



**Eike Stepper**

[stepper@esc-net.de](mailto:stepper@esc-net.de)  
<http://www.esc-net.de>

ES-Computersysteme  
Berlin, Germany



# CDO Model Repository

A Model Repository for EMF

A Framework for Distributed Shared Models



# Agenda

## § **EMF basics**

- § EMF persistence framework
- § XML file persistence

## § **CDO client overview**

- § Distributed shared models
- § Detailed design review

## § **CDO server overview**

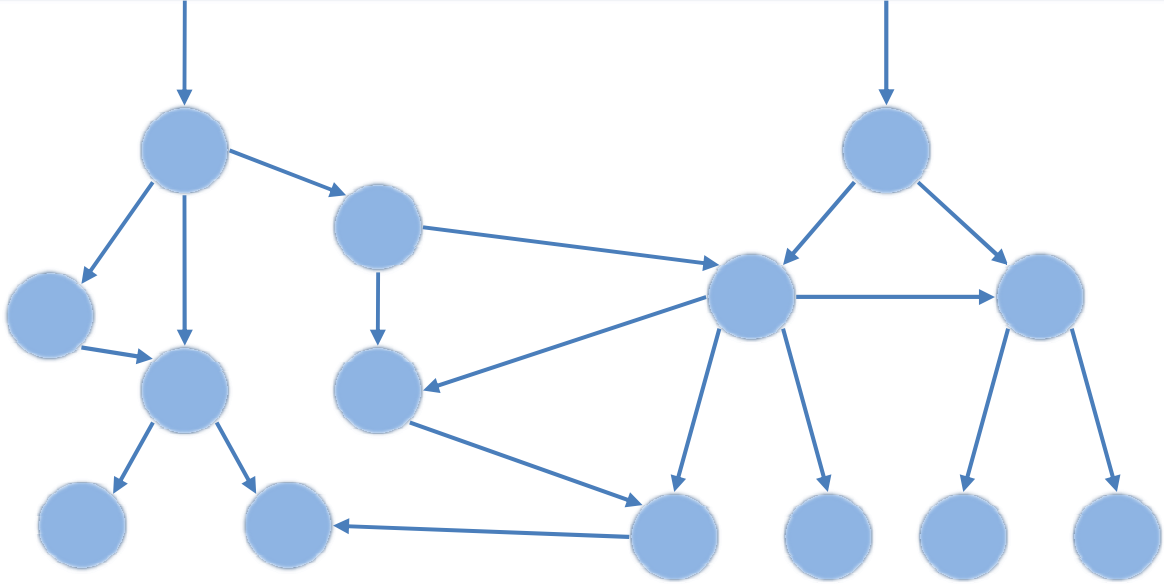
- § Brief design introduction

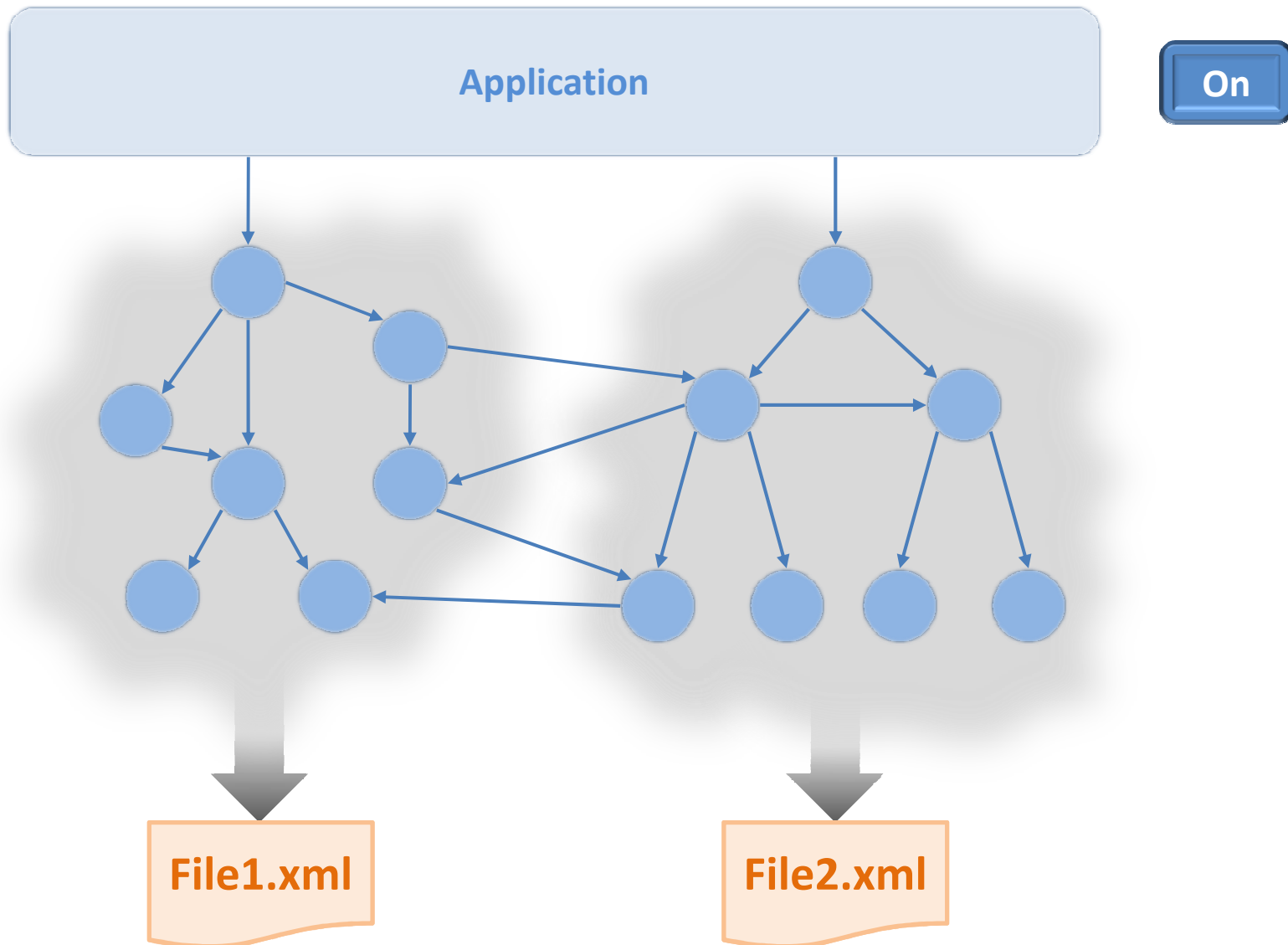
**Application**

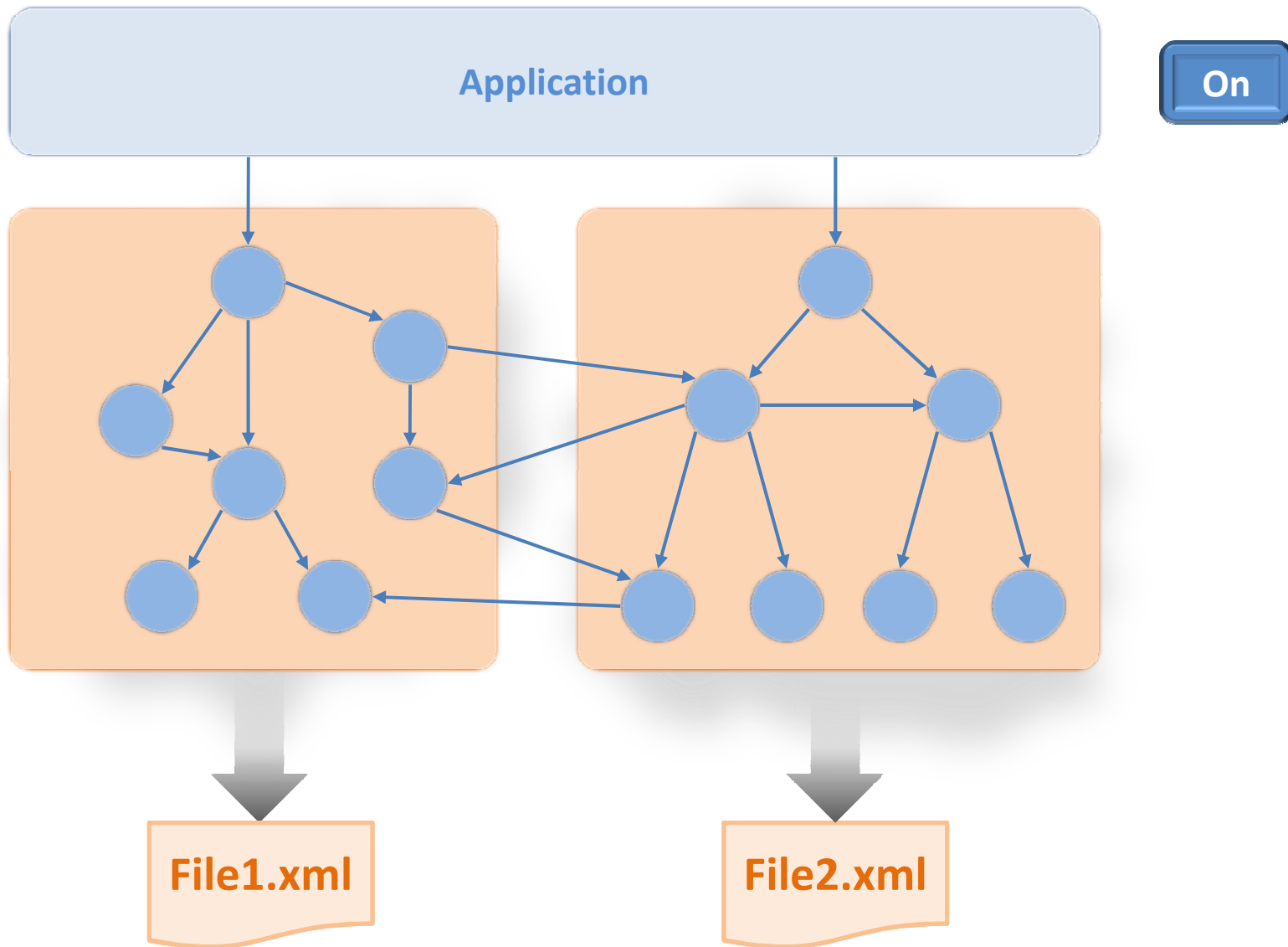
**On**

Application

On

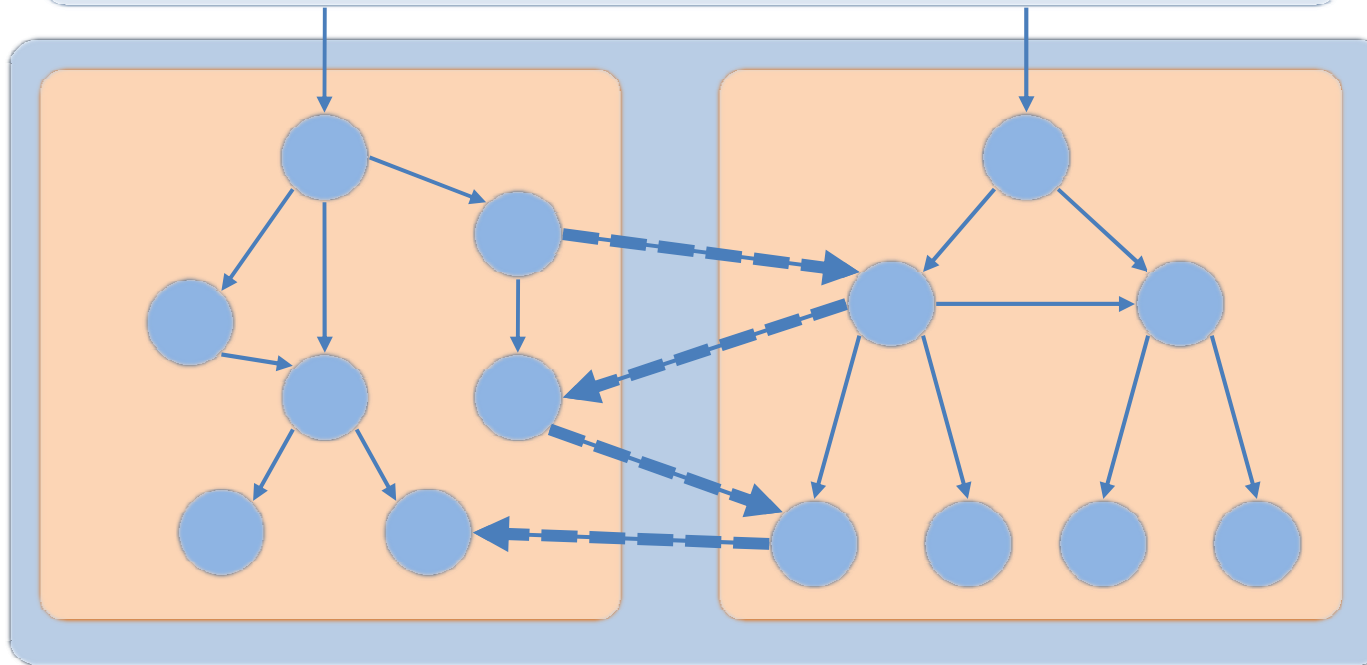






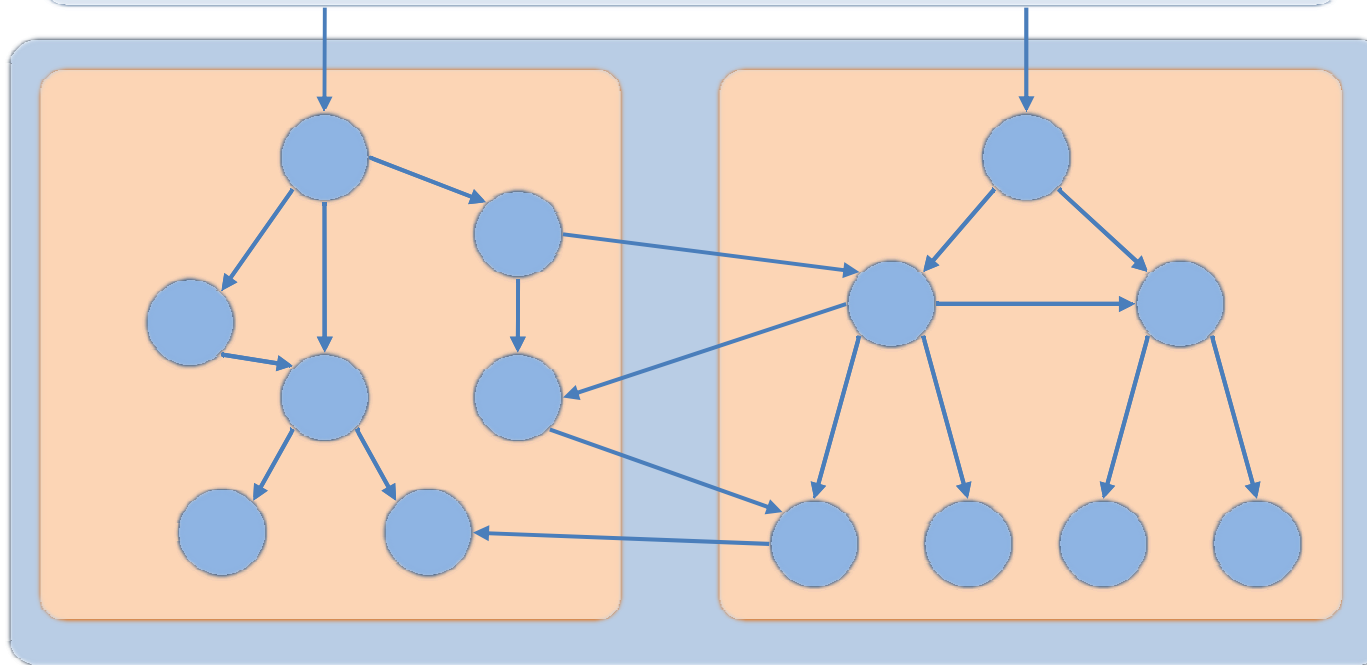
# Application

On



# Application

On



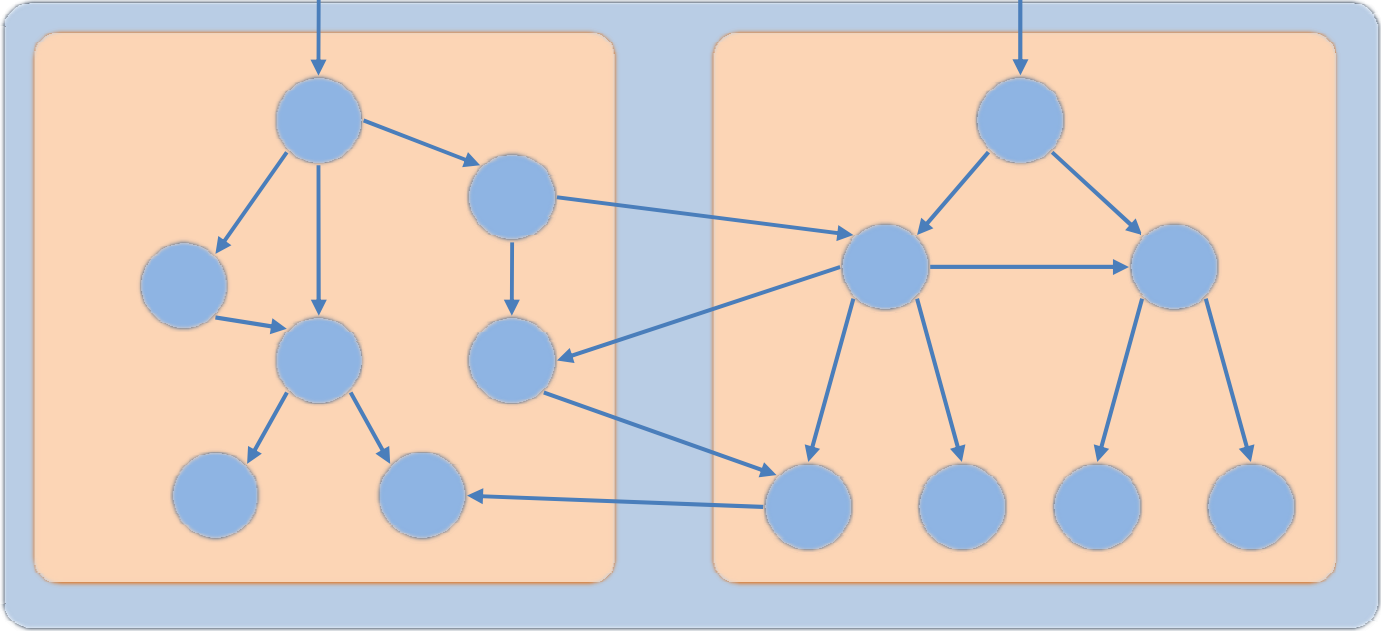
**File1.xml**

**File2.xml**



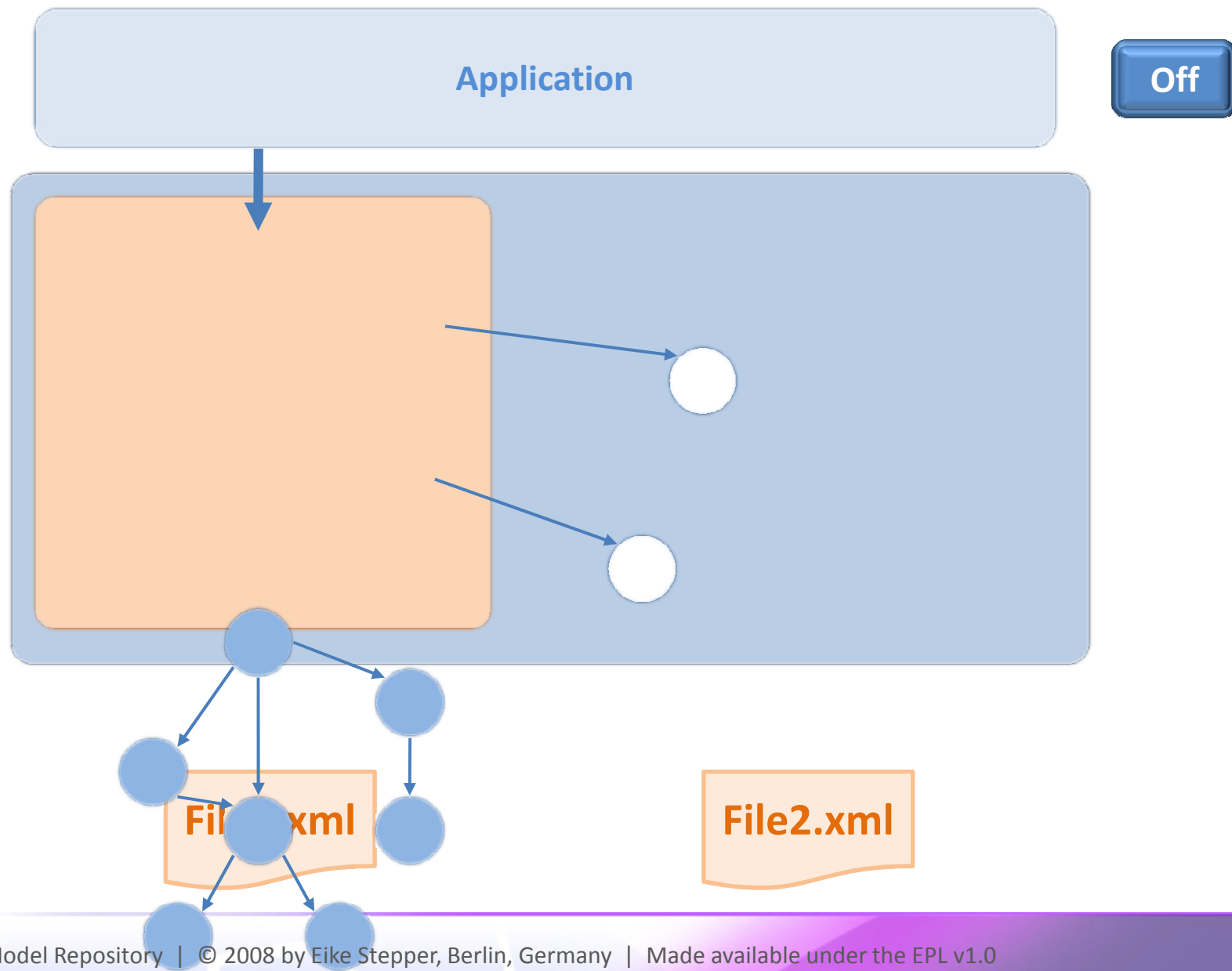
Application

Off



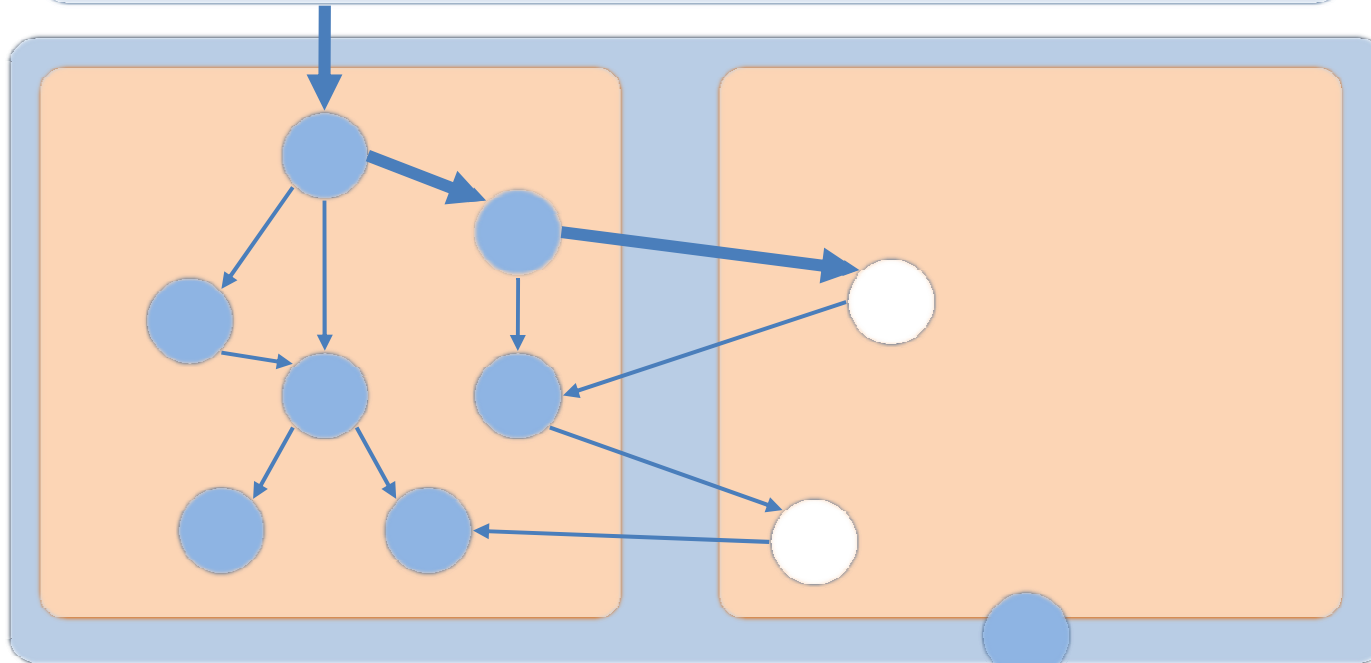
File1.xml

File2.xml



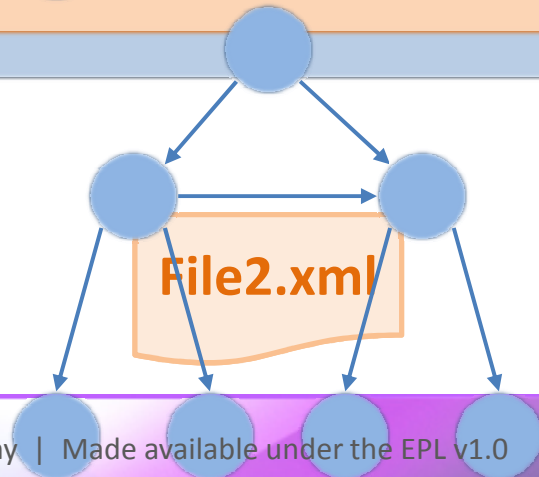
# Application

On



File1.xml

File2.xml



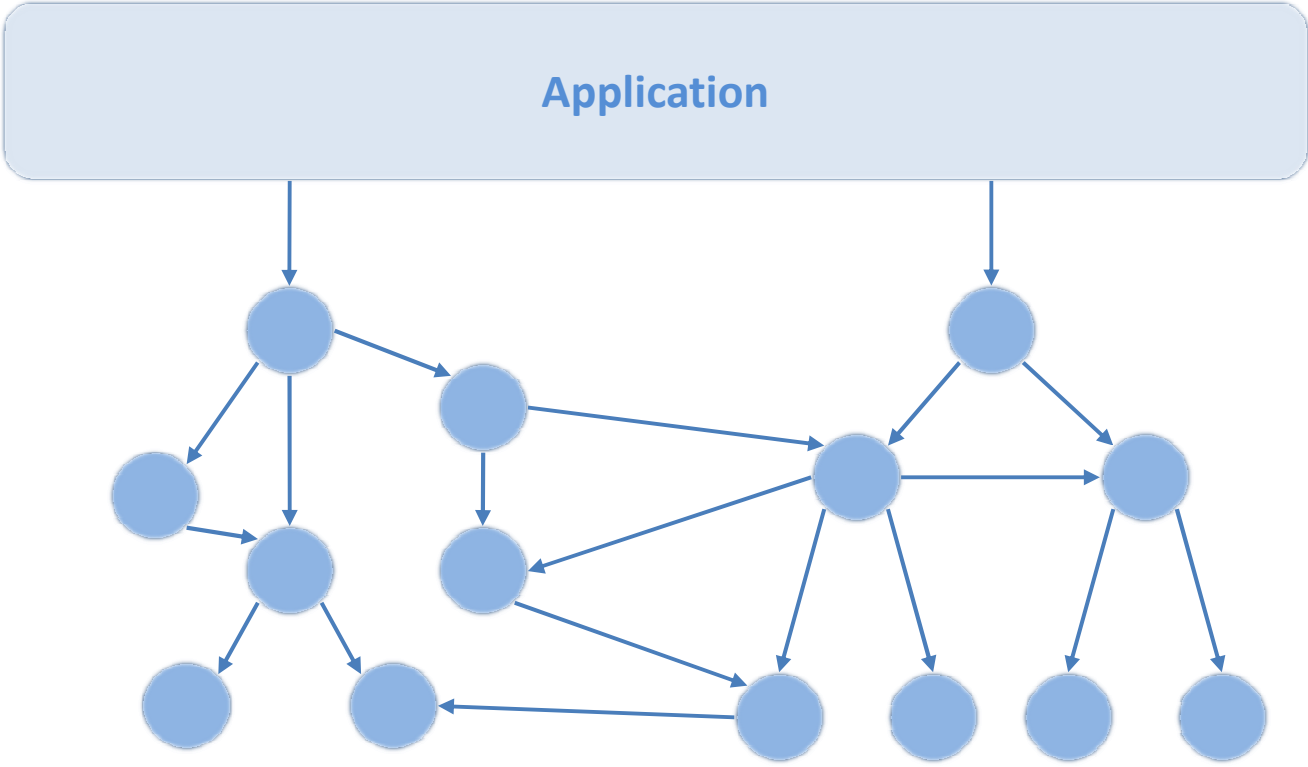
# Recall: With EMF you can...

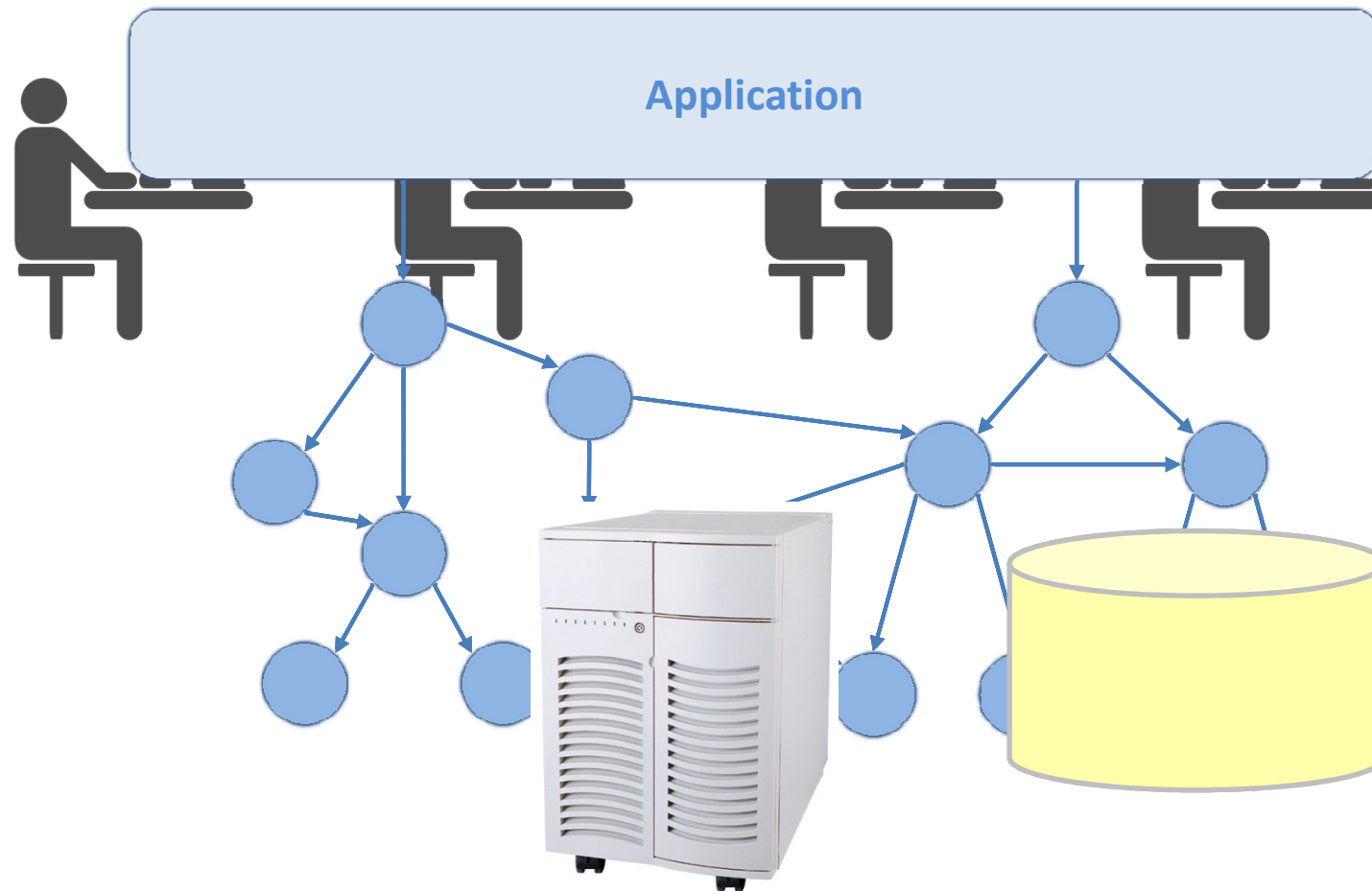
- § **Create Ecore models**
- § **Configure generator models**
- § **Generate Java code for**
  - § Your Ecore model
  - § Command framework
  - § Eclipse UI
- § **Use the EMF persistence framework**
  - § Save and load XML files
  - § Resolve cross references across files

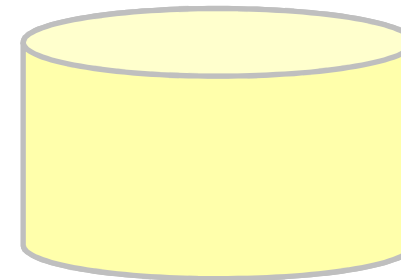
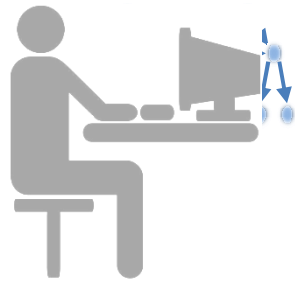
# With XML files you can **NOT**...

- § **Load single instances lazily**
  - § Resource size is limited
- § **Load collections partially**
  - § Collection size is limited
- § **Unload single instances transparently**
  - § Bad influence on footprint (garbage collection)
  - § Bad influence on model design (containment)
- § **Work concurrently**
  - § No transactions
  - § No fine grained locking
  - § No remote update notification

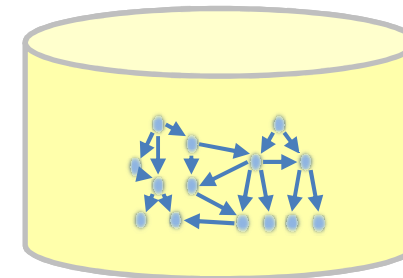
***They just don't behave like multi-user databases***

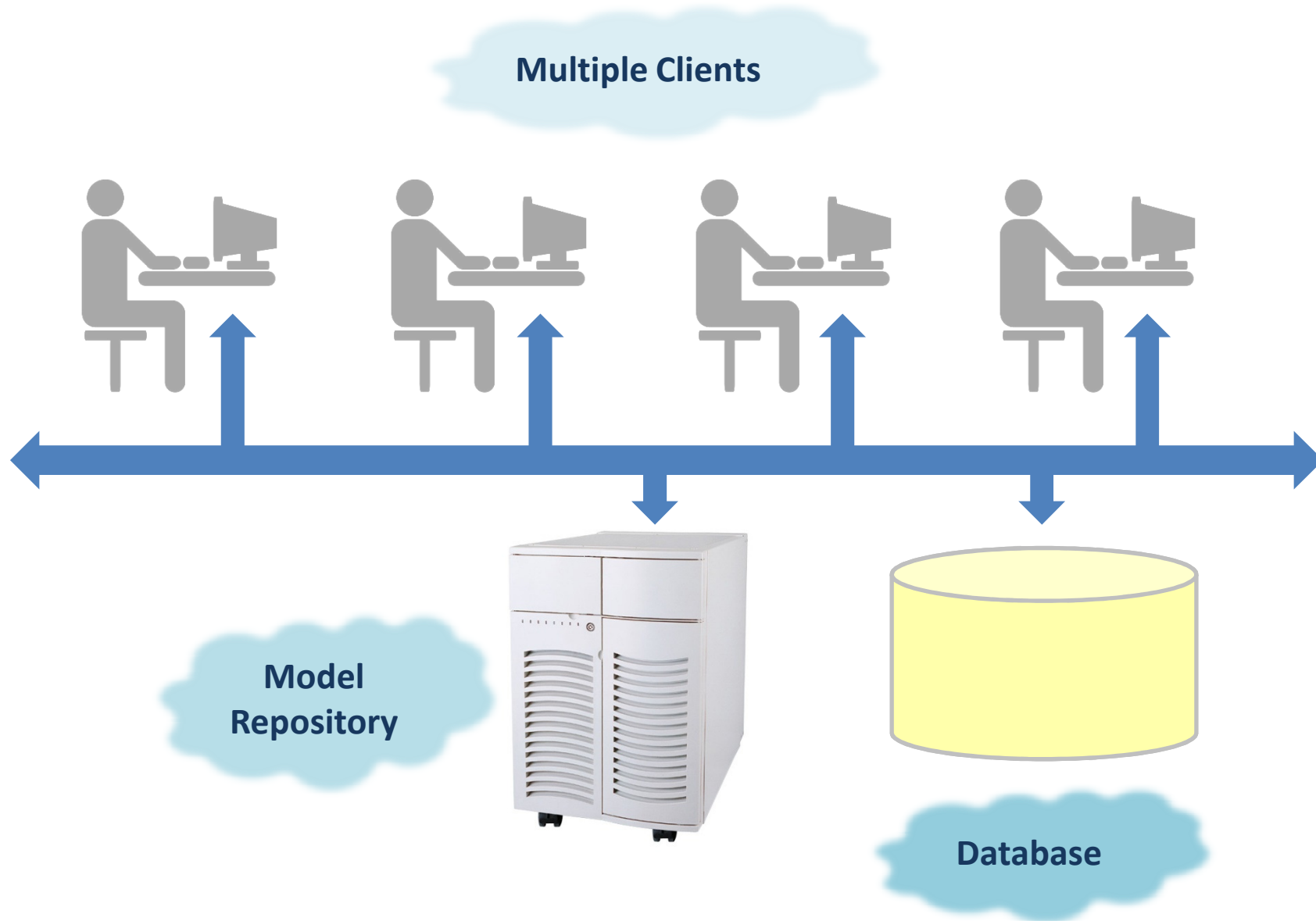






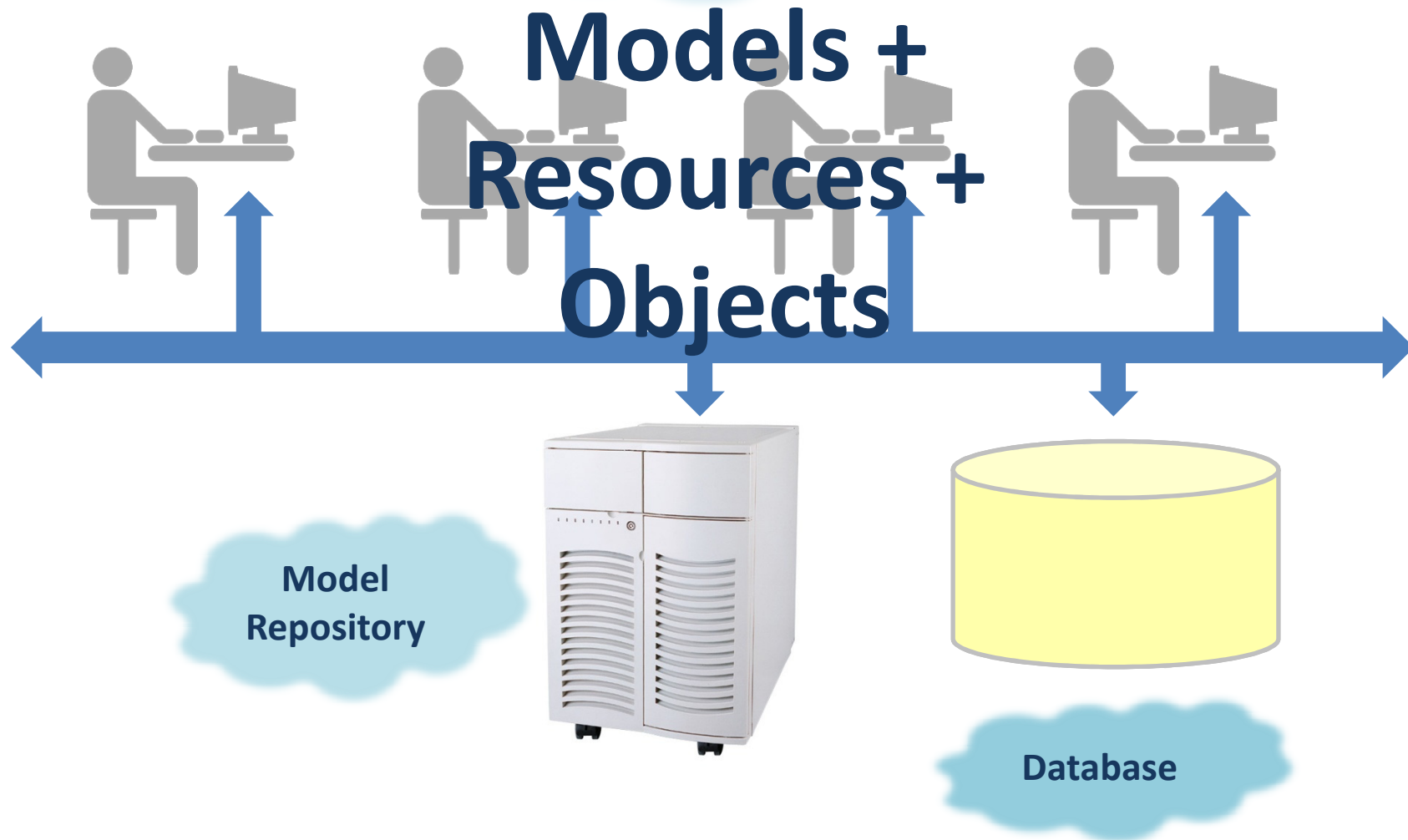






# Model Repository =

Multiple Clients



# With CDO you can...

## § **Maintain a central model repository**

- § Store models in arbitrary backends
- § Cache models in server memory

## § **Connect from multiple clients**

- § Manage sessions
- § Manage views and transactions

## § **Work with a distributed shared model**

- § Distribute change notifications
- § Present hot data

# CDO supports...

## § **Multiple sessions on a single client**

- § TCP connections

- § HTTP connections

- § JVM connections

## § **Pluggable connection security**

- § Challenge/Response negotiation

## § **Pluggable fail-over strategies**

# CDO supports...

## § **Multiple views per session**

- § Read-only views
- § Transactions and save points
- § Audit views (time machine)

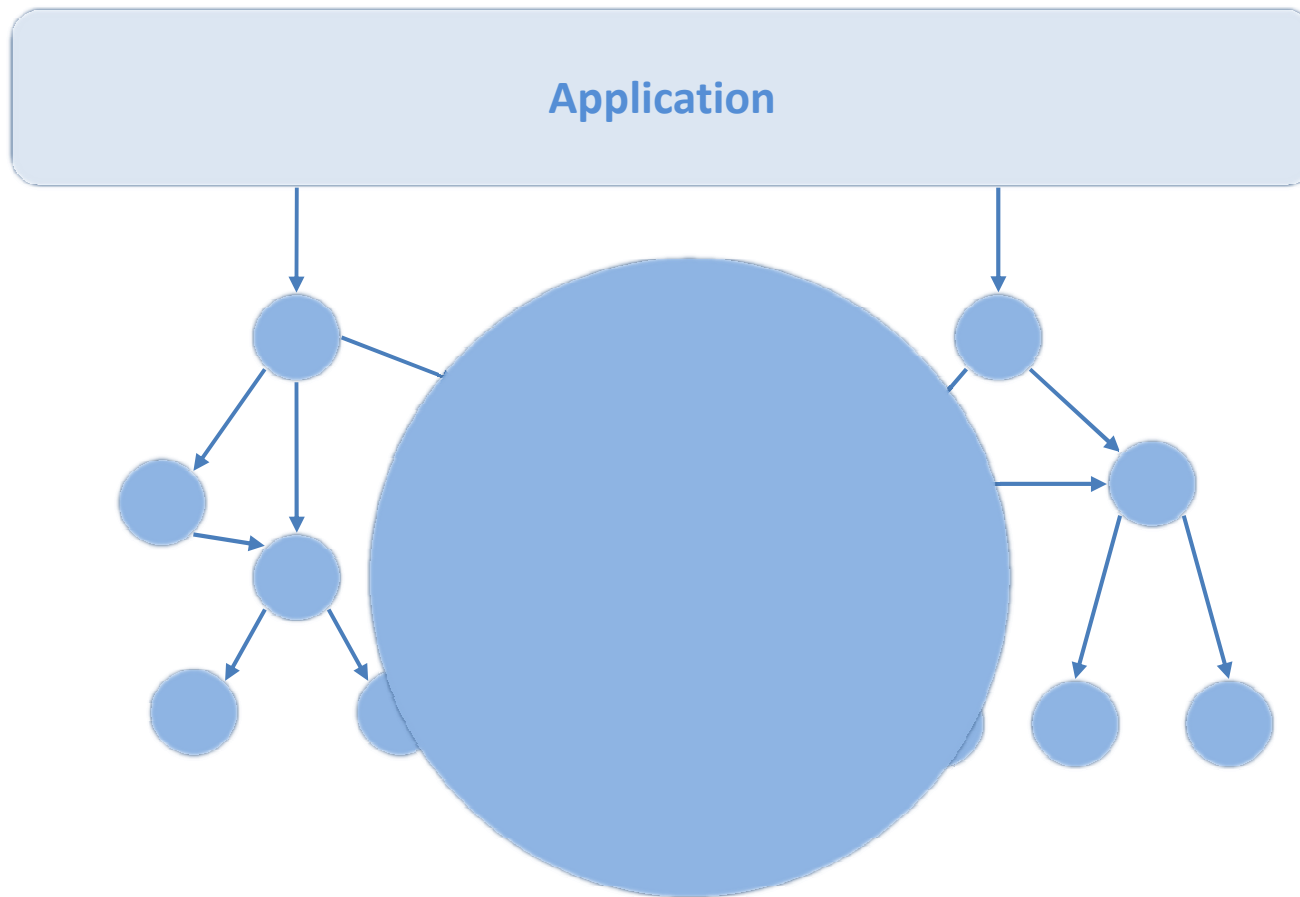
## § **Local sharing across all views**

## § **Potentially huge object graphs**


- § Transparent loading/unloading on demand
- § Partial collections
- § Intelligent prefetching



**How does all this work?**







**Book**  
State  
TID  
Pages  
Authors

# Book

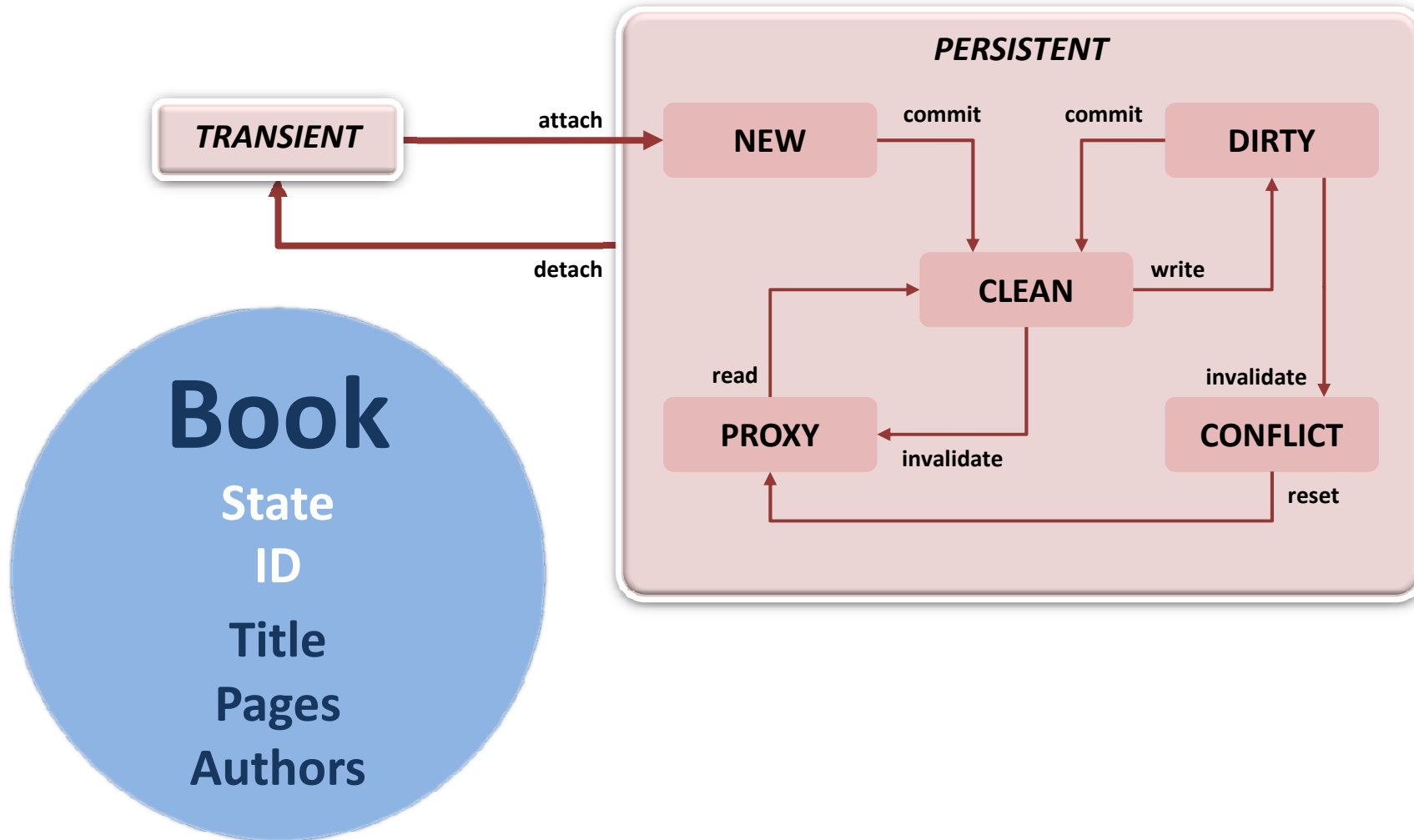
State

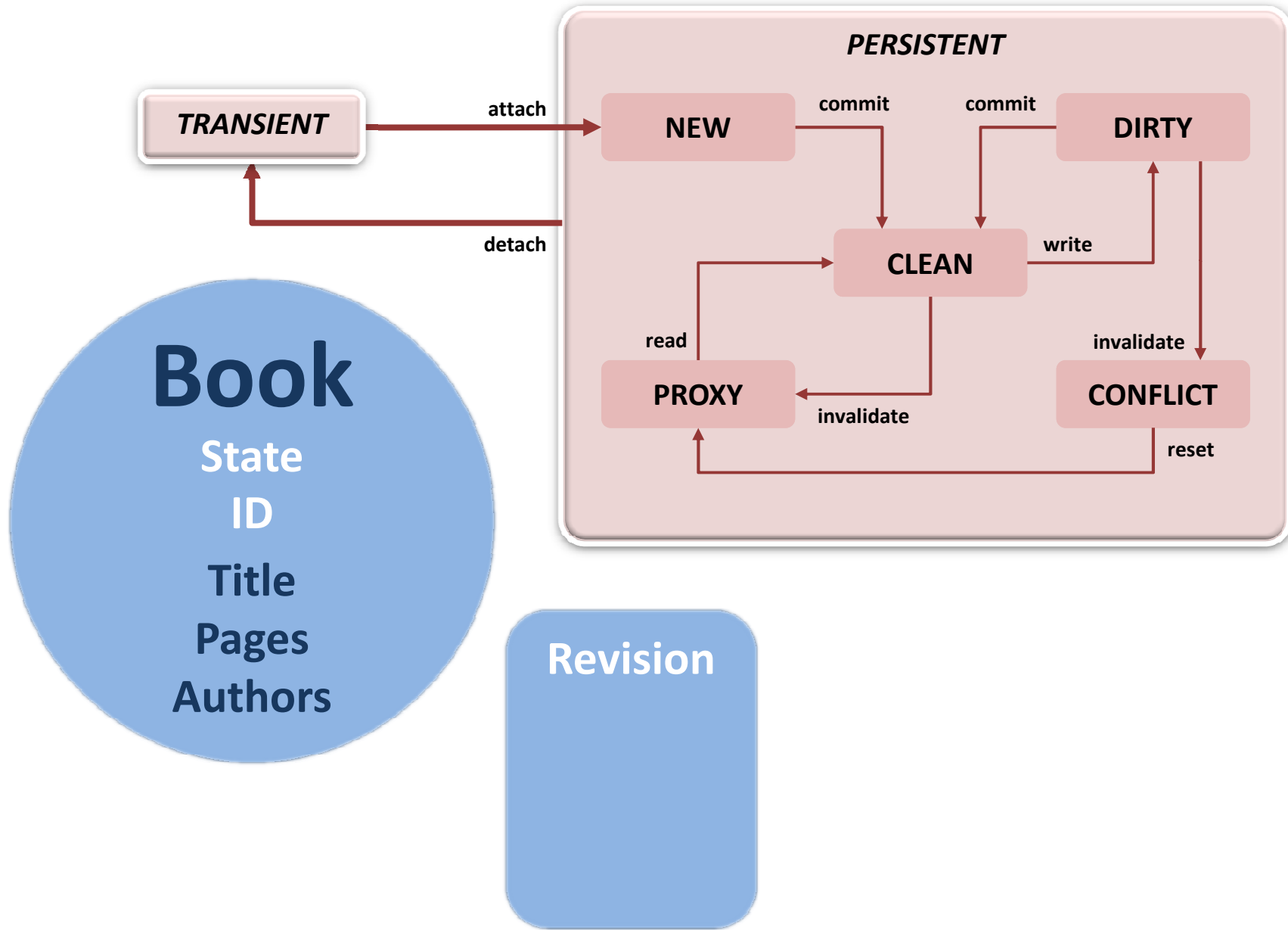
ID

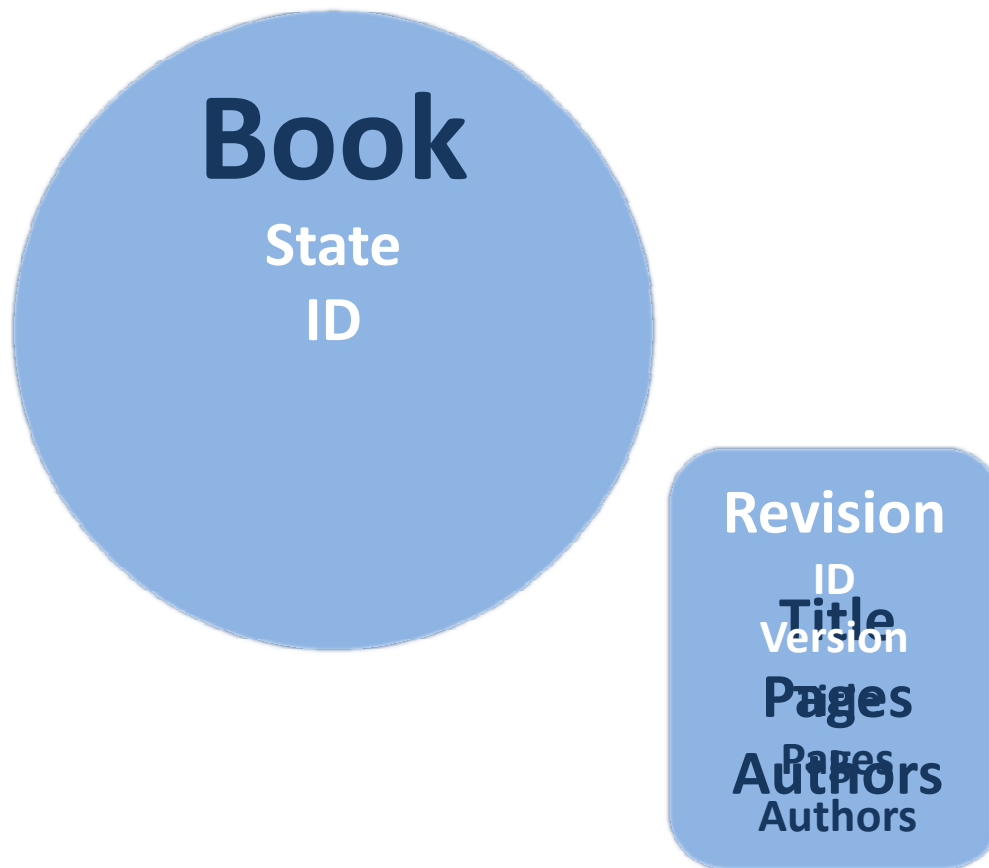
Title

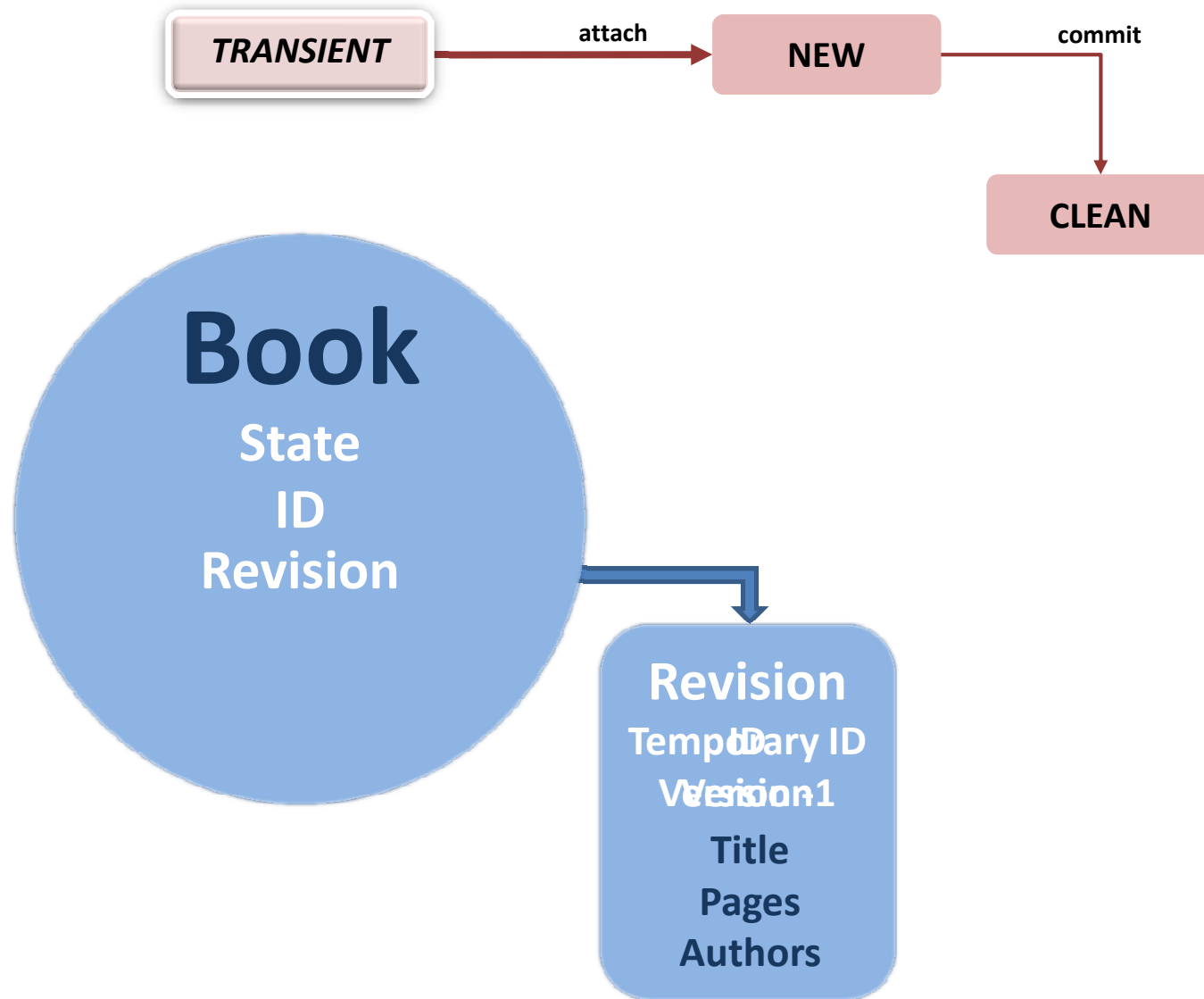
Pages

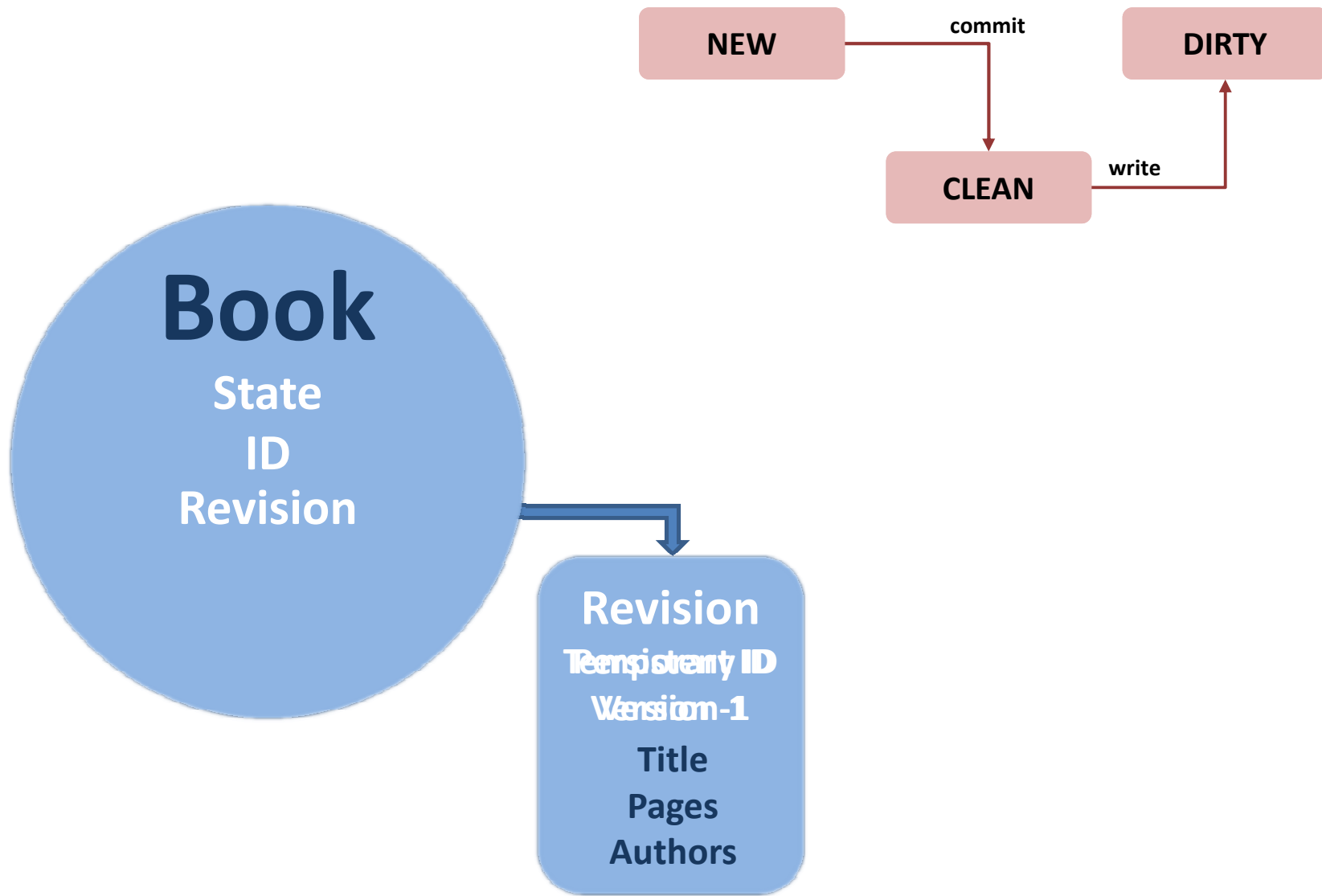
Authors

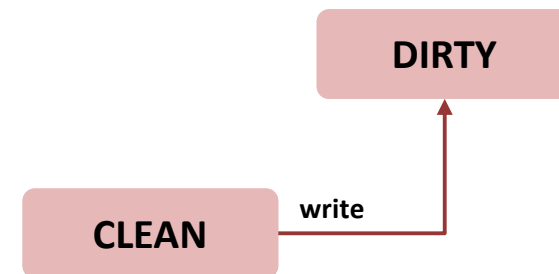
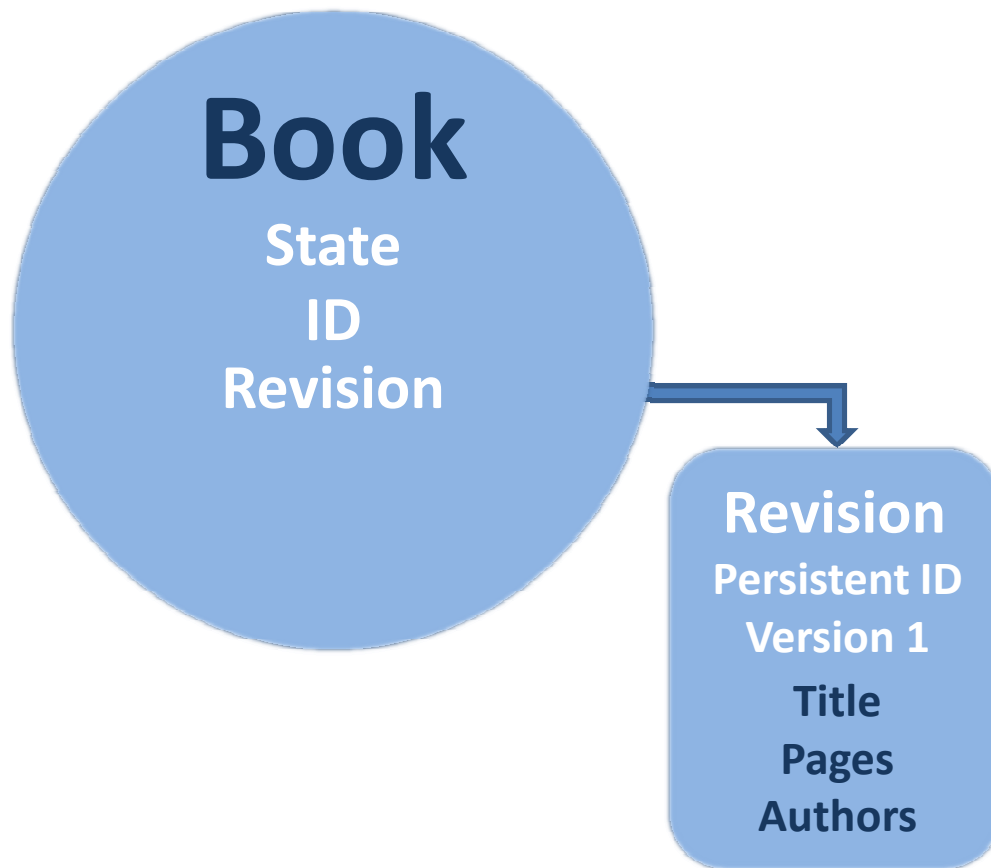




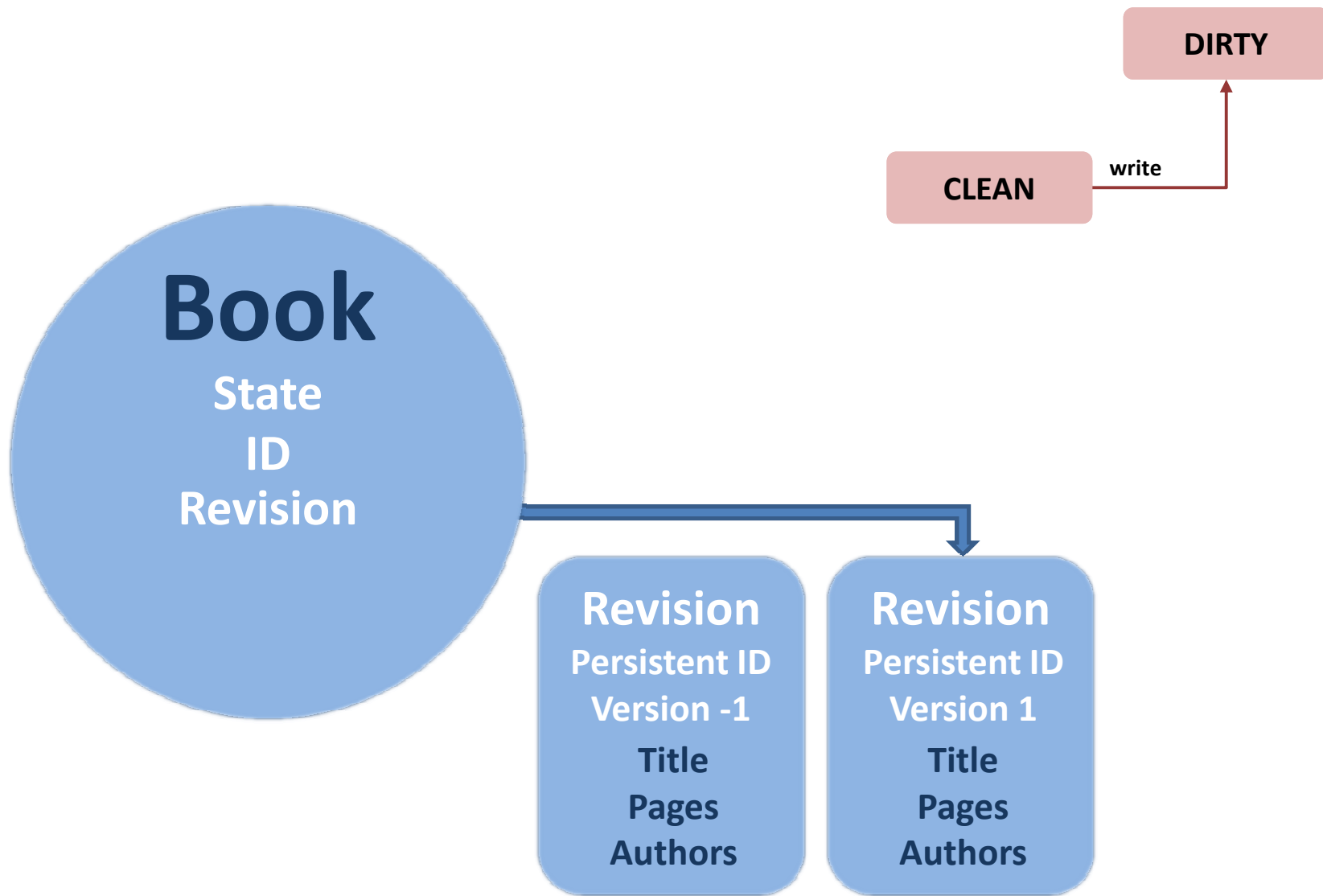


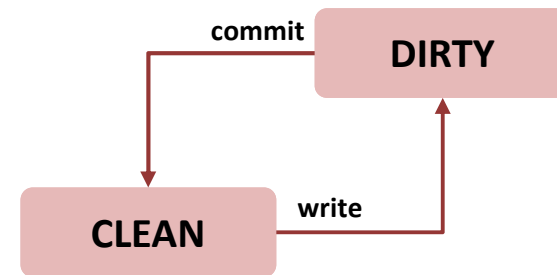
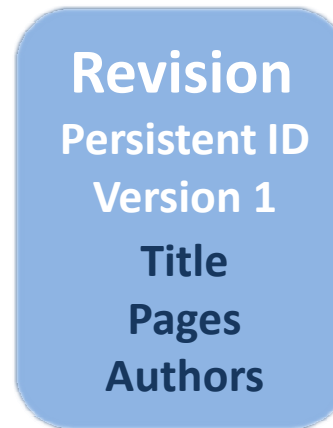
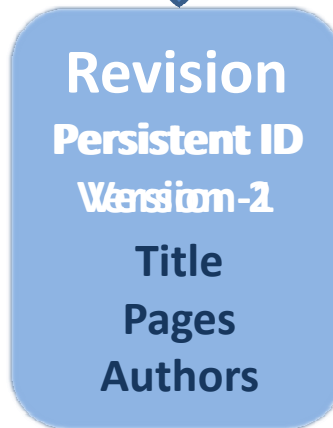
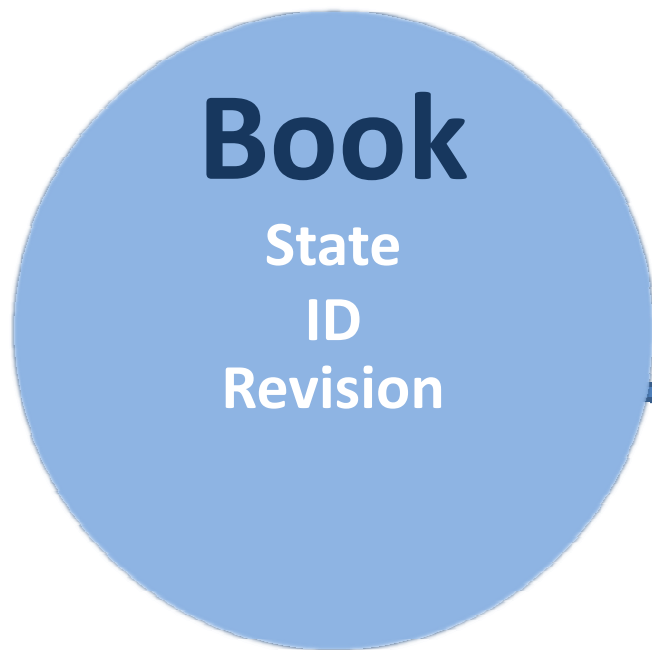


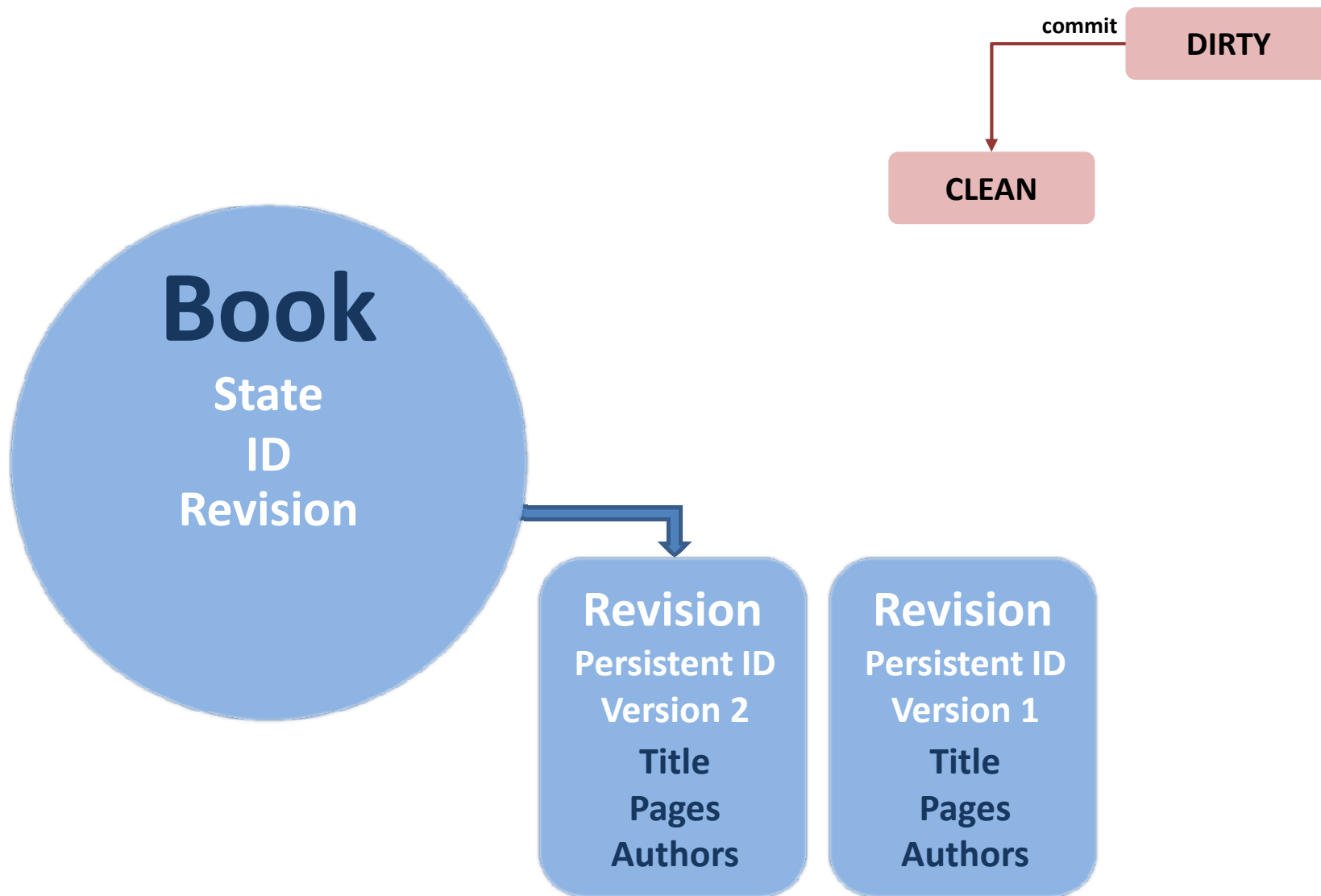


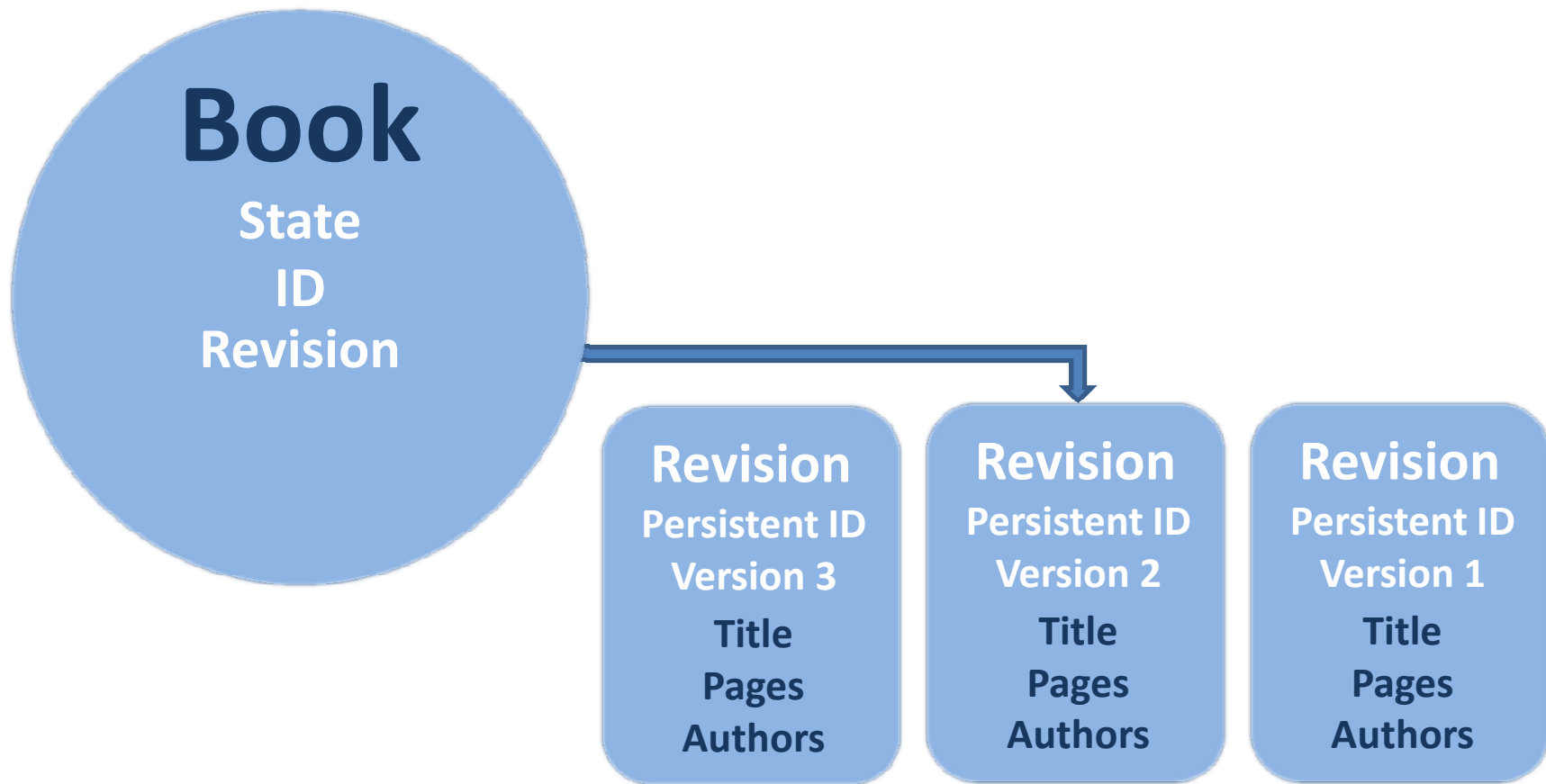




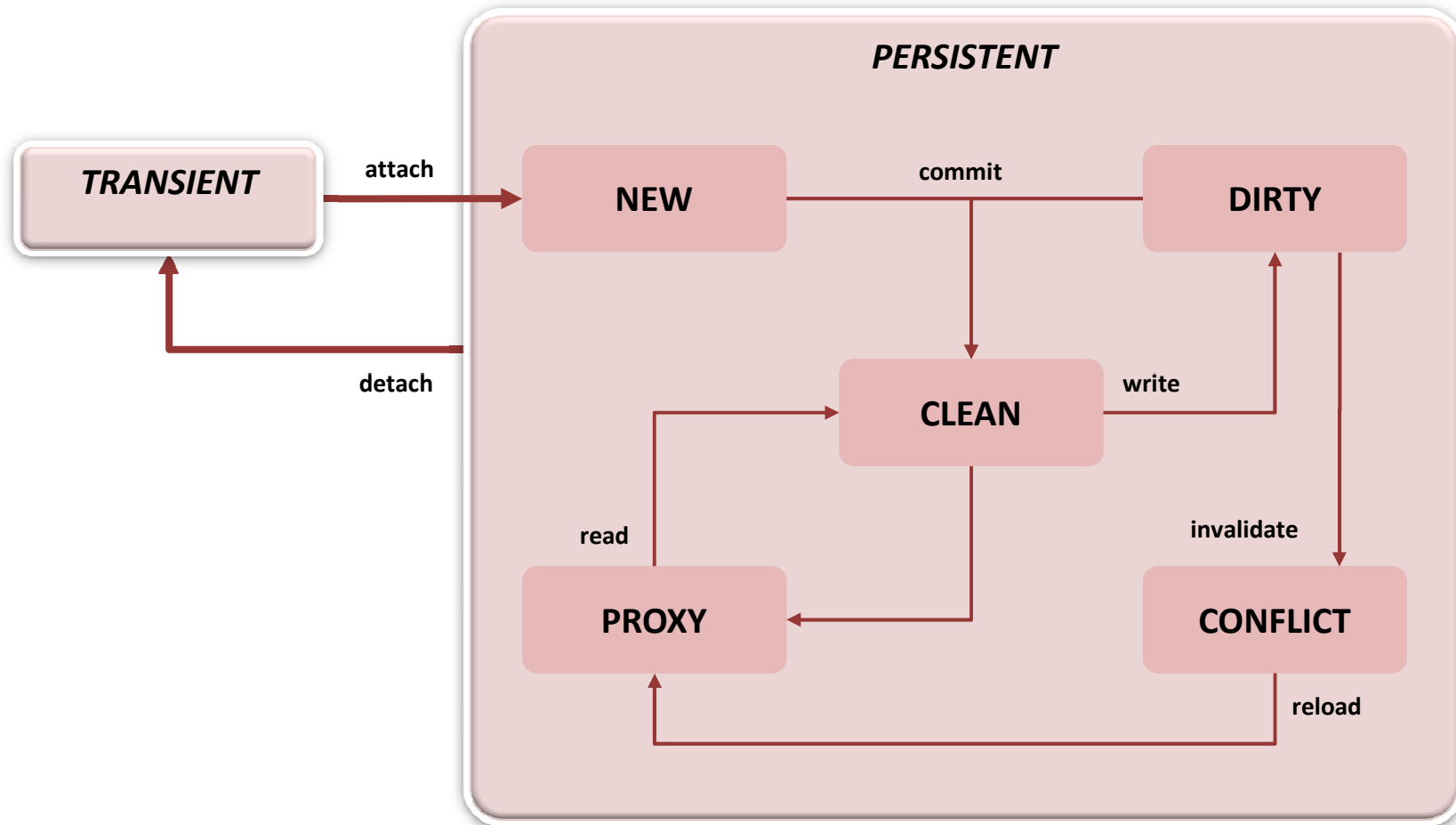






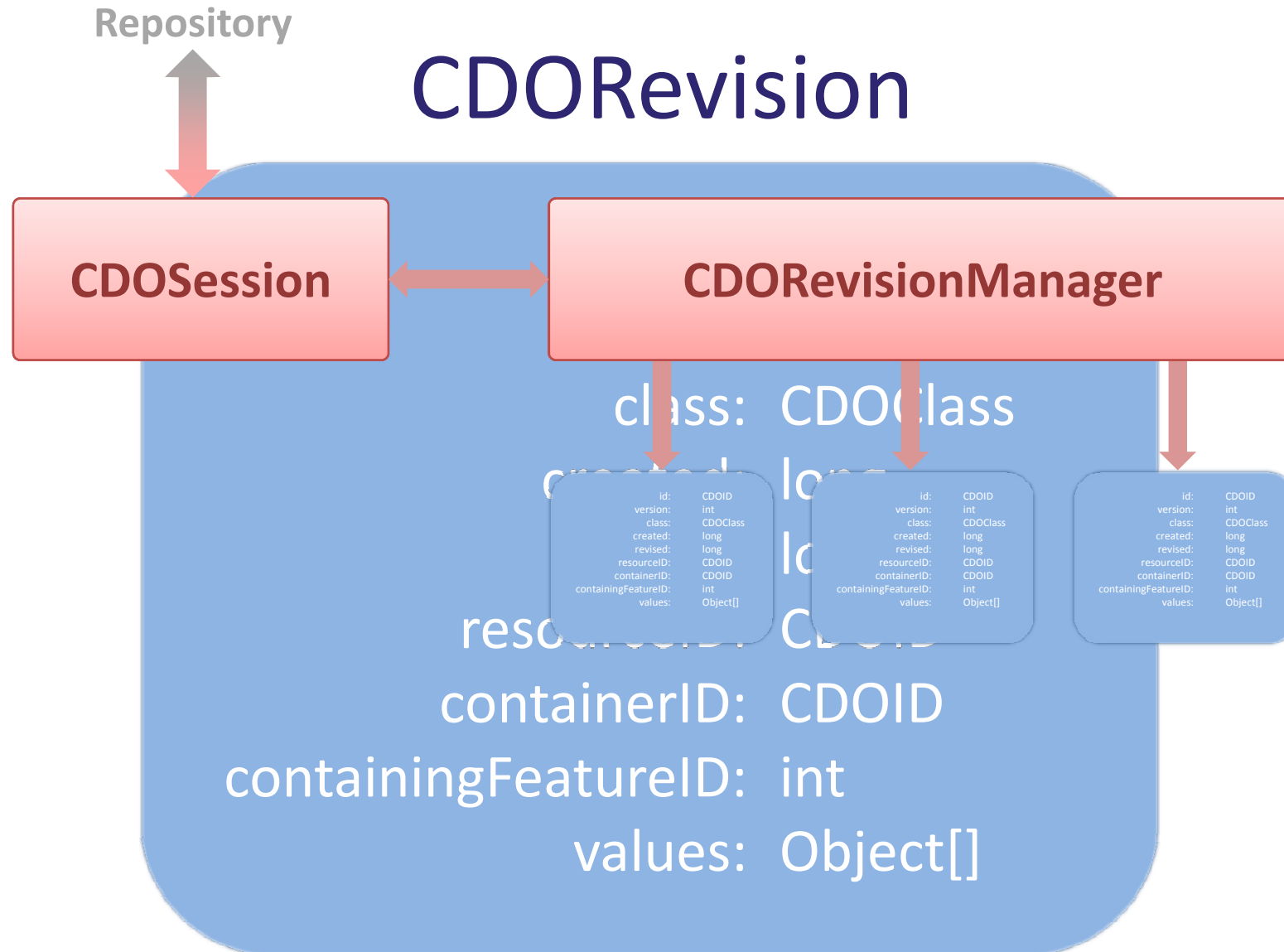


# CDOStateMachine



# CDORevision

```
id: CDOID
version: int
class: CDOClass
created: long
revised: long
resourceID: CDOID
containerID: CDOID
containingFeatureID: int
values: Object[]
```

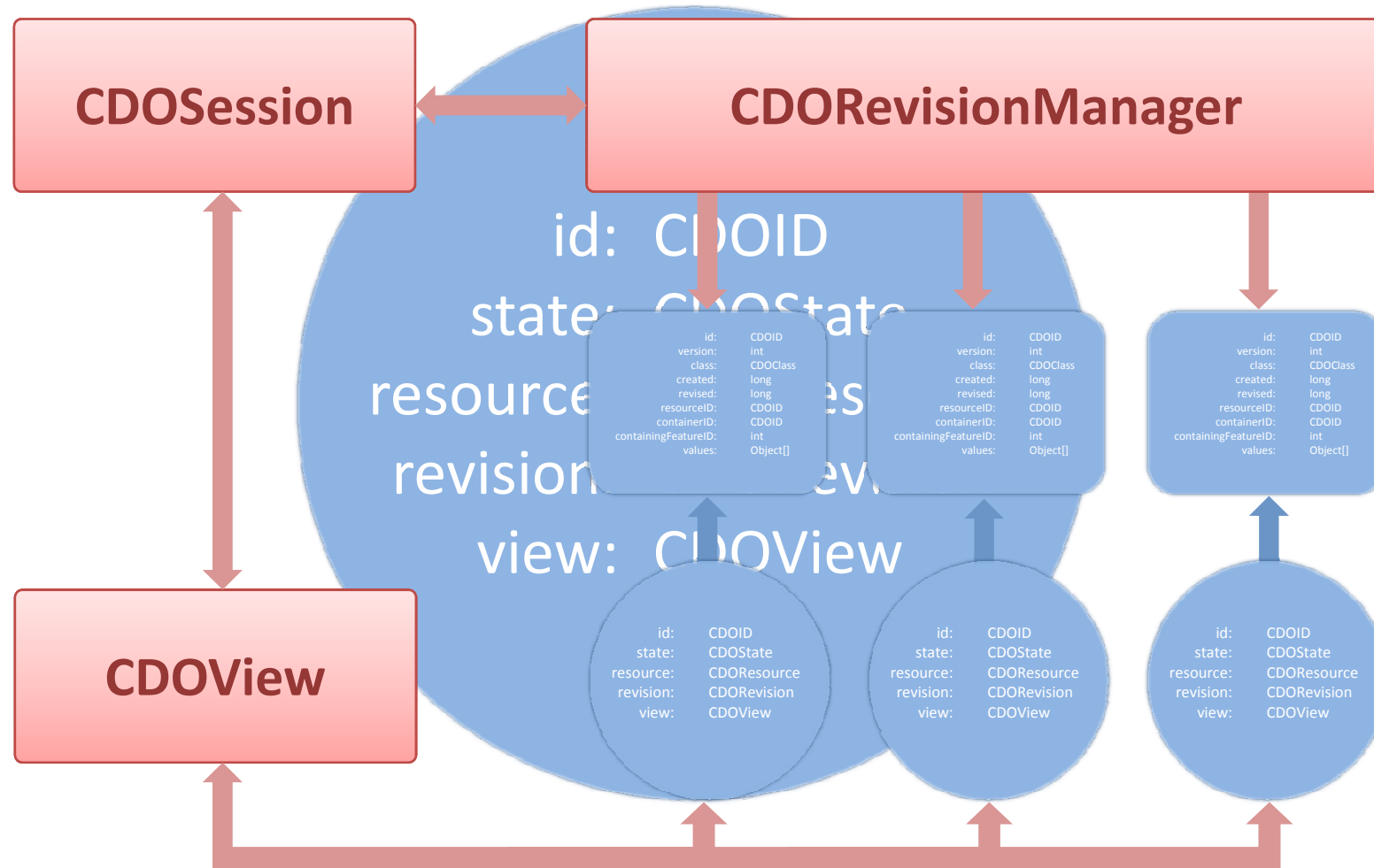


# CDOObject

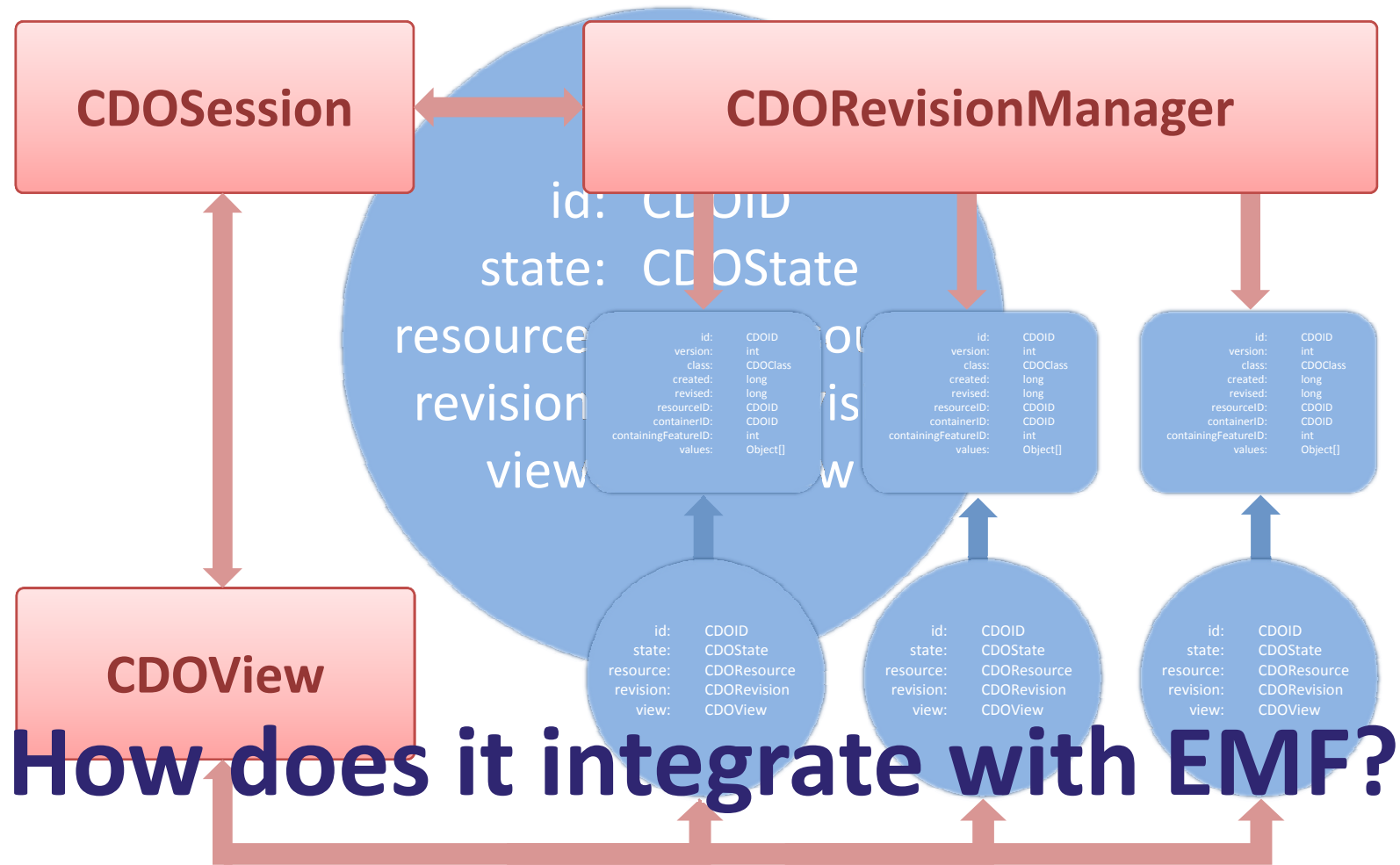
id: CDOID  
state: CDOState  
resource: CDOResource  
revision: CDORevision  
view: CDOView



# CDOObject

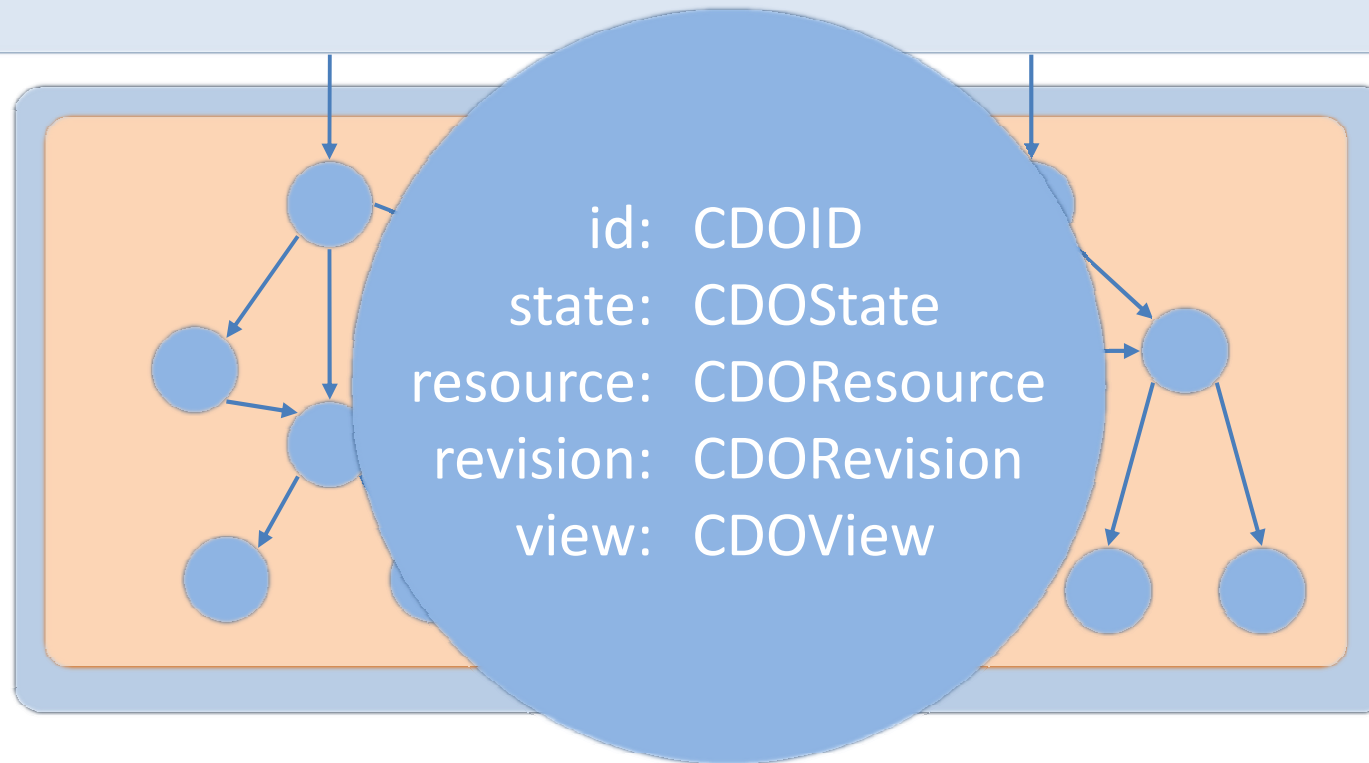


# CDOObject

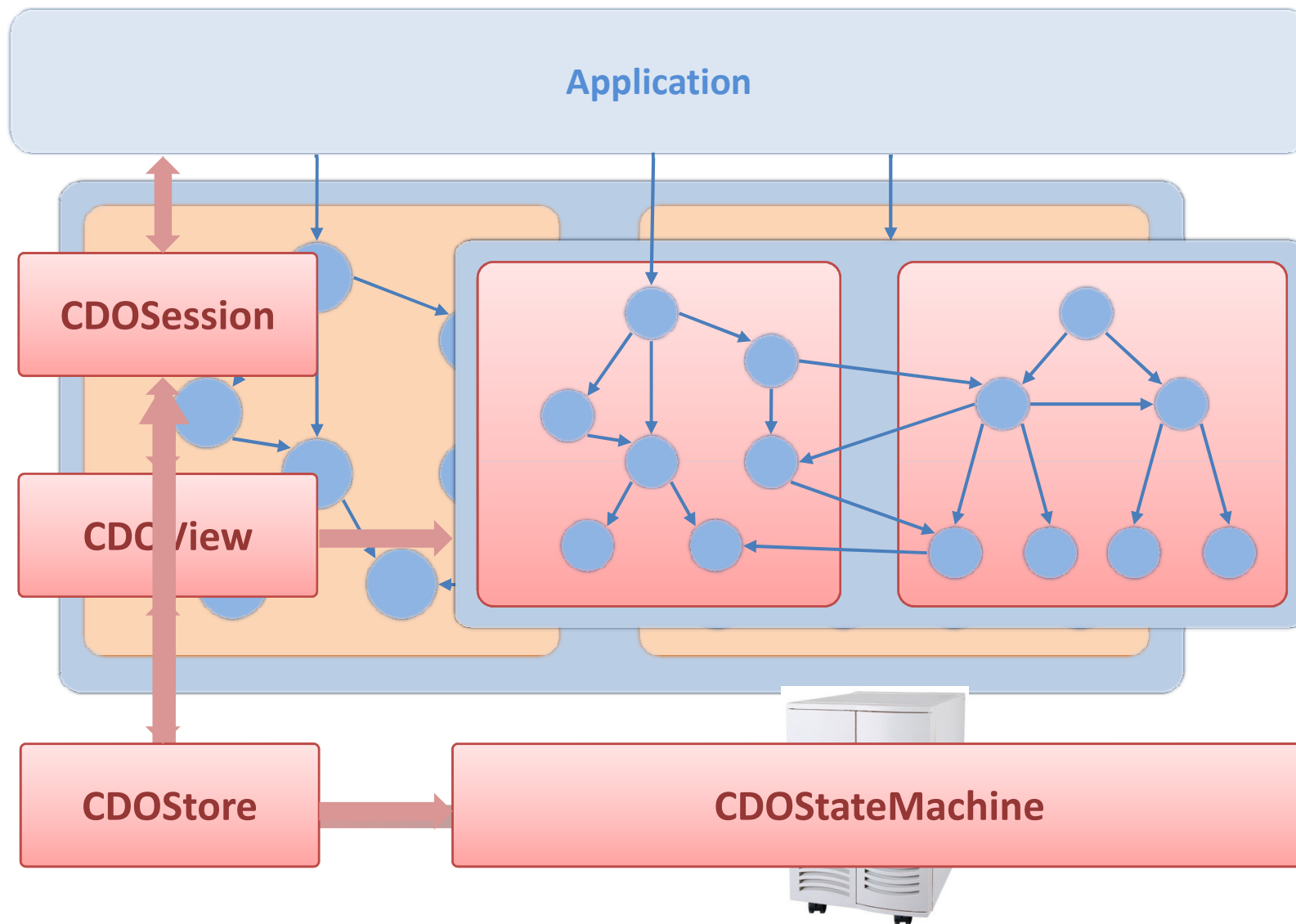


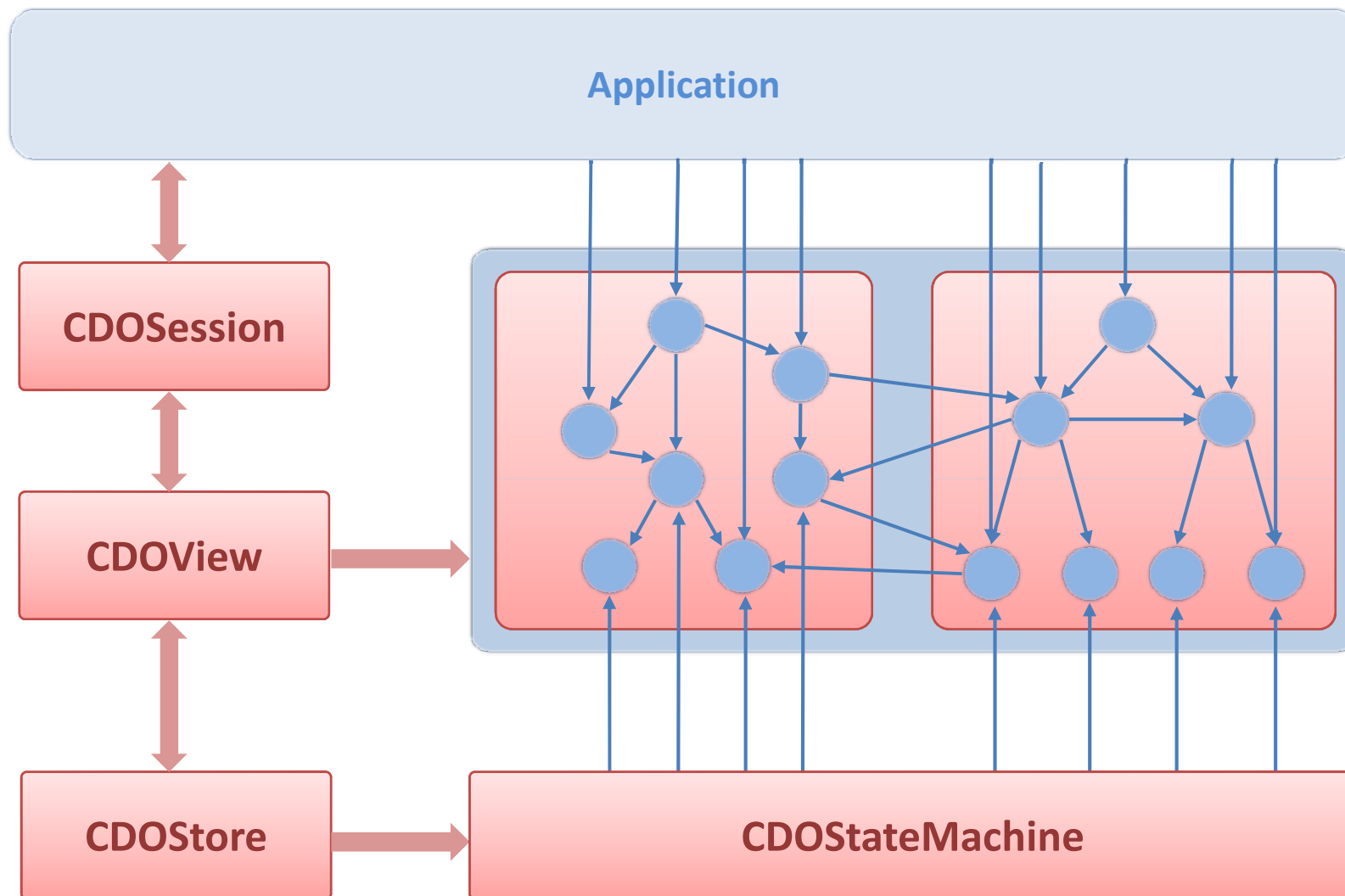
# CDOObject

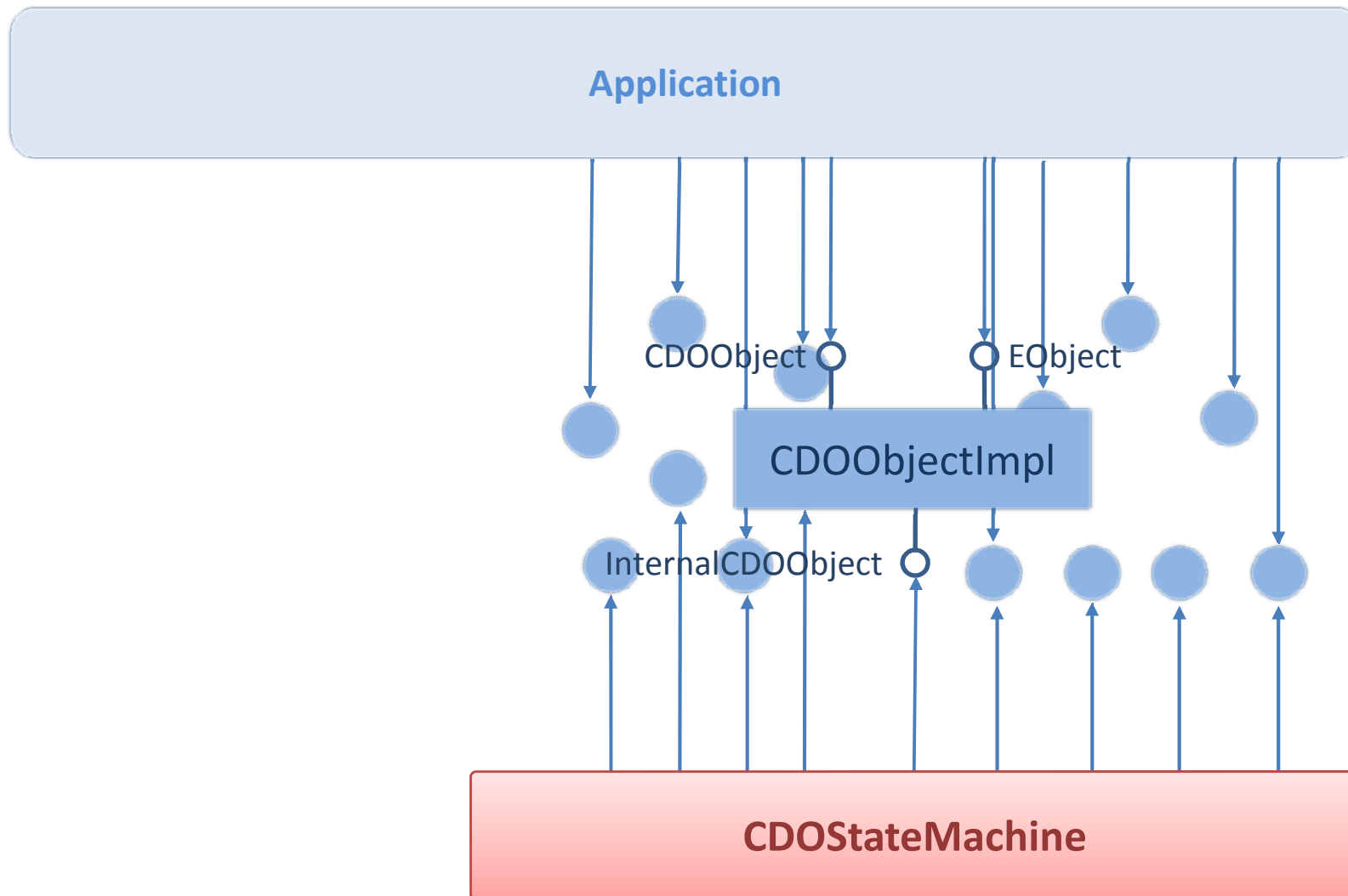
Application



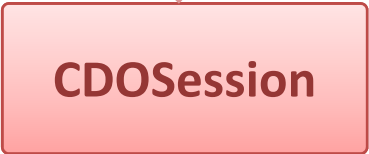
## How does it integrate with EMF?



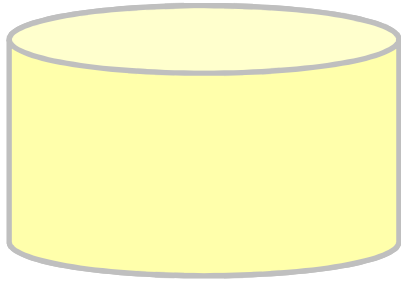




Model Type		Native		Legacy		Meta
		Dynamic	Generated	Unwoven	Woven	
Development Artifacts	Ecore	Unaffected				N/A
	Genmodel	N/A	Slightly modified	Unaffected		
Instance Interface		CDOObject		EObject		EModelObject
State Machine Interface				CDOAdapter	CDOCallback	CDOMeta
Location of Internal Values	class	DynamicCDOObject	Java Byte Code			
	store	CDOObject		N/A		
	view			EObject		CDOMeta
	id					CDOSession
	state			CDOAdapter	CDOCallback	N/A
	revision			EObject		
	resource			EObject and CDORevision		
TRANSIENT	EObject					
Location of Model Values per CDOState	NEW	CDORevision		EObject and CDORevision		N/A
	DIRTY			EObject and CDORevision		
	CLEAN			EObject and CDORevision		
				EObject and CDORevision		EModelObject

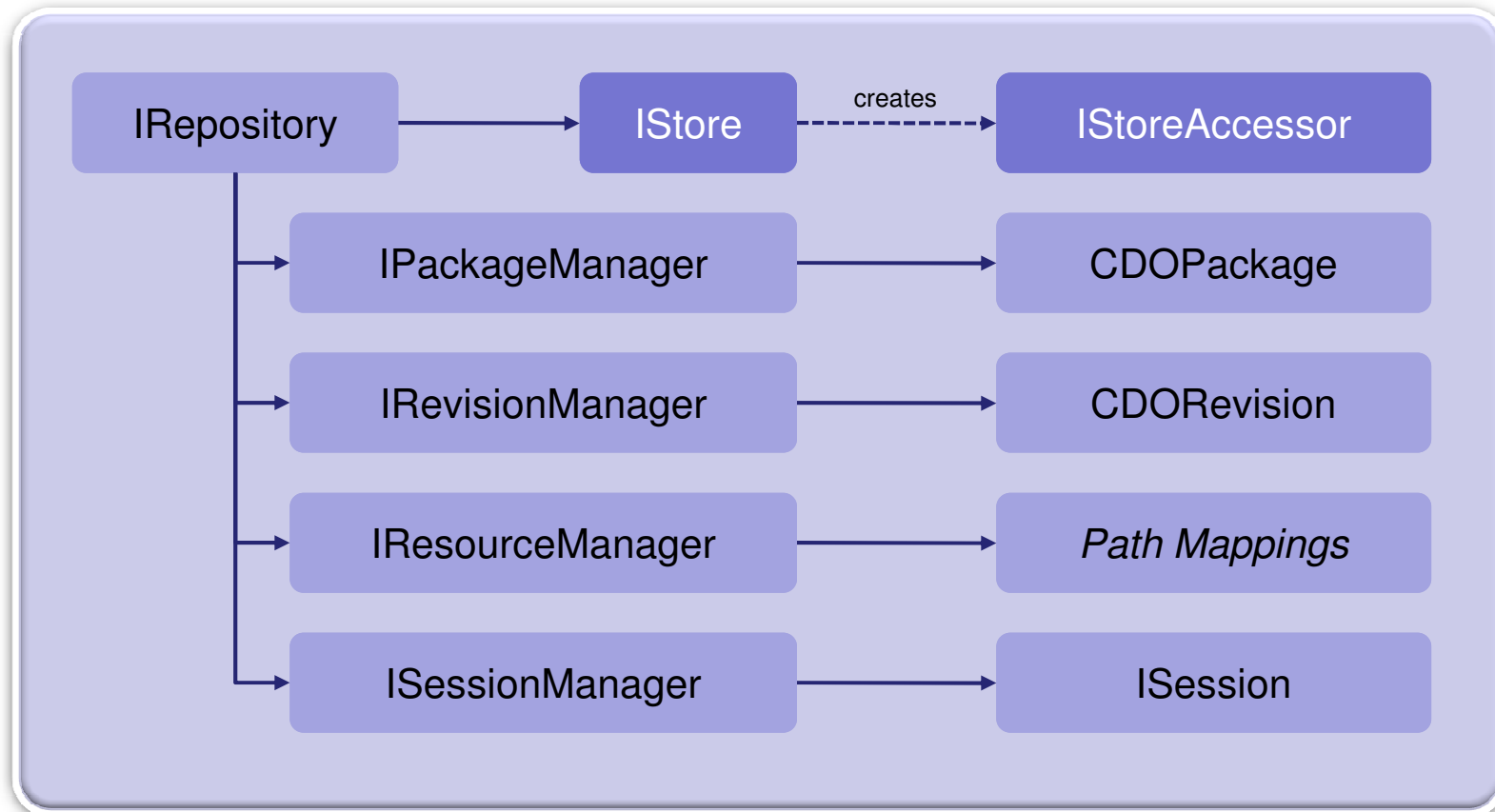


Net4j ?  
Protocol ?

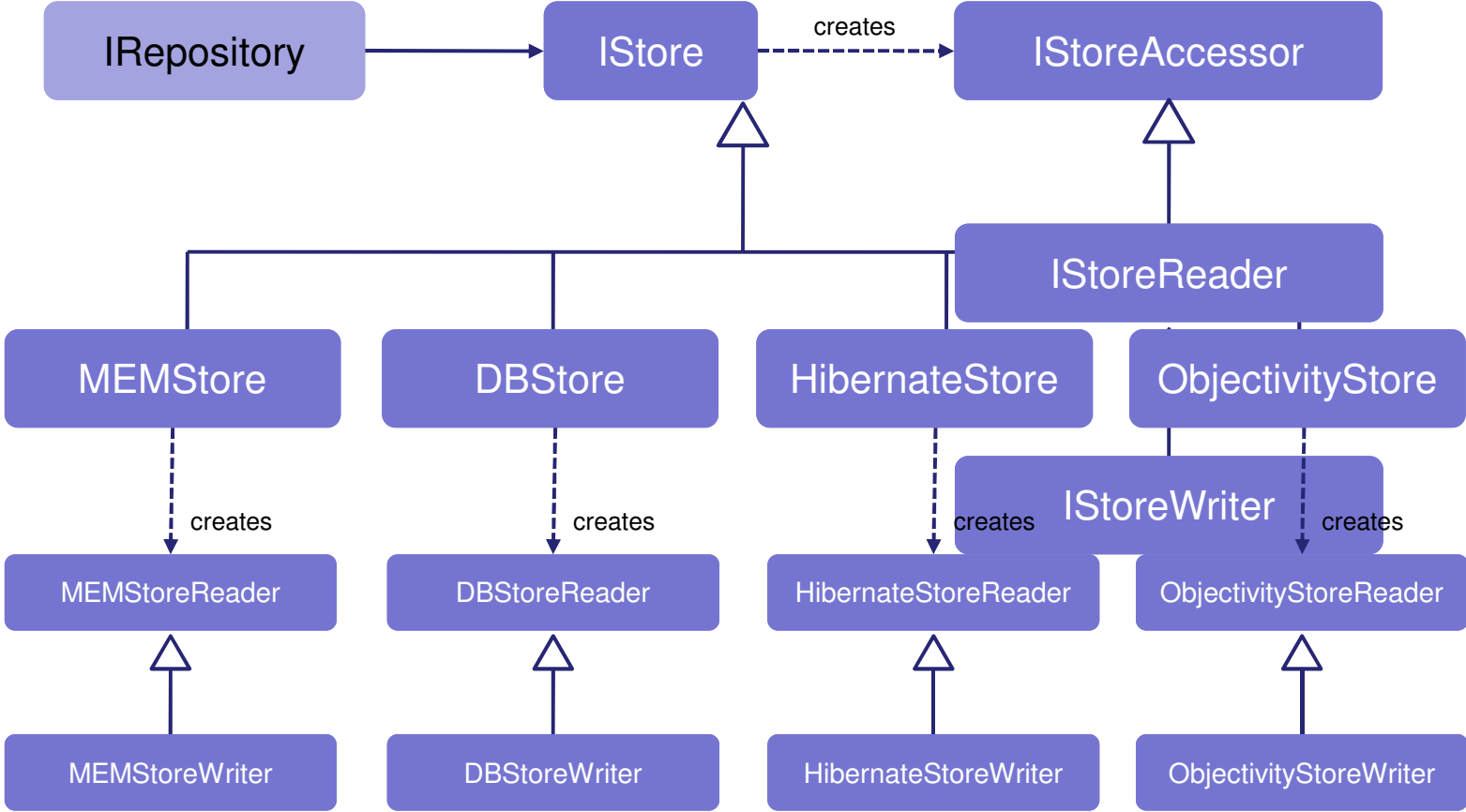




# CDO Server-Side



# CDO Server-Side



# How to continue the tutorial?

## § **Questions regarding the slides**

§ Switch desktop sharing to Eike

## § **Live Demonstration**

§ CDO server application

§ CDO user interface

## § **Open discussion + code walk-through**