

# Zeus-Framework

## Lesson 1 X-Objects

# Index

- Framework of classes
- Concept of Module Object Model (MOM)
- Implementation of MOM
- X-Objects

# Framework of Classes

The framework includes following classes:

- Data types
  - Lists (queues, stacks, linked lists and array lists)
  - Maps (Set)
  - Strings
  - File and directory classes
  - Streaming classes

# Framework of Classes

- API classes
  - Threading
  - Networking
  - Messaging
- Implementation of various concepts
  - MOM with X-Objects
  - Cell Computing Model
  - Serializing of Objects
  - RMI for C++

# Framework of Classes

- Classes for artificial intelligence:
  - Genetic algorithms
  - Fuzzy logic

# MOM

- High reusability
- Decentralized software development
- Stepwise development.
- Extensible software components.
- Flexible configuration
- Fast maintenance of complex systems.

# MOM (Basic)

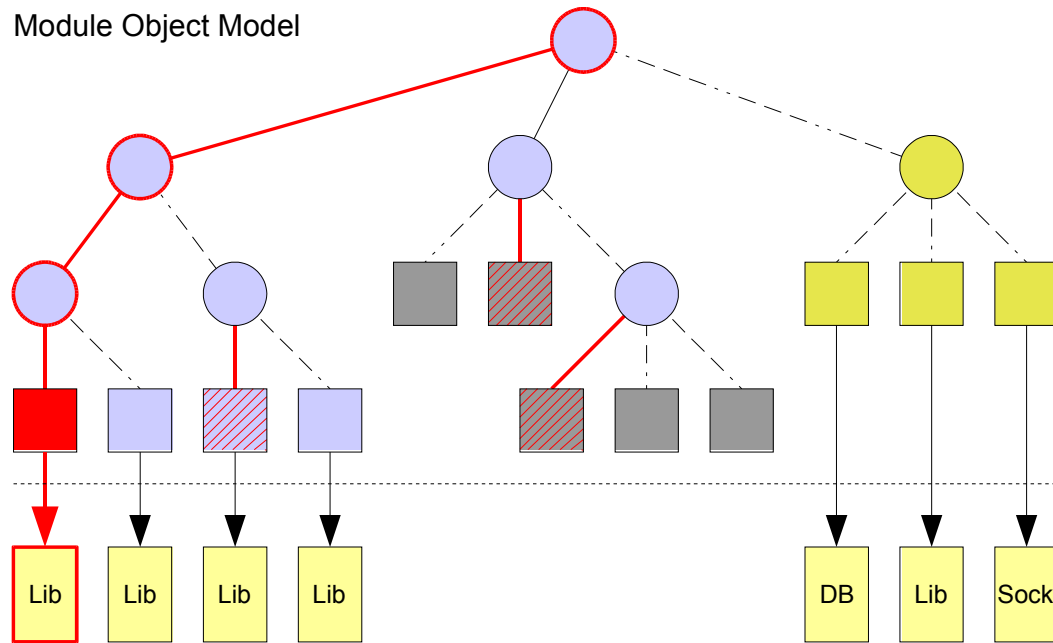
- X-Objects can be leaf nodes. They provide a service inside the software system.
- X-Objects can be nodes to group other X-Objects to logic units.
- X-Objects are either frozen or unfrozen. They can be switched to both states by other components.
- Freezing an X-Object, all owned references are released. Frozen objects don't respond any more.
- While unfreezing an object, it can estimate all references needed.
- The MOM defines the communication interface from X-Objects.
- An X-Object might load its children delayed. This technique is used to create child objects as late as possible. This saves time starting up the system.

# MOM (Ext.)

- Modules are being activated or deactivated.
- Modules are able to load resources.
- Managers are modules and able to manage modules and group them to logic units.
- Are modules used inside an object tree, its important that each module is owned by a manager. This guarantees the whole functionality of the extended MOM specification.

# MOM (Ext.)

Module Object Model

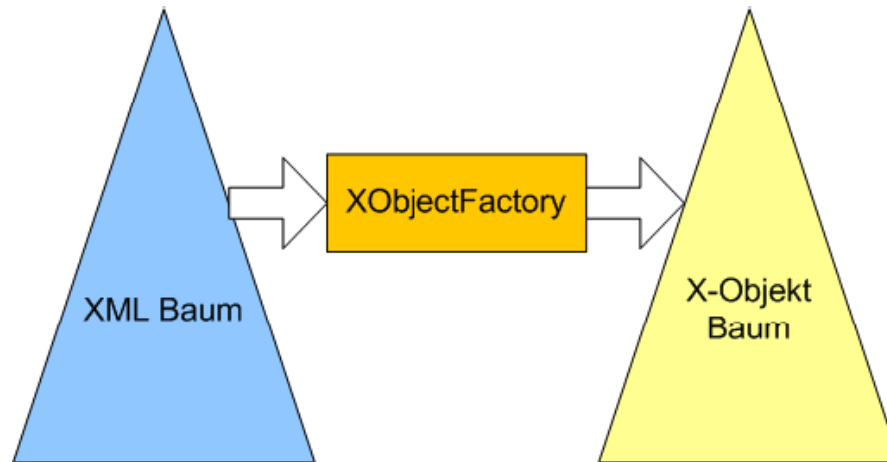


Ressourcen

- System Module und Dienstleistungen.
- Geladene, nicht aktive Module
- Nicht geladene Module
- Aktives Modul
- Anwärter für „aktives Modul“
- lib Ressource eines Moduls

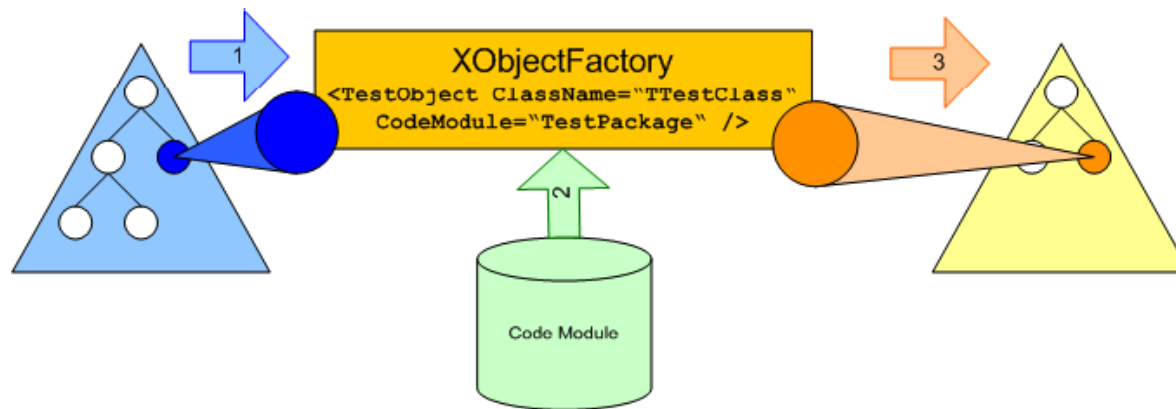
# X-Objects

- Creates objects out of an XML structure
- Tree oriented.
- Persistence layer

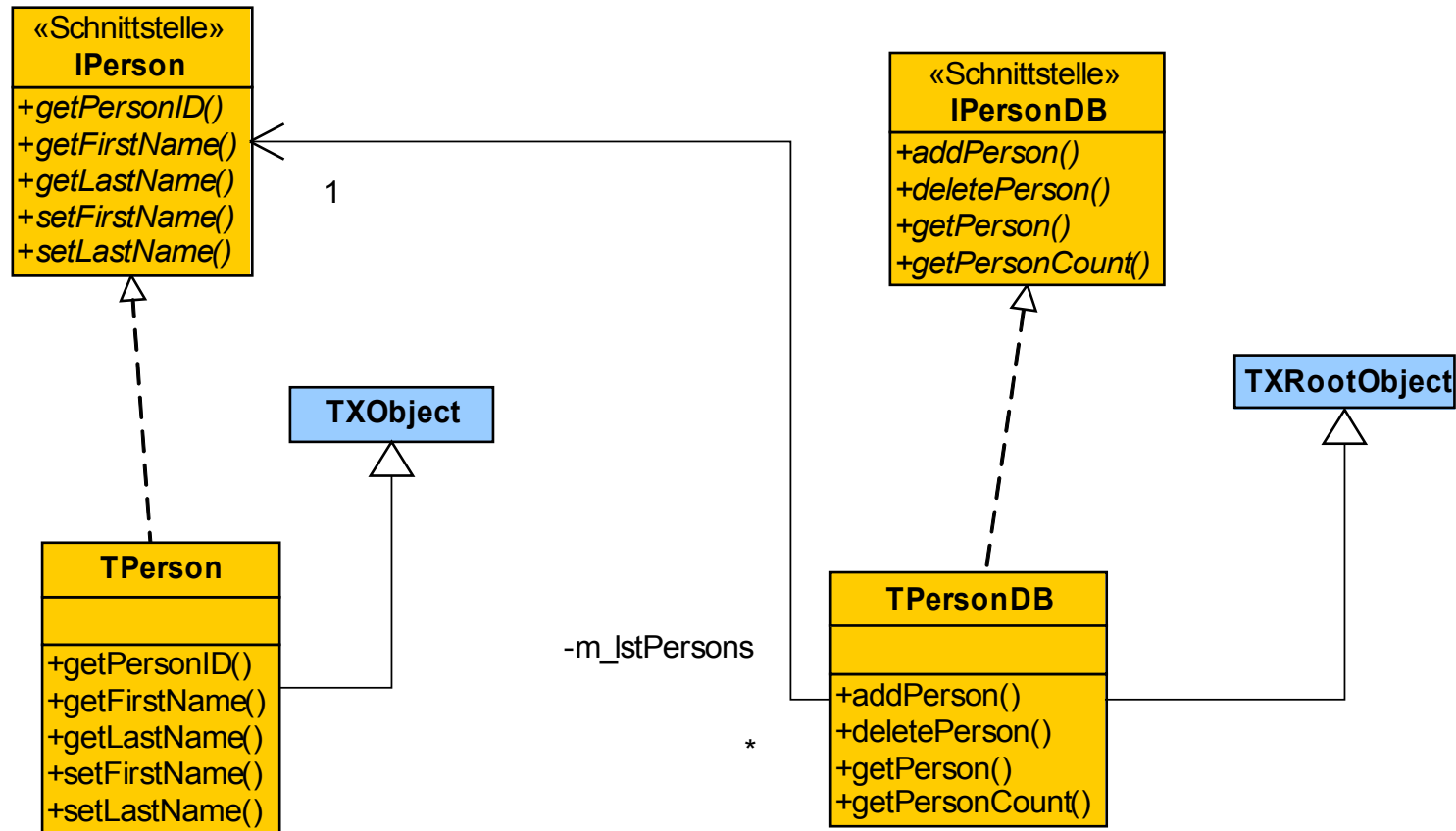


# X-Objects

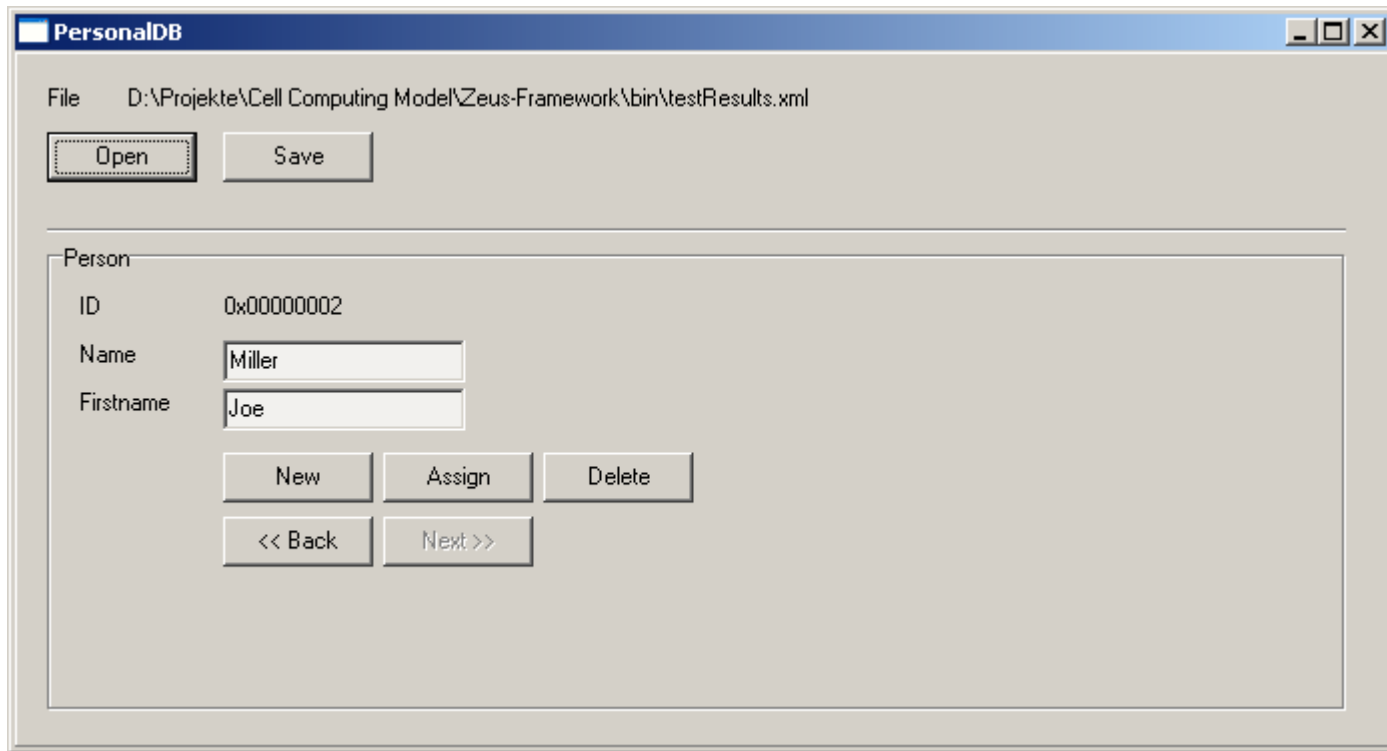
- Each XML node represents a X-Object
- Classes are loaded on demand.
- Name, ID, name of class and code module (library)



# Example of X-Objects



# Example of X-Objects



- Graphic Layout of Personal Database

# Example of X-Objects

- XML data file:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<DB ClassName="TPersonDB" MaxID="2">
  <Person ClassName="TPerson"
    ID="1"
    LastName="Benjamin"
    Name="Hadorn" />
  <Person ClassName="TPerson"
    ID="2"
    LastName="Miller"
    Name="Joe" />
</DB>
```

# Questions

- Questions?

