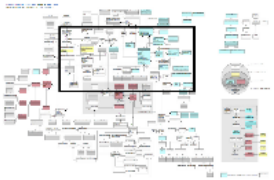
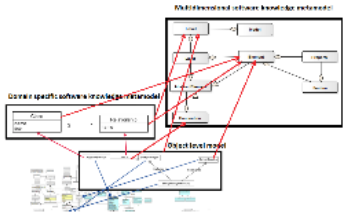
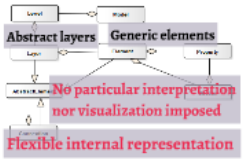




The knowledge is in relationships:
an interrelating model is necessary



Common 2D visualization
3D layered visualization
http://uml3d.herokuapp.com/

InterSKnow: an experimental modeling tool
for multidimensional software knowledge

Evaluation performed on a model of
a web based insurance system

InterSKnow outperforms
Enterprise Architect
in interrelating
heterogeneous software artifacts

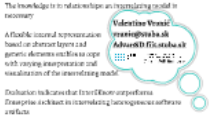
Tasks assessed by:

- Number of context switches
- Number of files opened
- Time

Two perspectives:

1. Efficiency of searching for software knowledge
2. Comprehension of software knowledge

Ability to answer selected questions about the model



M2P @ ADBIS 2019

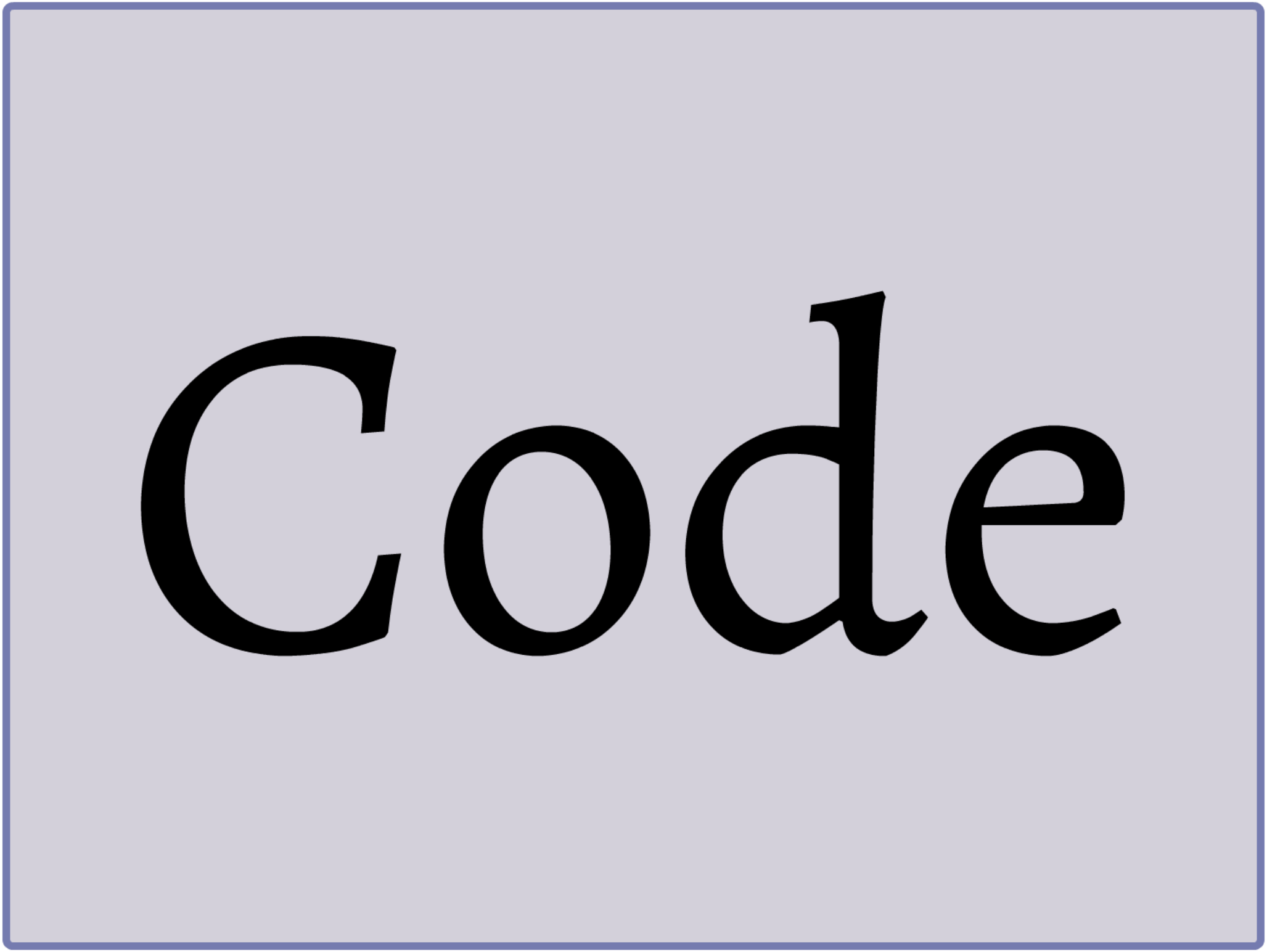
Abstract Layers and Generic Elements as a Basis for Expressing Multidimensional Software Knowledge

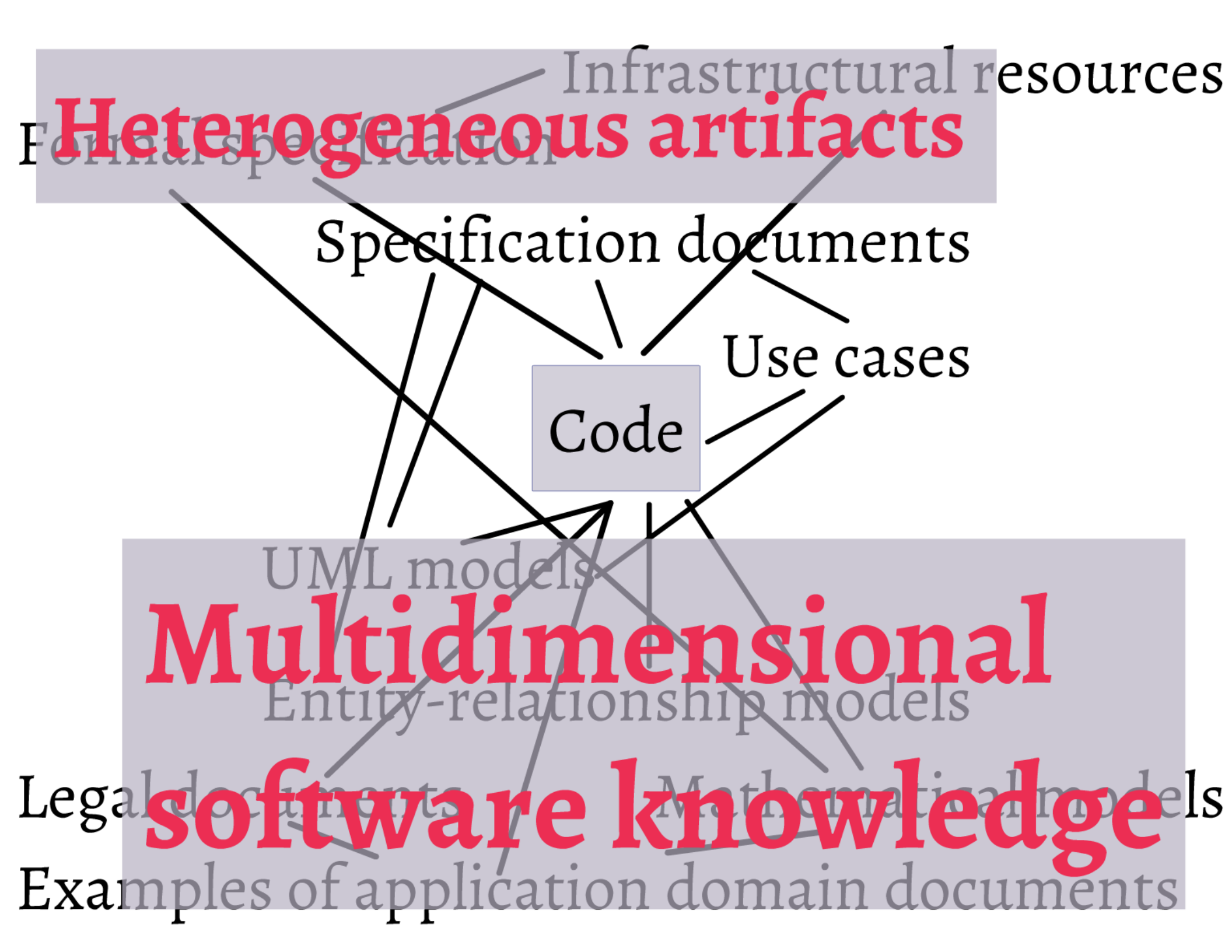
Valentino Vranić and Adam Neupauer

Institute of Informatics, Information Systems
and Software Engineering

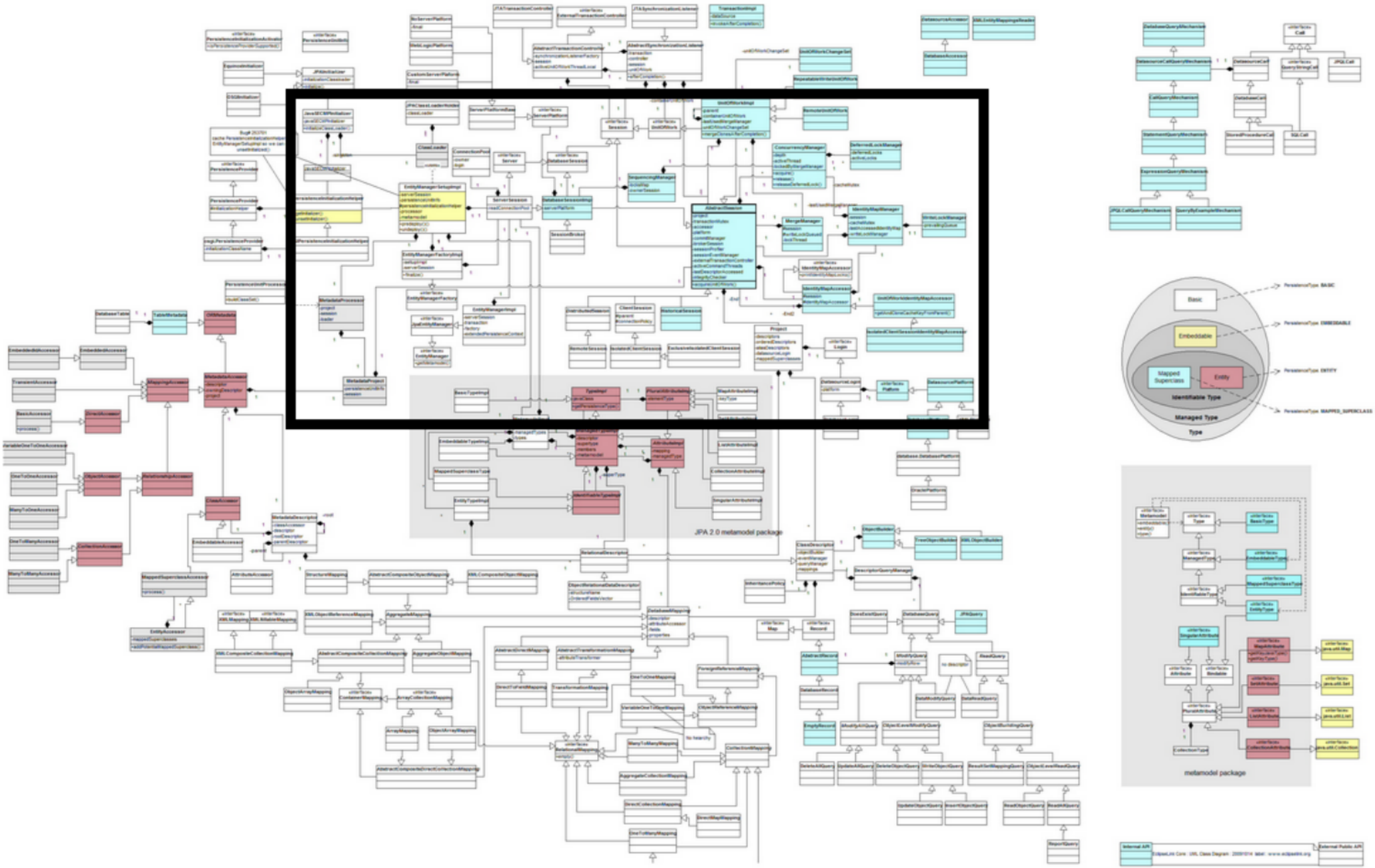
SLOVAK UNIVERSITY OF
TECHNOLOGY IN BRATISLAVA
FACULTY OF INFORMATICS
AND INFORMATION TECHNOLOGIES

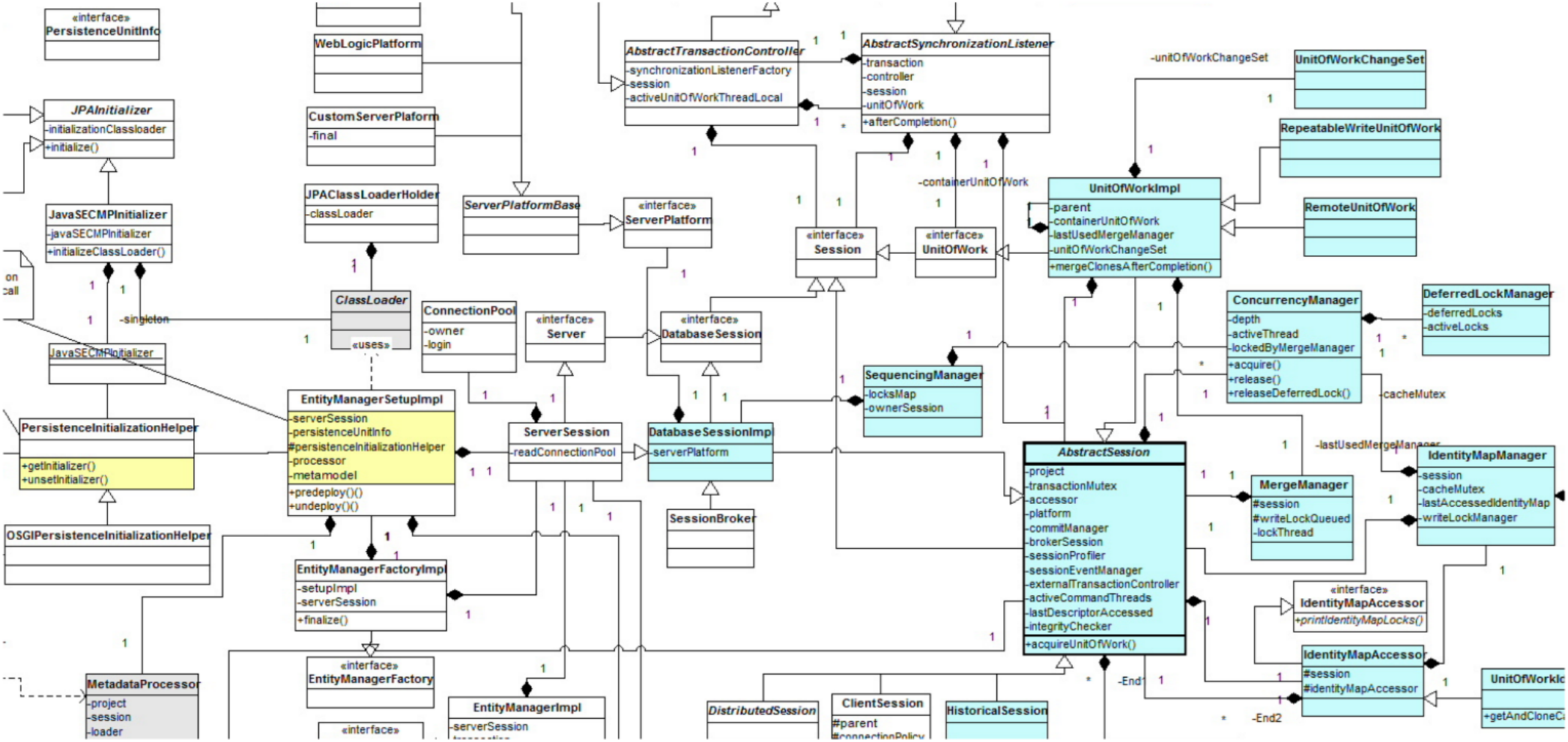
vranic@stuba.sk
fiit.sk/~vranic

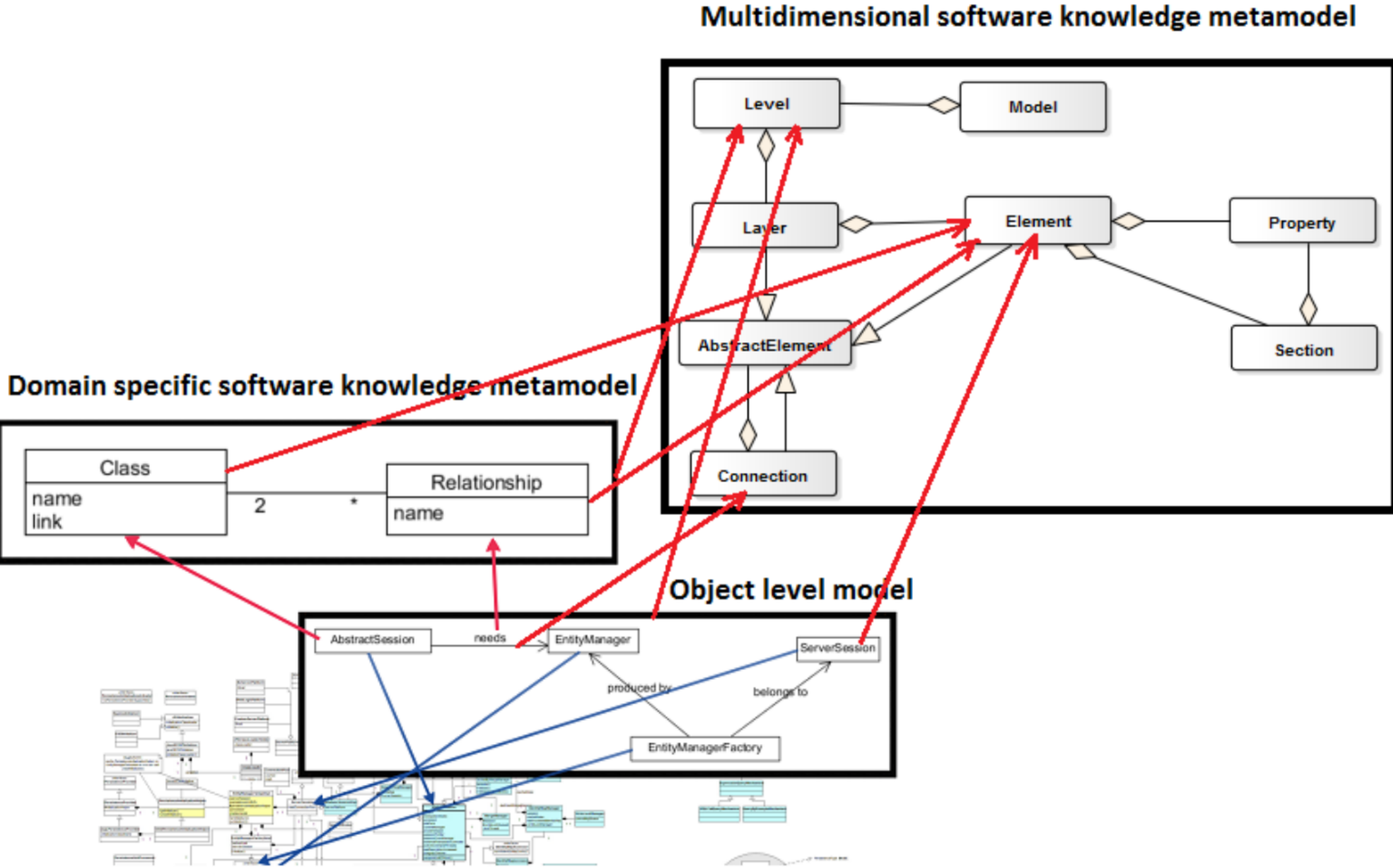

Code


Heterogeneous artifacts
Infrastructural resources
Formal specification
Specification documents
Use cases
Code
UML models
Multidimensional
Entity-relationship models
software knowledge
Legal documents
Mathematical models
Examples of application domain documents

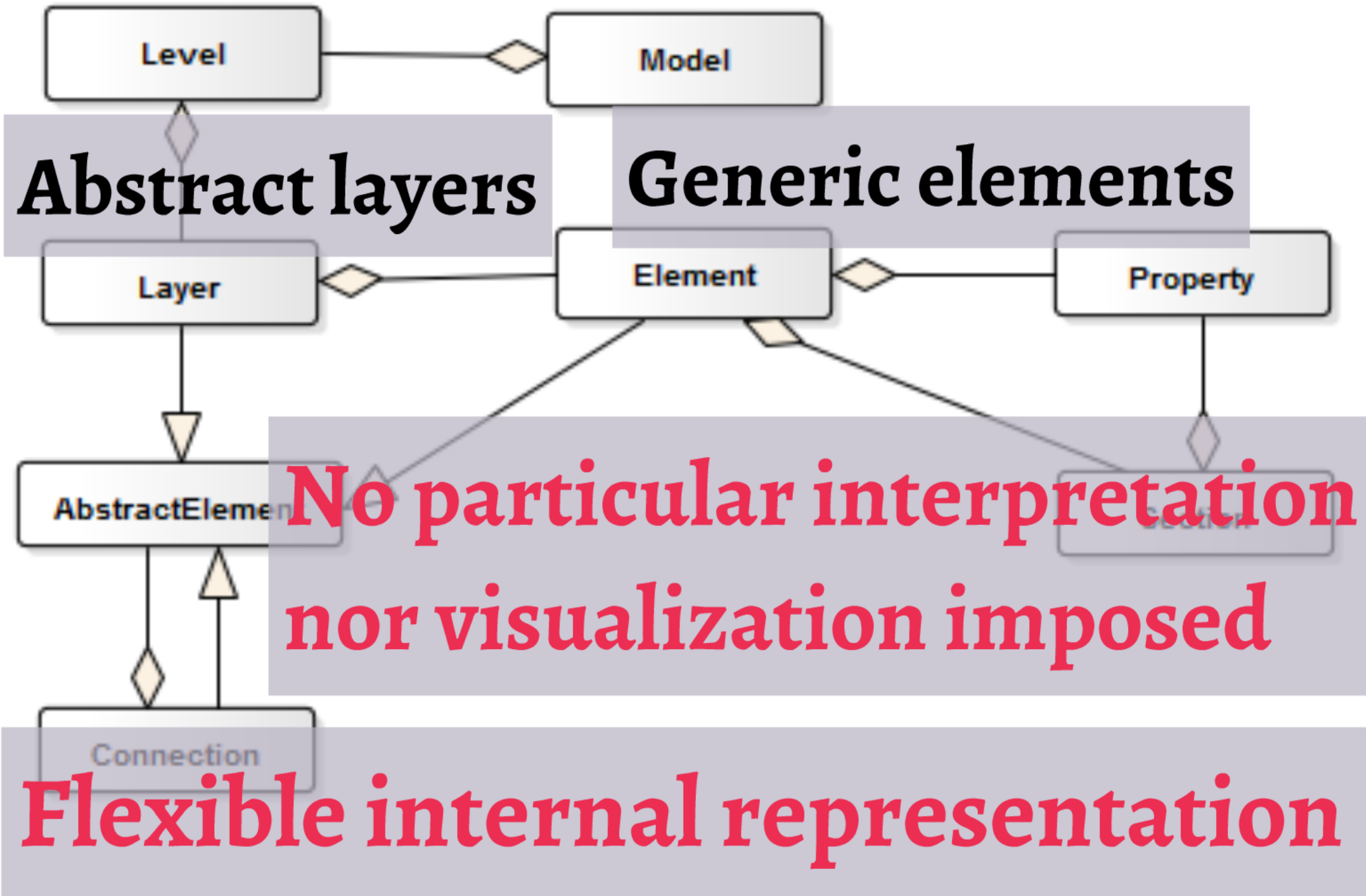
UML Source : Microsoft Visio 2000 VSD format.It hosted on the Amazon S3 cloud.It
EclipseLink build 20091014 UML Class Diagram with JPA 2.0 Metamodel API

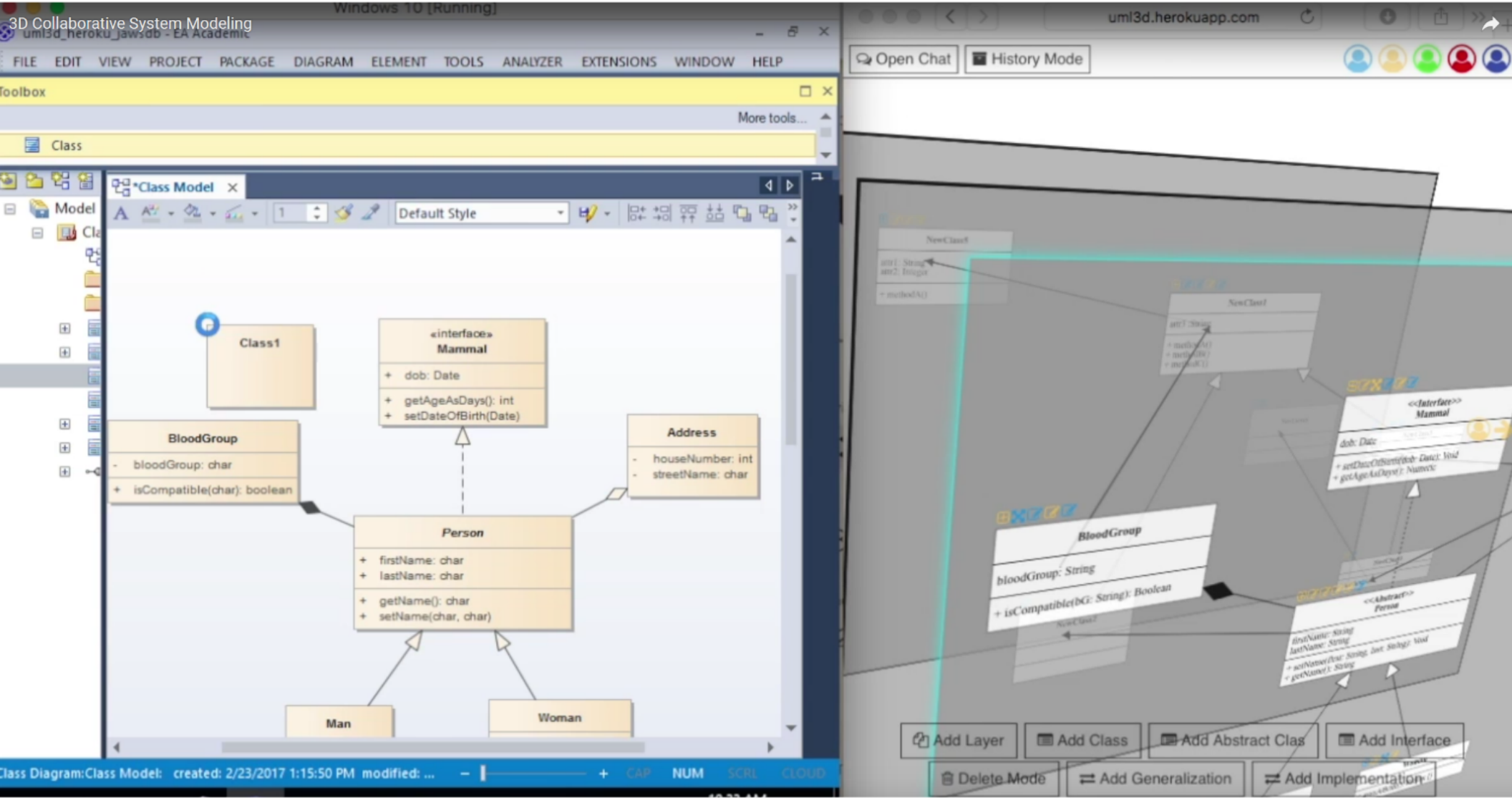



«interface»
PersistenceUnitInfo
WebLogicPlatform
AbstractTransactionController
-synchronizationListenerFactory
-session
-activeUnitOfWorkThreadLocal
AbstractSynchronizationListener
-transaction
-controller
-session
-unitOfWork
+afterCompletion()
-unitOfWorkChangeSet
UnitOfWorkChangeSet
JPAInitializer
-initializationClassloader
+initialize()
CustomServerPlaform
-final
RepeatableWriteUnitOfWork
JPAClassLoaderHolder
-classLoader
ServerPlatformBase
«interface»
ServerPlatform
-containerUnitOfWork
UnitOfWorkImpl
-parent
-containerUnitOfWork
-lastUsedMergeManager
-unitOfWorkChangeSet
+mergeClonesAfterCompletion()
RemoteUnitOfWork
JavaSECMPInitializer
-javaSECMPInitializer
+initializeClassLoader()
«interface»
Session
«interface»
UnitOfWork
ClassLoader
«uses»
-singleton
ConnectionPool
-owner
-login
«interface»
Server
«interface»
DatabaseSession
ConcurrencyManager
-depth
-activeThread
-lockedByMergeManager
+acquire()
+release()
+releaseDeferredLock()
DeferredLockManager
-deferredLocks
-activeLocks
-cacheMutex
-lastUsedMergeManager
SequencingManager
-locksMap
-ownerSession
PersistenceInitializationHelper
+getInitializer()
+unsetInitializer()
EntityManagerSetupImpl
-serverSession
-persistenceUnitInfo
#persistenceInitializationHelper
-processor
-metamodel
+predeploy()()
+undeploy()()
ServerSession
-readConnectionPool
DatabaseSessionImpl
-serverPlatform
AbstractSession
-project
-transactionMutex
-accessor
-platform
-commitManager
-brokerSession
-sessionProfiler
-sessionEventManager
-externalTransactionController
-activeCommandThreads
-lastDescriptorAccessed
-integrityChecker
+acquireUnitOfWork()
MergeManager
#session
#writeLockQueued
-lockThread
IdentityMapManager
-session
-cacheMutex
-lastAccessedIdentityMap
-writeLockManager
SessionBroker
OSGIPersistenceInitializationHelper
EntityManagerFactoryImpl
-setupImpl
-serverSession
+finalize()
«interface»
IdentityMapAccessor
+printIdentityMapLocks()
«interface»
EntityManagerFactory
IdentityMapAccessor
#session
#identityMapAccessor
MetadataProcessor
-project
-session
-loader
EntityManagerImpl
-serverSession
DistributedSession
ClientSession
#parent
HistoricalSession
-End1
-End2
UnitOfWorkIc
+getAndCloneC


Multidimensional software knowledge metamodel
Level
Model
Layer
Element
Property
AbstractElement
Section
Connection
Domain specific software knowledge metamodel
Class
name
link
2
*
Relationship
name
Object level model
AbstractSession
needs
EntityManager
ServerSession
produced by
belongs to
EntityManagerFactory

The knowledge is in relationships: an interrelating model is necessary

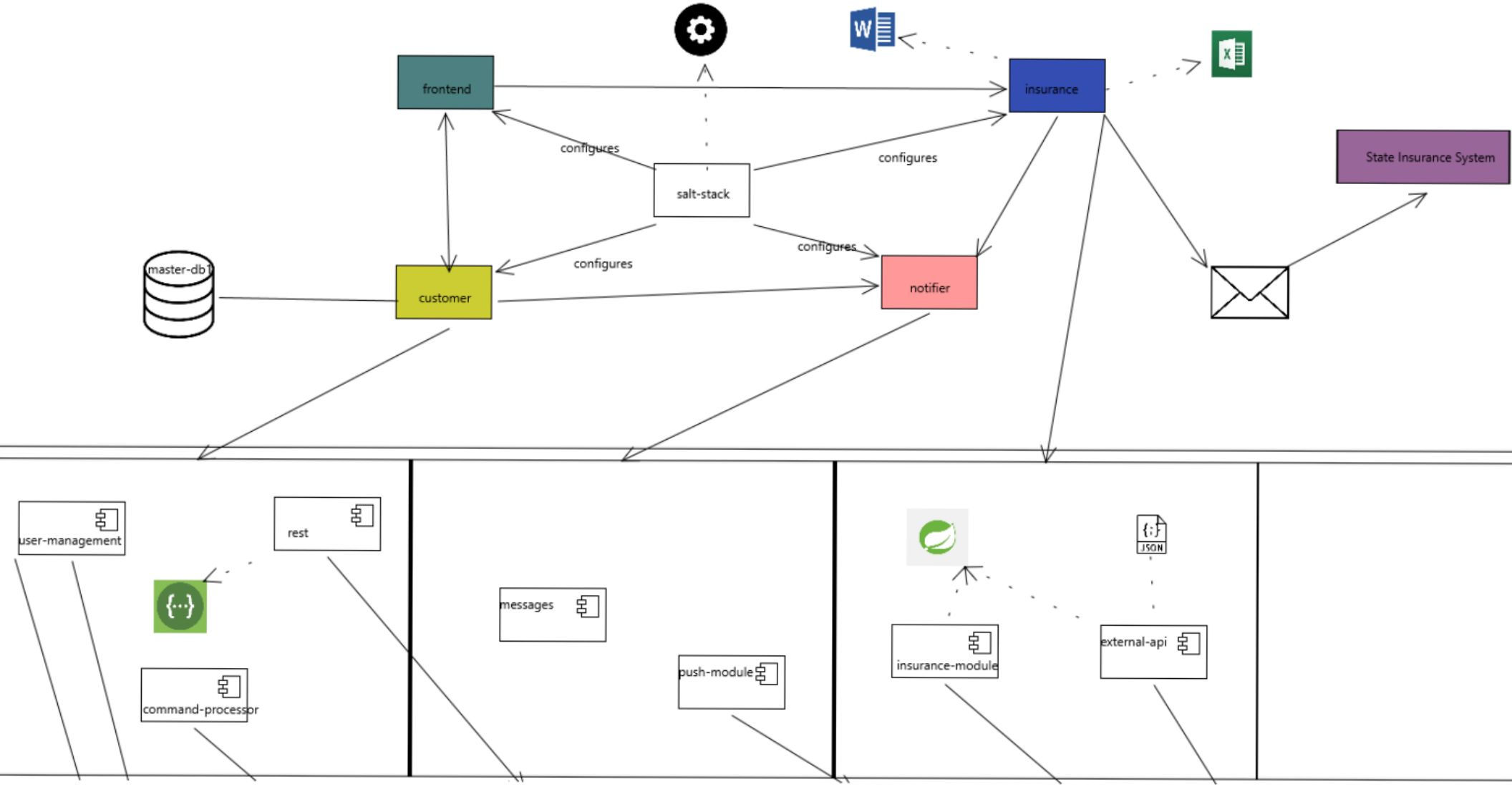

Level
Model
Abstract layers
Generic elements
Layer
Element
Property
AbstractEleme
No particular interpretation
nor visualization imposed
Connection
Flexible internal representation



Common 2D visualization

3D layered visualization

http://uml3d.herokuapp.com/



InterSKnow: an experimental modeling tool for multidimensional software knowledge

Evaluation performed on a model of a web based insurance system

InterSKnow outperforms
Enterprise Architect
in interrelating
heterogeneous software artifacts

Evaluation performed on a model of
a web based insurance system

InterSKnow outperforms
Enterprise Architect
in interrelating
heterogeneous software artifacts

Tasks assessed by:

- Number of context switches
- Number of files opened
- Time

Two perspectives:

1. Efficiency of searching for software knowledge
2. Comprehension of software knowledge

Ability to answer selected questions about the model

Software development involves working with large amounts of heterogeneous artifacts forming multidimensional software knowledge

The knowledge is in relationships: an interrelating model is necessary

A flexible internal representation based on abstract layers and generic elements enables to cope with varying interpretation and visualization of the interrelating model

Valentino Vranić
vranic@stuba.sk
AdvanSD.fiit.stuba.sk

STU FIIT
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Evaluation indicates that InterSKnow outperforms Enterprise Architect in interrelating heterogeneous software artifacts