

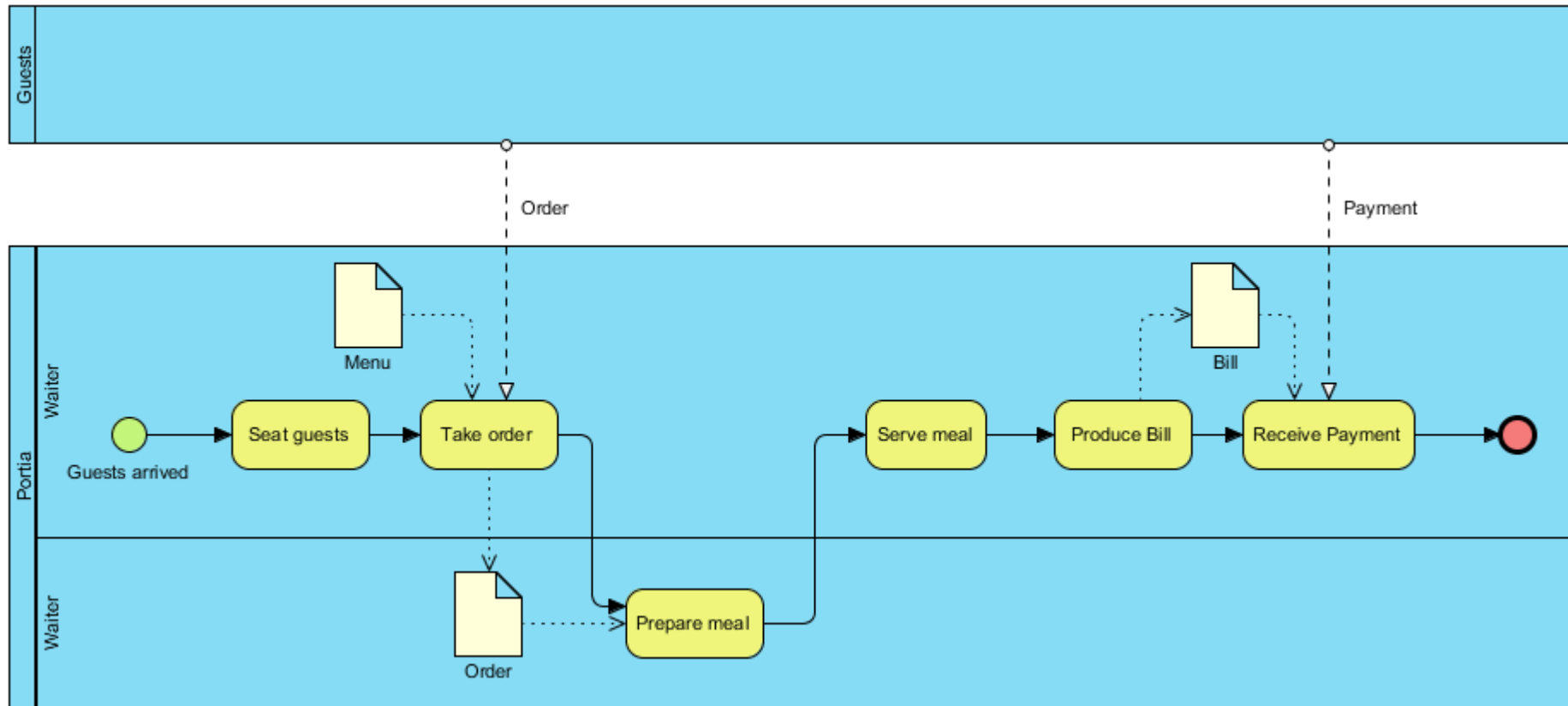


# ***MODELING DATA***



## An Example Process

- This is a simplified version of the process for serving guests
- There are three data objects. Can you see a difference between these data objects?



## *Modelling Data*

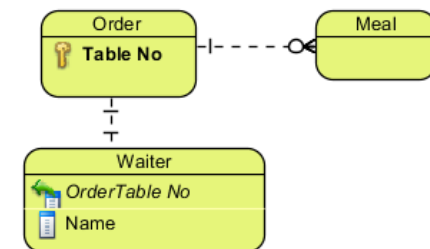
- Data objects in BPMN can represent different kinds of data
  - ◆ structured data
  - ◆ documents
- Documents themselves either represent
  - ◆ a **document class** represents a generic documents for which a specific instance exists for each process instance
    - Example: The bill
  - ◆ a **specific document**
    - Example: The menu which the guests get to choose their meals
    - Hint: For a specific document we can specify a file name or a URL
- Another example: An application form is a specific document while an application would be represented as a class



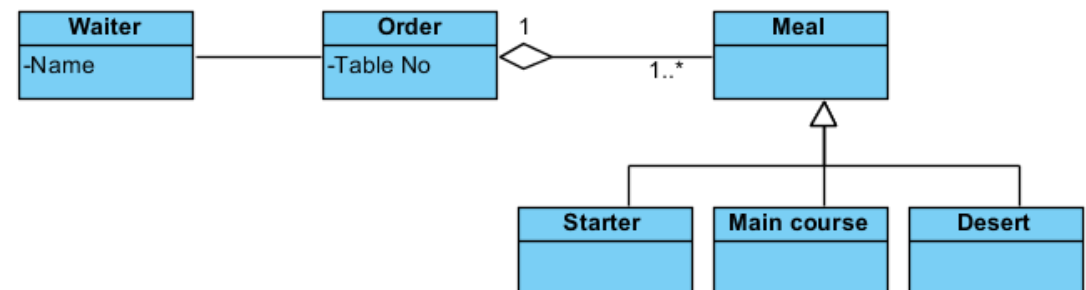
## Modelling Structured Data

- Structured data can be represented for example as
  - ◆ Entity Relationship Diagram
  - ◆ UML Class Diagram/Object Diagrams
- Data models represent
  - ◆ entities/classes
  - ◆ columns/attributes
  - ◆ relations/associations

ERD:



UML Class Diagram:



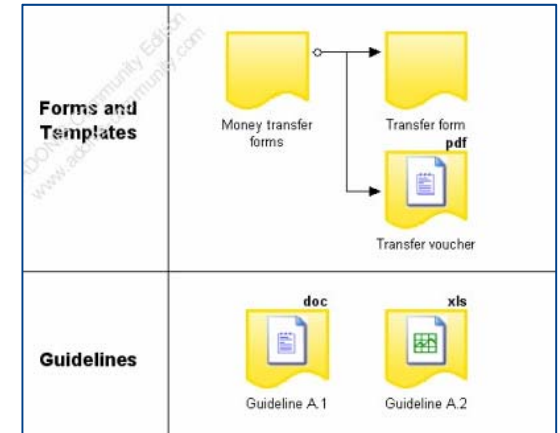
## Document Models

- Documents can be grouped into **document classes** (also called document types) according to their usage:
  - ◆ Examples: invoice, application, menu, report
- There can be specialisations of document classes.
  - ◆ Example: There can be special kinds of reports like project report, expert opinions, or reviews.
- **Metadata** are attribute values which describe documents.
  - ◆ Example: a report might have an creator, a creation date and a subject.
- There are standards for metadata like the Dublin Core Metadata Initiative (<http://dublincore.org>)

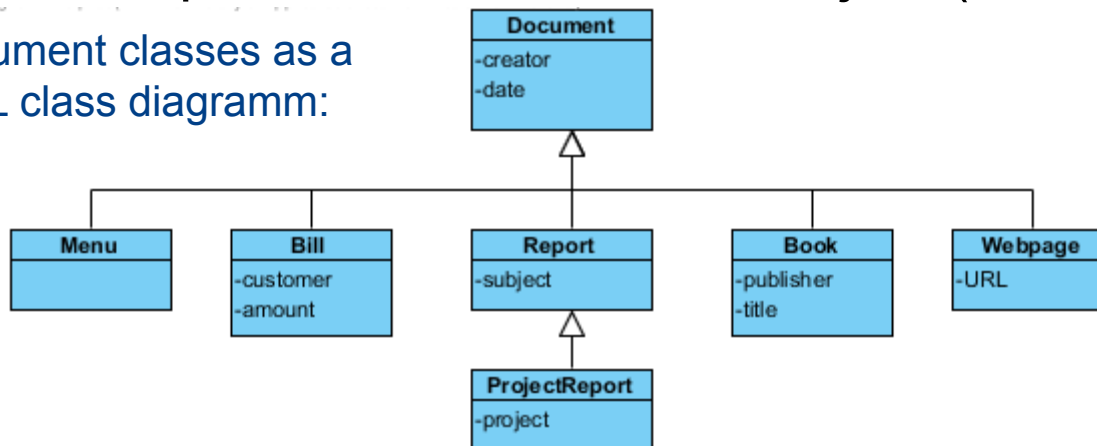
## Document model

- Although some tools like ADONIS have a model type for documents, there is no standard for modeling documents
- However, we can use UML class diagrams and object diagrams to model documents <sup>1)</sup>
  - ◆ A document class is represented as a class object with
  - ◆ A specific document is an object (i.e. an instance of a class)

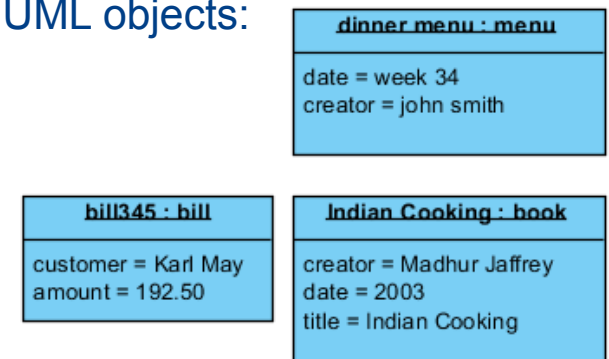
ADONIS document model:



document classes as a UML class diagramm:



specific documents as UML objects:

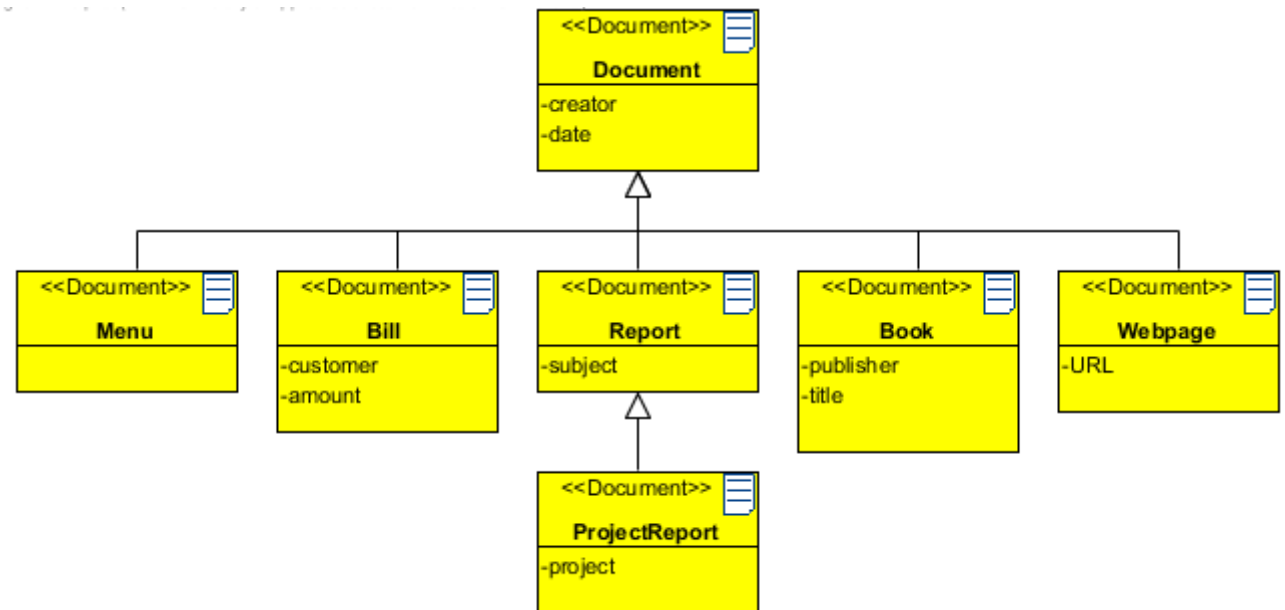


<sup>1)</sup> Remember the UML is used by OMG as the meta-meta-language



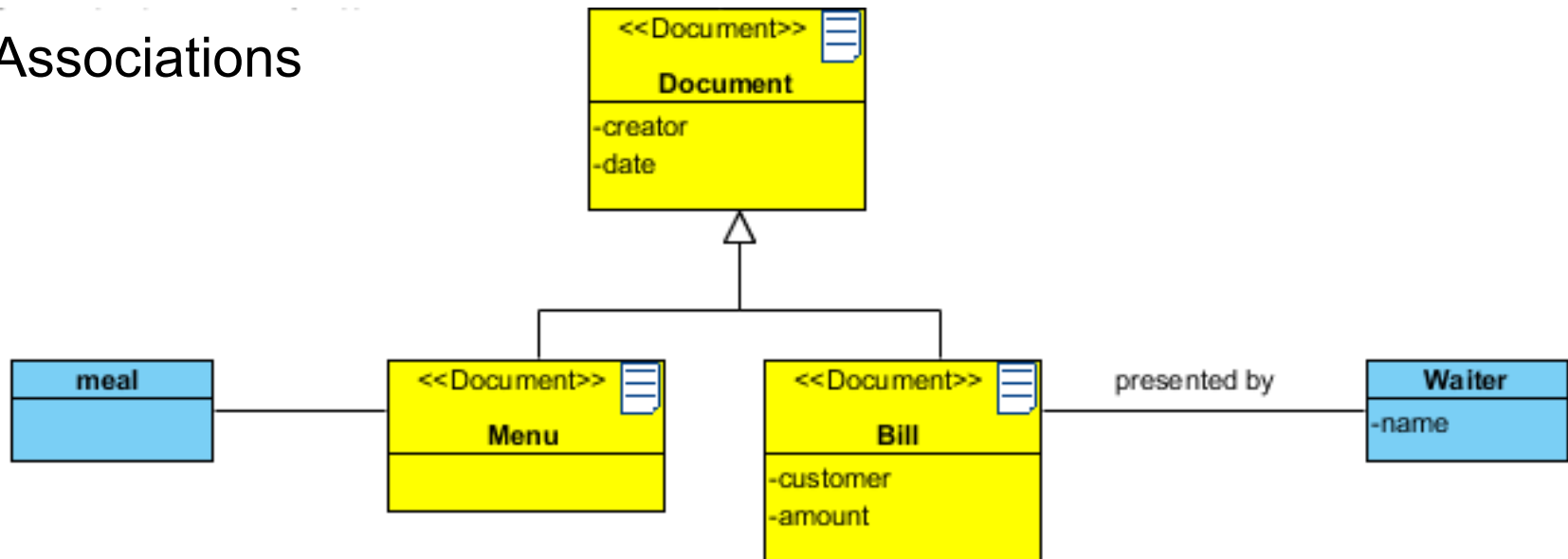
## Modeling Documents in Agilian

- In the Agilian Enterprise tool we can use stereotypes to specialize UML class diagrams for modeling documents.
- We can define a new stereotype "Document" and
  - ◆ change color
  - ◆ add an icon



## A Data Model in Agilian

- A Data Model can combine
  - ◆ Document classes
  - ◆ Objects
  - ◆ Structured Data
  - ◆ Associations







# ***MODELING PRODUCTS***



## ***Product Models***

- Similar to Document Models there is also no standard for product and service models
- Again we can use UML class diagrams to model products
- In Agilian we can again define specific stereotypes





# ***APPLICATION MODELS***



## *Application Models*

- BPMN does not have modeling elements for applications
- However, it might make sense also on business level to model references to applications
- There is no standard for modelling applications on business level
- Different systems have different model types for IT landscape, IT systems etc.
- A possibility would be to use
  - ◆ UML Component Diagrams (although it is more adequate for the IT level)
  - ◆ ArchiMate Diagrams (Application Level)

