

```

<xs:complexType name="CategoryType">
  <xs:sequence>
    <xs:element name="description" type="xs:string" />
    <xs:element name="category" type="CategoryType"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="books">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="book" type="BookType"
            minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

Introduction to UML

Martin Nečaský

Dept. of Software Engineering

Faculty of Mathematics and Physics

Charles University in Prague



What is UML?

- Unified Modeling Language
- standard graphical notation for modeling software systems from different viewpoints
 - architecture
 - data structures
 - processes
 - functions
 - states
 - communication
 - etc.
- and at different levels of abstraction
 - conceptual
 - implementation

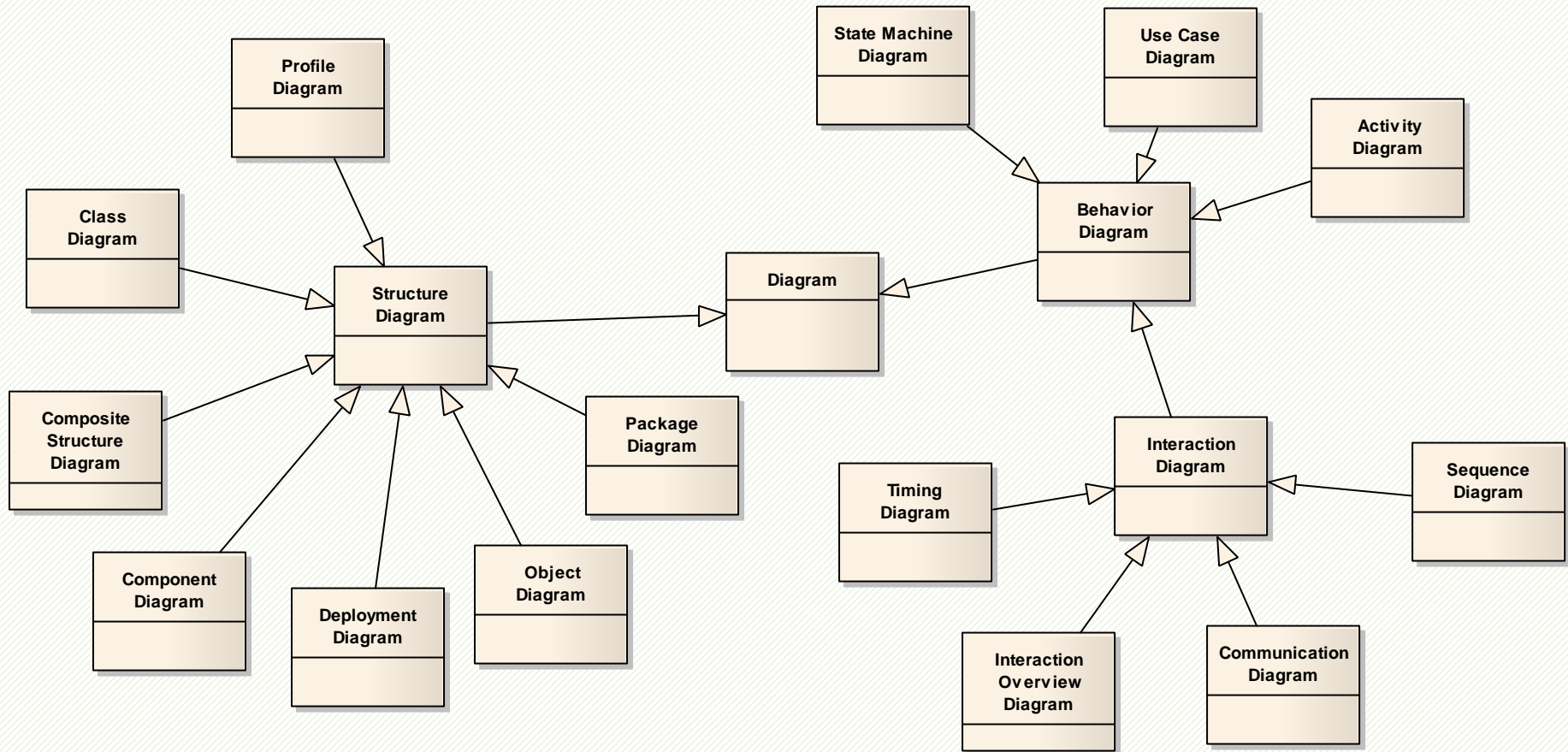
What UML is not?

- ❑ replacement for textual programming languages
 - at least nowadays
- ❑ methodology
 - it does not say how to use it
- ❑ universal language
 - it is not a single language for any problem

Why UML?

- ❑ (standardized) graphical notation is a natural way of communication and understanding between different stakeholders and inside a development team
 - ISO standard
- ❑ allows to highlight important properties and features while avoiding unnecessary details
 - → abstraction of software/data/processes
- ❑ analysts, designers, architects, developers started to use it widely
- ❑ well supported by different CASE tools

Classification of UML Diagram Types



Basic Terminology

Model vs. Schema vs. Diagram

- **model** = modeling language
 - a set of constructs you can use to express something
 - e.g. the UML class model = {class, attribute, association}
 - e.g. the relational model = {relation schema, attribute}
- **schema** = expression in a given modeling language
 - an instance of a model
 - e.g. a relational schema = {Person(name, email)}
- **diagram** = schema visualization