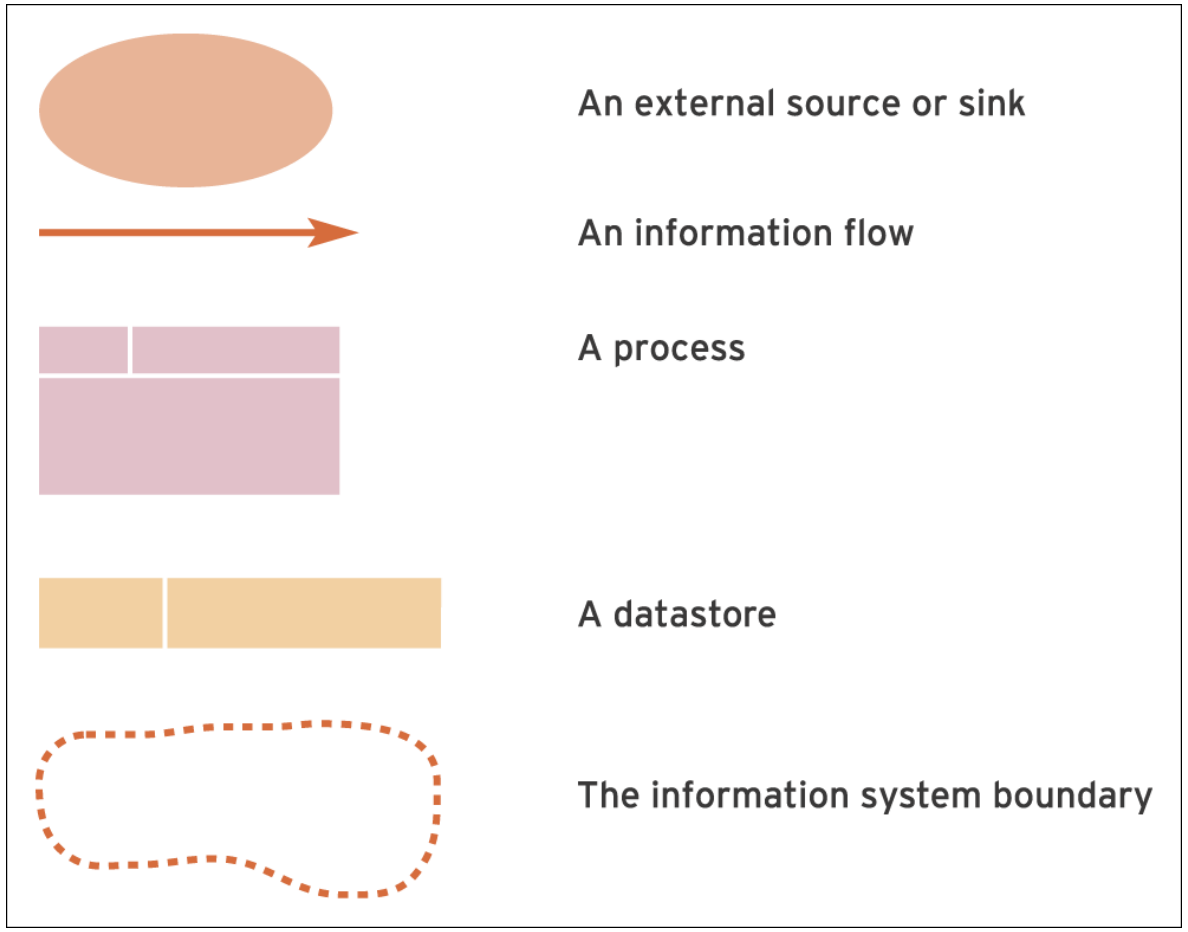


Dataflow diagrams (DFDs): Define the different processes in a system and the information that forms the input and output of the processes. They may be drawn at different levels.



Level 0 provides an overview of the system with Levels 1 and 2 providing progressively more details.

Figure 10.8 Symbols used in dataflow diagrams



DFD elements

Datastores: A datastore can either provide data as input to a process or receive data that has been output from a process.

Dataflows: A dataflow describes the exchange of information and data between datastores and processes and between processes and sources or sinks.

Systems boundary: This remains the same as for an IFD – it indicates the boundary between what lies inside the system under consideration and what lies outside.

DFD – best practice

- Data do *not* flow directly between processes – the data that enter a process will come either from a source or from a datastore, they cannot exist in a vacuum!
- Data do *not* flow directly between datastores – there must be an intervening process that takes the input data and converts them into a new form and outputs them to either a datastore or a sink.
- Data do *not* flow directly from a datastore to a sink, or from a source to a datastore – there *must* be an intervening process.

DFD – five steps

1. Identify and list all *processes* which take place in the system under consideration.
   - A process is an event where an input of some kind, from either a source or a datastore, is transformed into an output (the output being either to a sink or to a datastore).

2. Identify all the datastores which you think exist in the system under consideration.
   - A datastore will exist wherever a set of facts needs to be stored about persons, places, things or events.

DFD – five steps

3. For each process identified in Step 1, identify where the information used in the process comes from (this can be from a *source* or a *datastore* .

4. Draw a ‘mini-DFD’ for each single process, showing the process box and any relevant sources, sinks or datastores.

5. Link the mini-DFDs to form a single diagram, using the datastores to link the processes together.

Figure 10.9 An example of a Level 1 process in a DFD

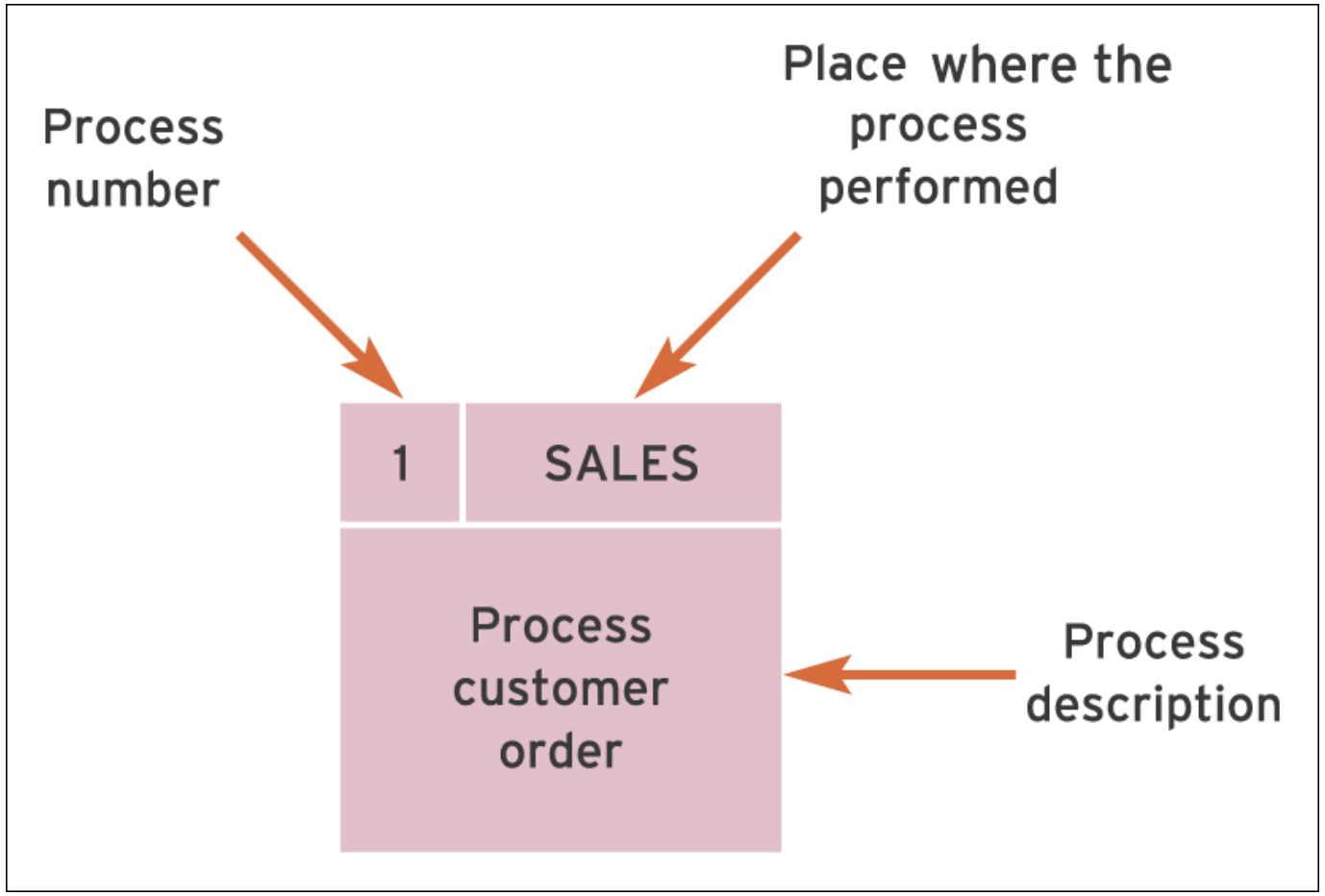


Figure 10.10 An example of a Level 2 process in a DFD

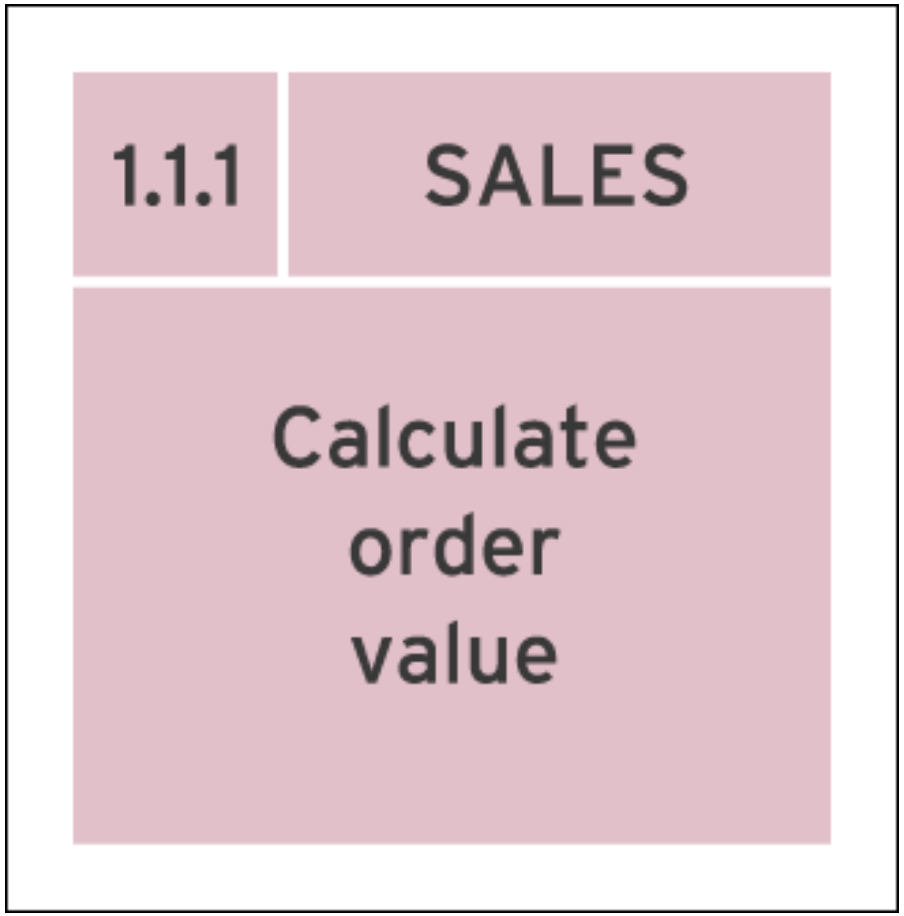
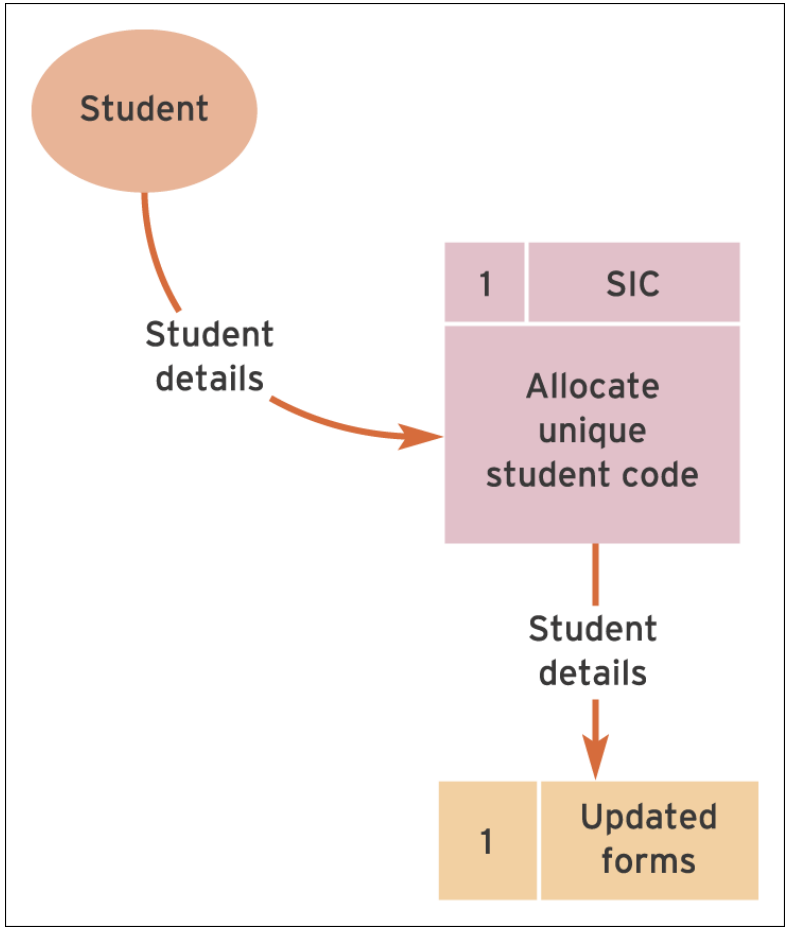


Figure 10.11 An example of a Level 3 process in a DFD

Figure 10.12 Mini-DFD for process 1



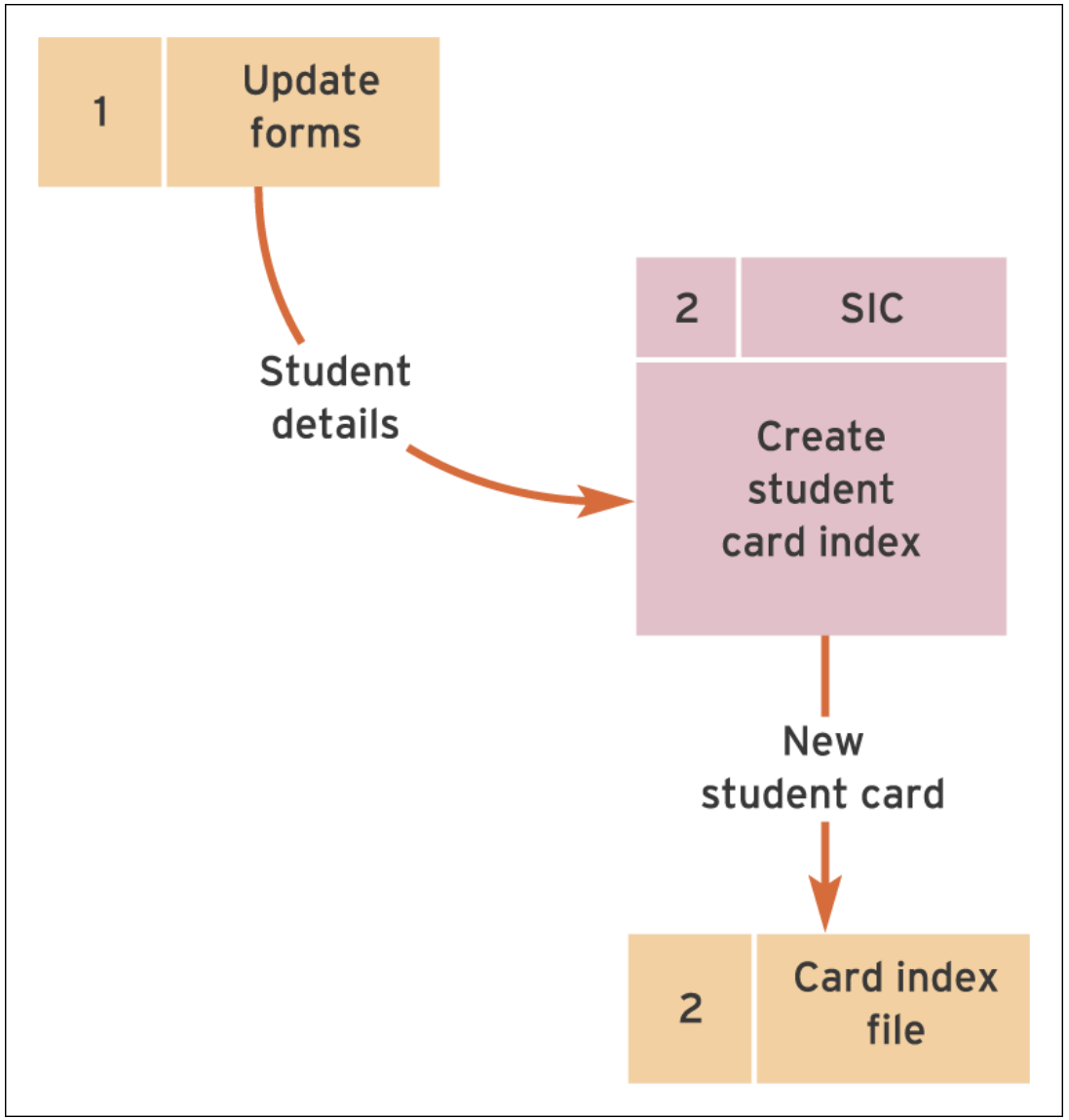


Figure 10.13 Mini-DFD for process 2

Figure 10.14 Mini-DFD for process 3

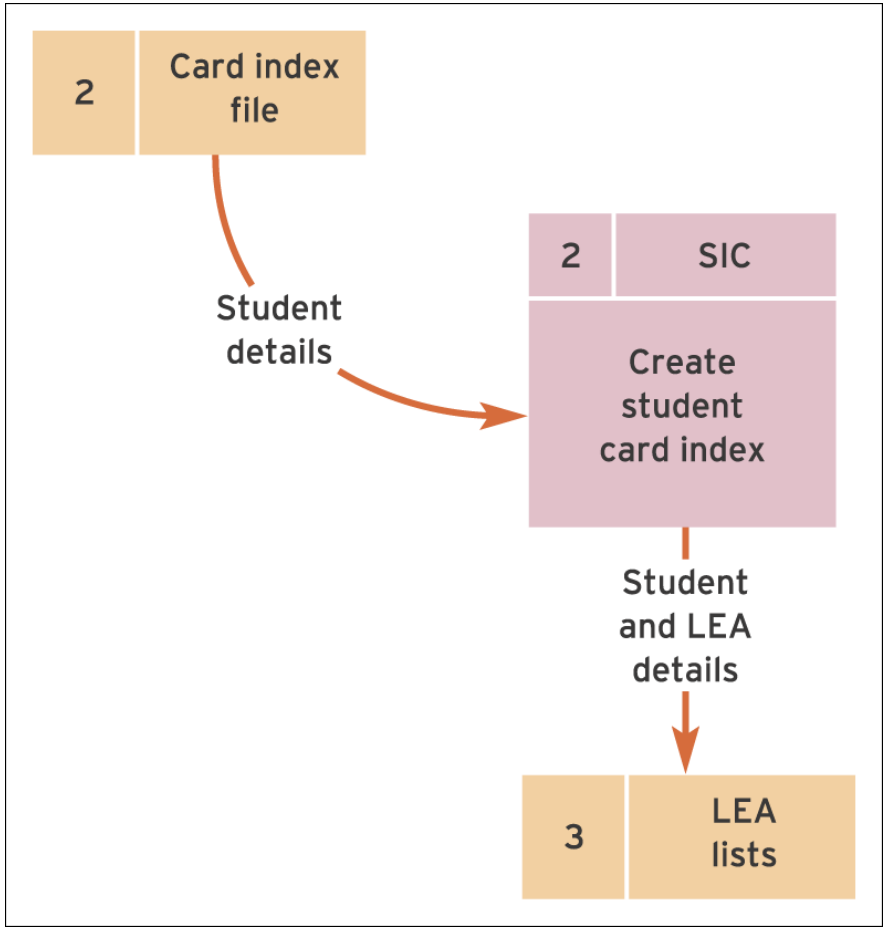
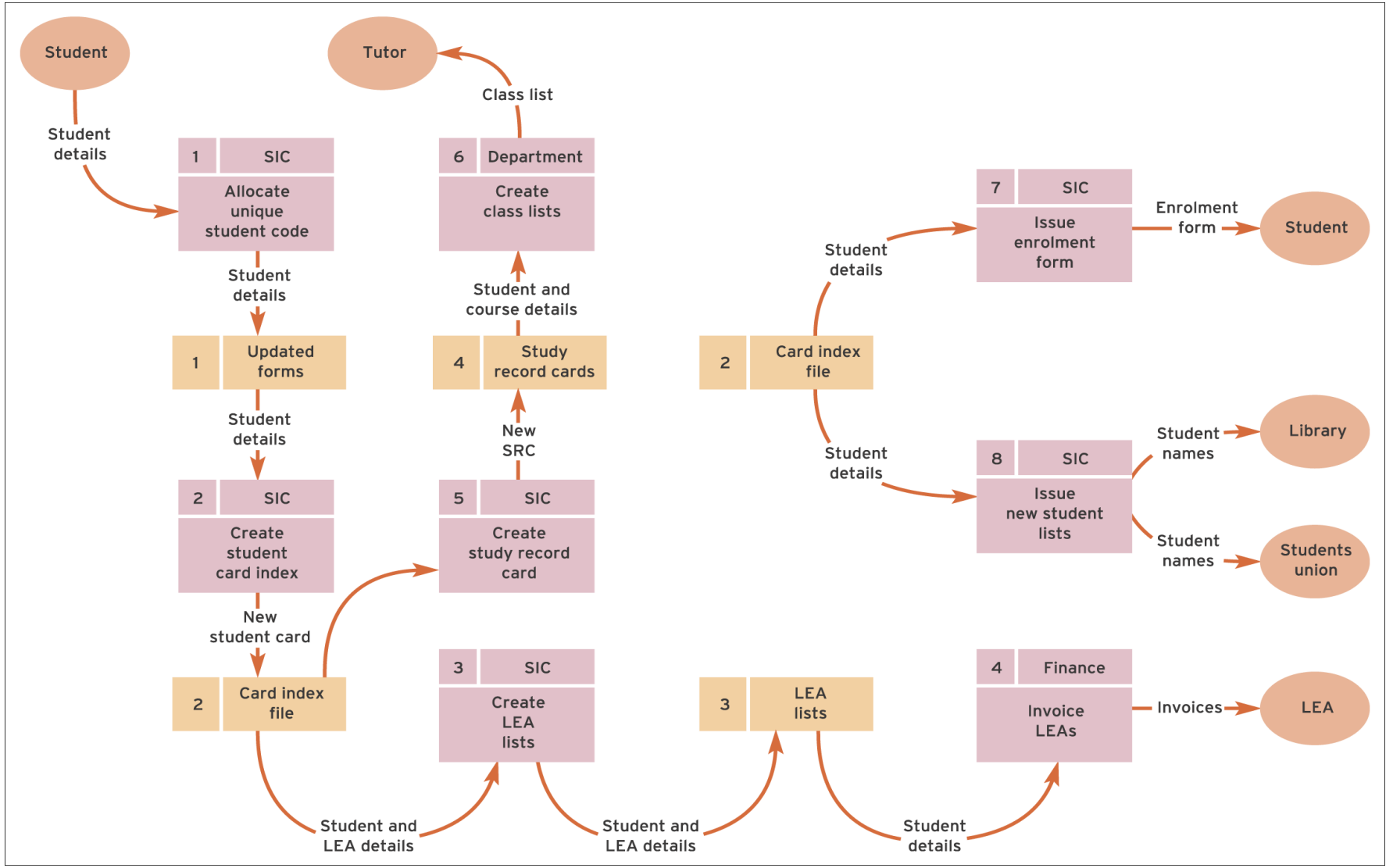


Figure 10.15 Completed DFD for the student record system



Summary

- Data Modelling diagrams
- the importance of conducting the analysis phase to the overall success of the system.
- choosing appropriate techniques for analysing users' requirements for an information system.
- construct appropriate textual descriptions and diagrams to assist in summarising the requirements as an input to the design phase.

Questions