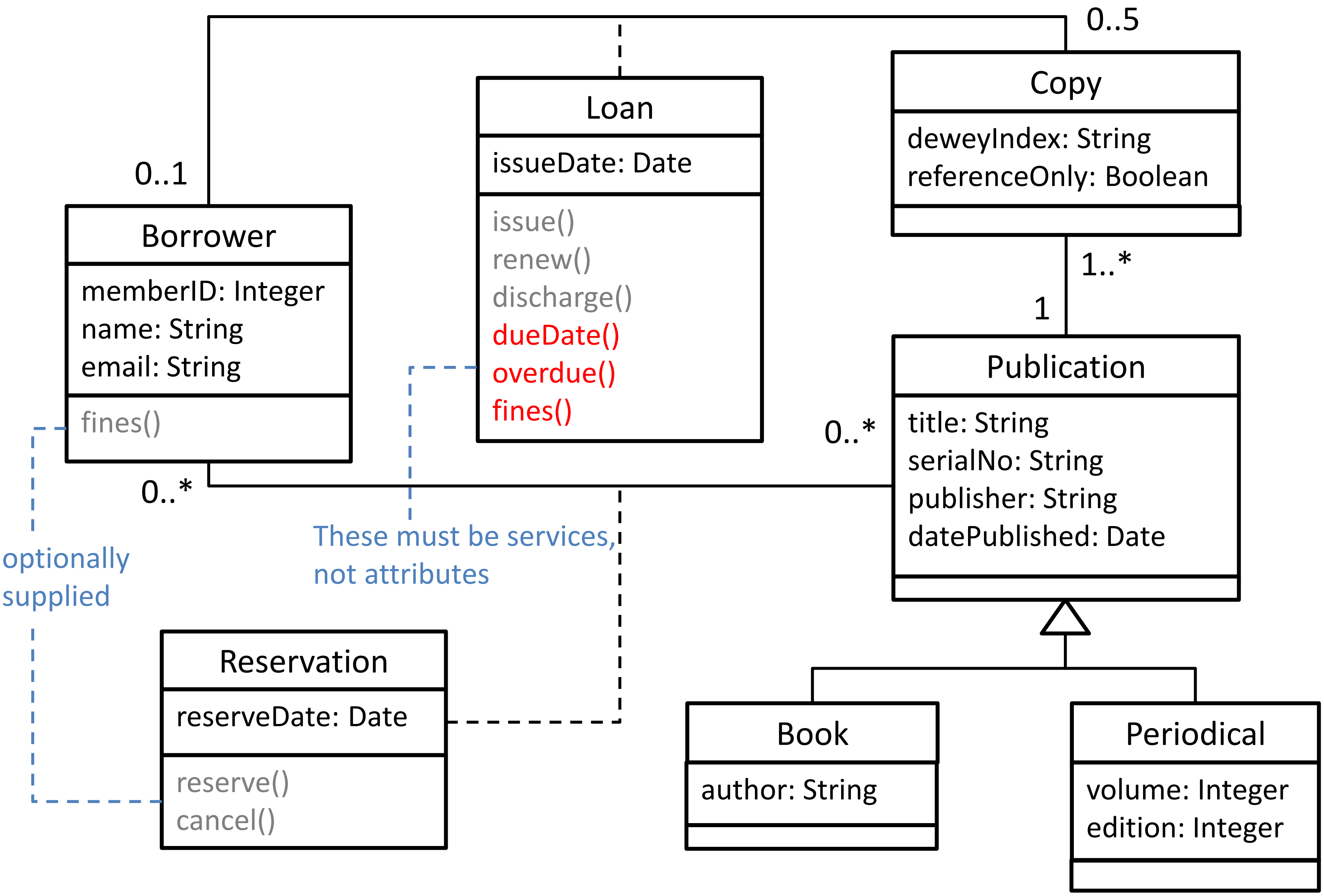
Association end-role names are optional; the multiplicities must be exactly as given



Borrower

memberID: Integer
name: String
email: String

fines()

Loan

issueDate: Date

issue()
renew()
discharge()
dueDate()
overdue()
fines()

Copy

deweyIndex: String
referenceOnly: Boolean

Publication

title: String
serialNo: String
publisher: String
datePublished: Date

Reservation

reserveDate: Date

reserve()
cancel()

Book

author: String

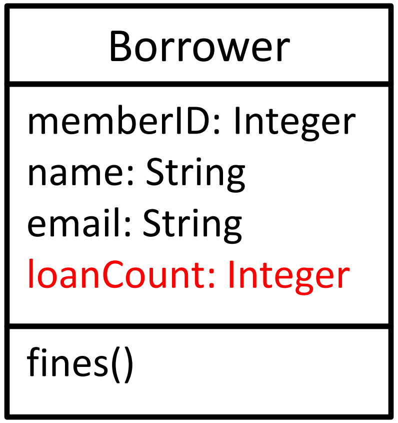
Periodical

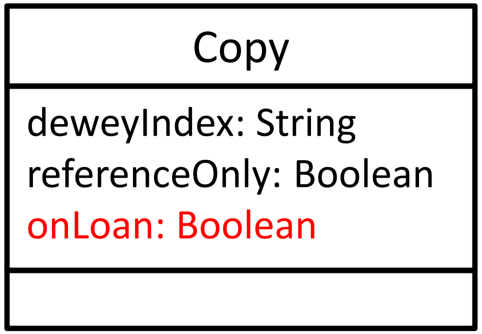
volume: Integer
edition: Integer

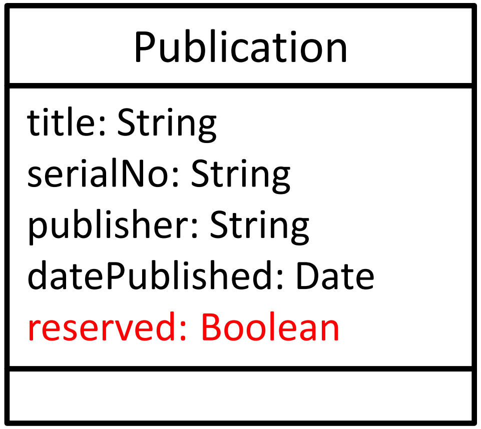
optionally supplied

These must be services, not attributes

Class Diagram – Main Solution





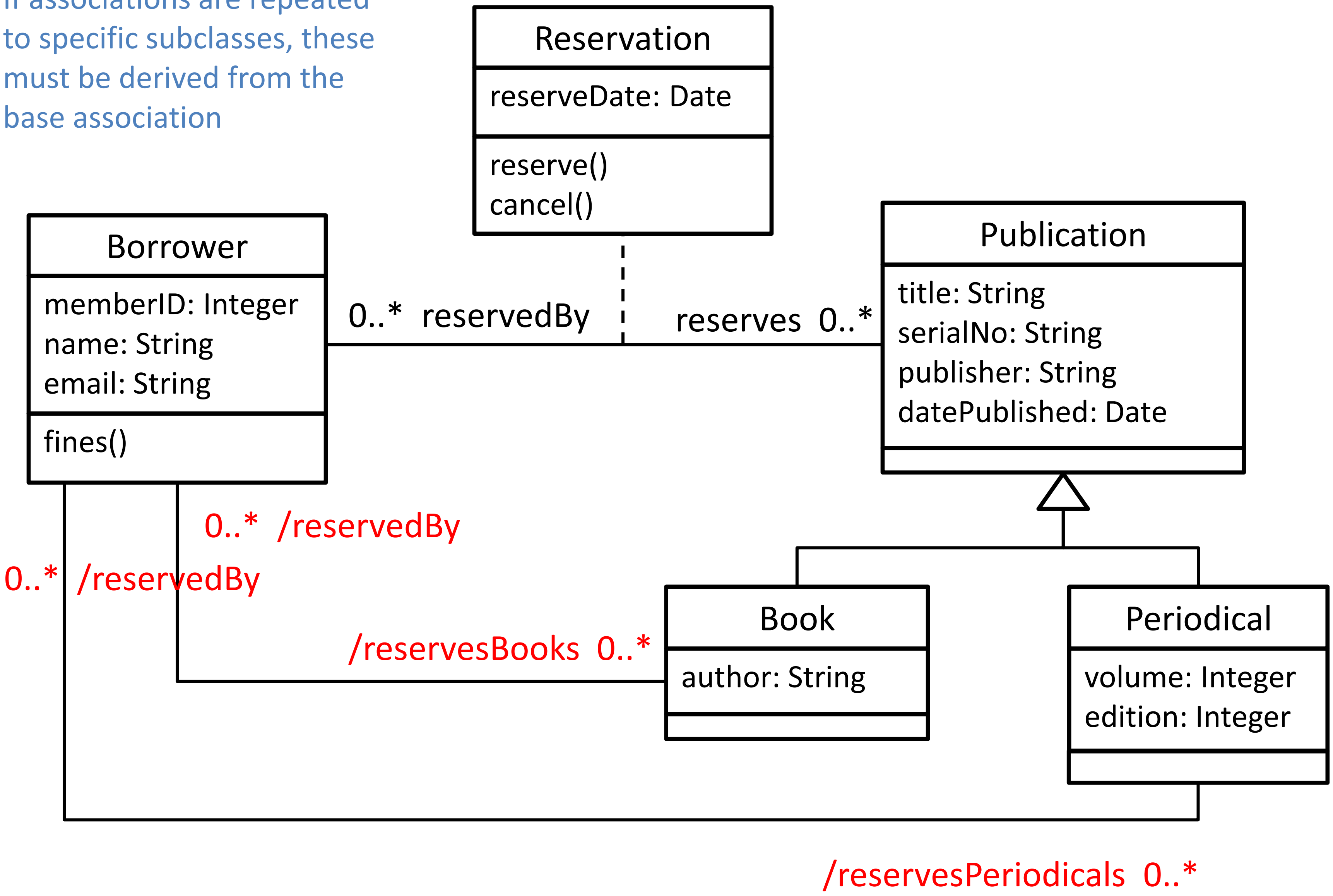


If a class has the extra highlighted attributes, this is wrong because these values should be calculated.

eg: onLoan is determined by the presence of a related Loan.

Class Diagram – mistakes that should lose a mark each

If associations are repeated to specific subclasses, these must be derived from the base association



Class Diagram – allowed variation to Associations

Marking Scheme

- Class Diagram out of 25 points
  - 1 point for identifying each main class (Borrower, Publication, Book, Periodical, Copy) but lose a mark if multiple copy-classes exist [5 total]
    - Name variations OK, eg: LibraryItem (for Publication), BookTitle, PeriodicalTitle, ItemCopy
  - 1 point for each class if all the attributes for each main class above are exactly as given (in main or variant solution) [5 total]
    - Slight attribute name variations OK; some attribute type variations OK (eg: integer for serialNo)
    - lose mark for that class if redundant attributes are supplied, that should be calculcated
  - 2 points for generalisation of Book, Periodical as Publication [2 total]
    - 1 point for trying to show, 2 points for getting arrowhead notation correct, ignore any multiplicities
  - 1 point for each of the 3 associations linking Publication-Copy, Borrower-Copy, Borrower-Publication [3 points]
    - lose marks for multiple copies of associations to separate subclasses, unless derived associations
    - reserving association must link Borrower and Publication (not Borrower and Copy)
  - 1 point each for correct multiplicity at each end of the above [6 total]
  - 1 point each for identifying and linking the association classes (Loan, Reservation) [2 total]
    - Linkage must be syntactically correct, viz. with dashed line
    - I did not ask for a normalised model, but if they have, see me for solution
  - 1 point for each assoc. class if its attributes and services are correct [2 total]
    - Loan’s dueDate(), overdue(), fines() must be a service, not an attribute, as it is calculated