FOKUS

Fraunhofer Institute for Open Communication Systems

# UML Notation for an Automated Deployment Process

Julia Reznik, Marc Born

GMD Fokus

*{reznik,born}@fokus.fhg.de*

SECOND WORKSHOP ON UML™ for Enterprise Applications: Model Driven Solutions for the Enterprise

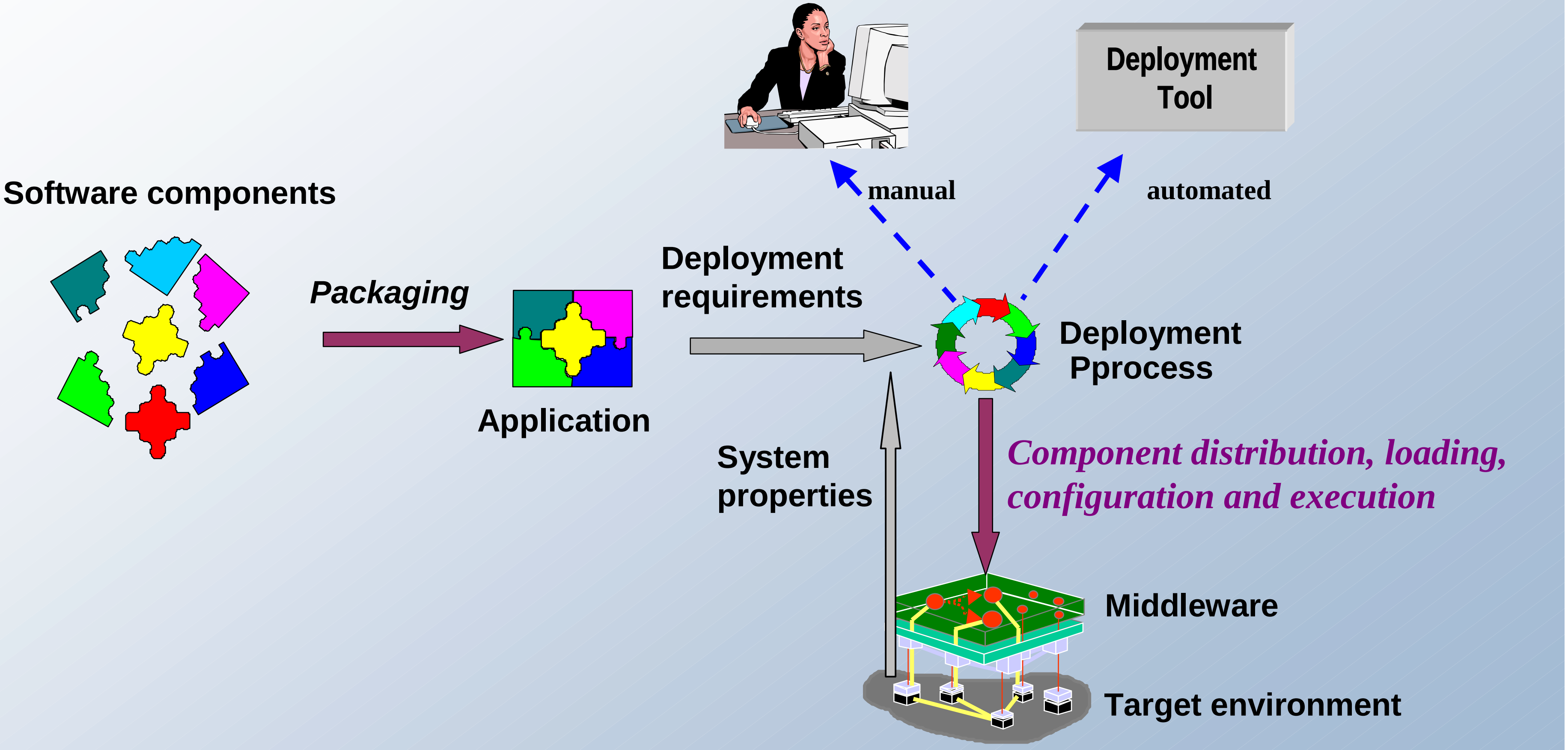
# Agenda

- Motivation
- Packaging and Deployment
- Model Driven Solution in order to support the automation of software deployment process
  - UML Notation for software components
  - CCM: Packaging and Deployment
  - automated Generation of CCM Deployment Description

# Motivation

- Large network environments have moved to the center stage in the field of software deployment
- New tools are needed to automate the component-based software deployment life cycle in a distributed environment
- Deployment life cycle: wide range of deployment tasks:
  - Packaging, installation, update, reconfigure, adapt, remove

# Packaging and Deployment



Software components

*Packaging*

Application

Deployment requirements

System properties

manual

automated

Deployment Tool

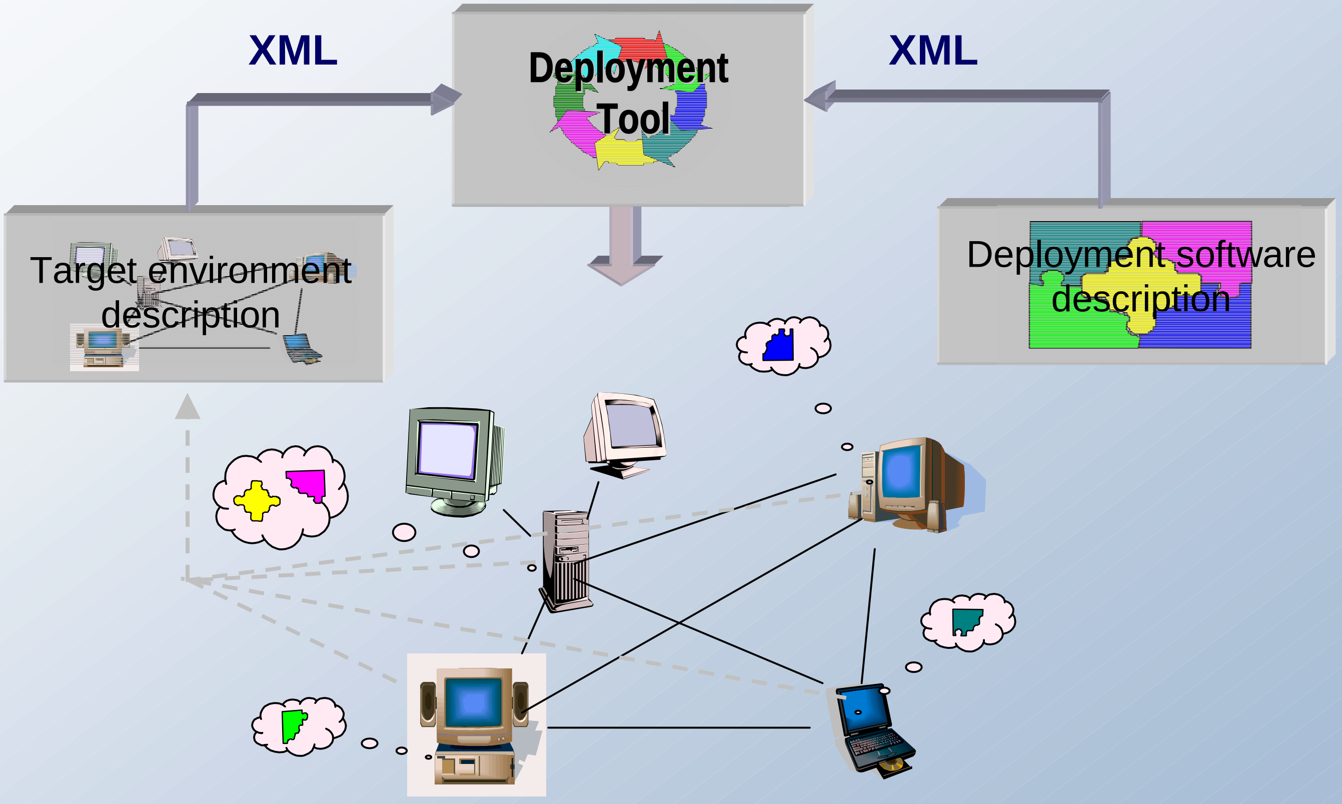
Deployment Pprocess

*Component distribution, loading, configuration and execution*

Middleware

Target environment

# Deployment Tool

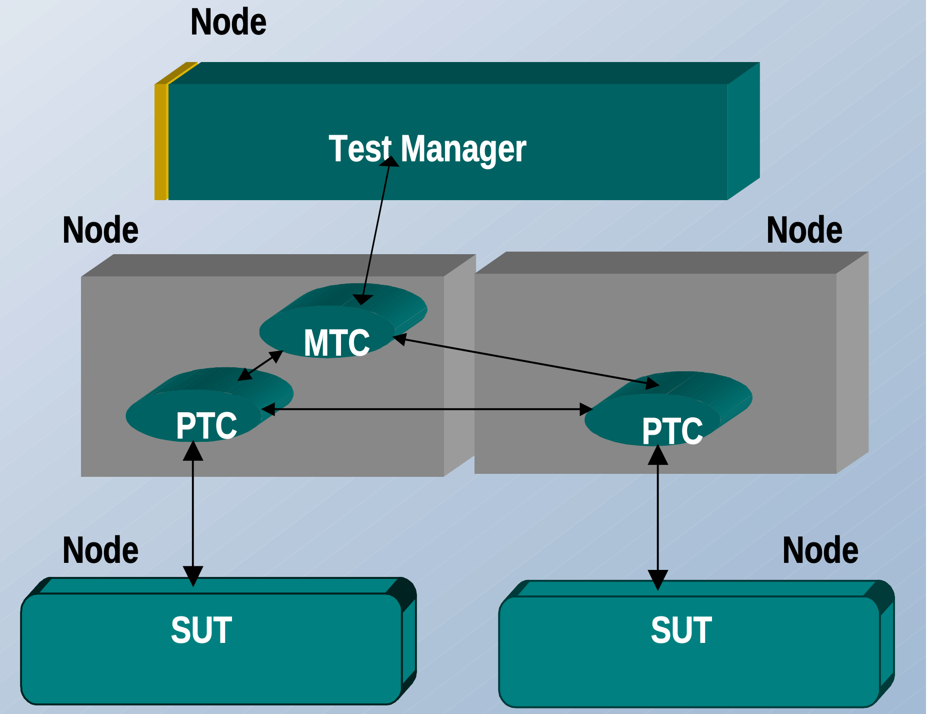

XML
Deployment Tool
XML
Target environment description
Deployment software description

# MDA Solution: Deployment Software Description

- Bridges the gap between design and deployment phase of distributed applications
- Provides conventions for applying and specializing standard UML to the graphical deployment notation
- Bases on the the definitions of the XML DTDs used by the CORBA Components
- Can be realized with existing UML tools (e.g. Rational Rose)
- Provides automatic generation of XML descriptors with a UML tool

# Test Service Example for distributed systems

- Computational Object Types (COs):
  - TestManager
  - Main Test Component (MTC)
  - Parallel Test Component (PTC)
  - System Under Test (SUT)



# UML Extension Mechanisms

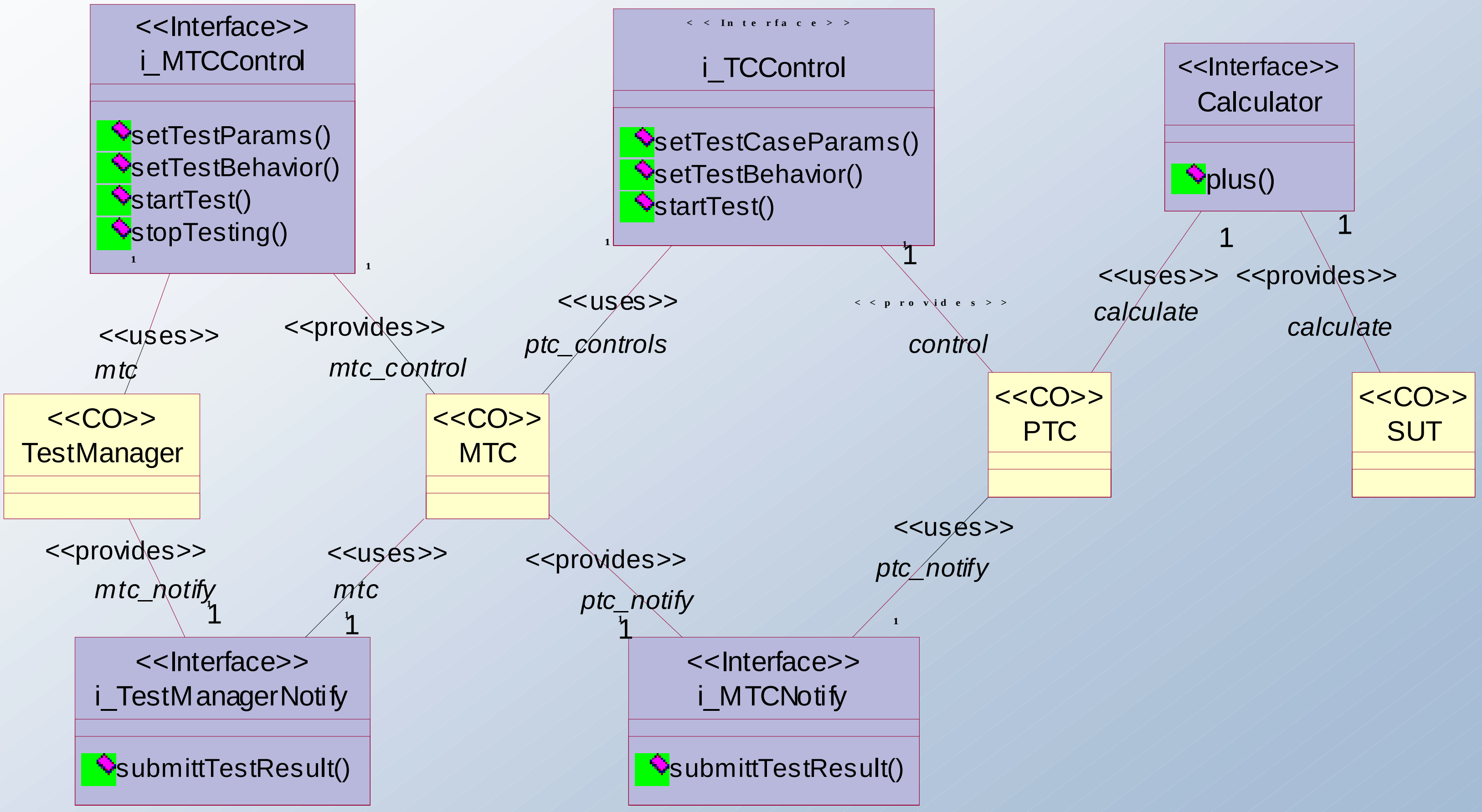
- Stereotypes:
  - Sub-classification of an existing UML element
  - The new element has its own special properties (expressed as tagged values), semantics and notation
- Tagged Values:
  - New information about model elements and presentation elements (new properties)
- Constraints:
  - Conditions and restrictions, that apply to model elements

# Diagrams and Stereotypes

- Class diagram : Computational Object Types (CO Types) and their interfaces
- Collaboration diagram: Initial configuration of COs and their factories
- Component diagram: implementation components
- Stereotypes:

| UML-Metamodel element | Stereotype |
|---|---|
| Class | <<CO>> |
| Class | <<Factory>> |
| Class | <<Interface>> |
| Component | <<implementation>> |
| Package | <<assembly>> |
| Association | <<uses>> |
| Association | <<provides>> |
| Dependency | <<instantiate>> |

# Class diagram: CO Types and interfaces



<<Interface>>
i_MTCControl

setTestParams()
setTestBehavior()
startTest()
stopTesting()

<<Interface>>
i_TCControl

setTestCaseParams()
setTestBehavior()
startTest()

<<Interface>>
Calculator

plus()

<<uses>>
mtc

<<provides>>
mtc_control

<<uses>>
ptc_controls

<<provides>>
control

<<uses>>
calculate

<<provides>>
calculate

<<CO>>
TestManager

<<CO>>
MTC

<<CO>>
PTC

<<CO>>
SUT

<<provides>>
mtc_notify

<<uses>>
mtc

<<provides>>
ptc_notify

<<uses>>
ptc_notify

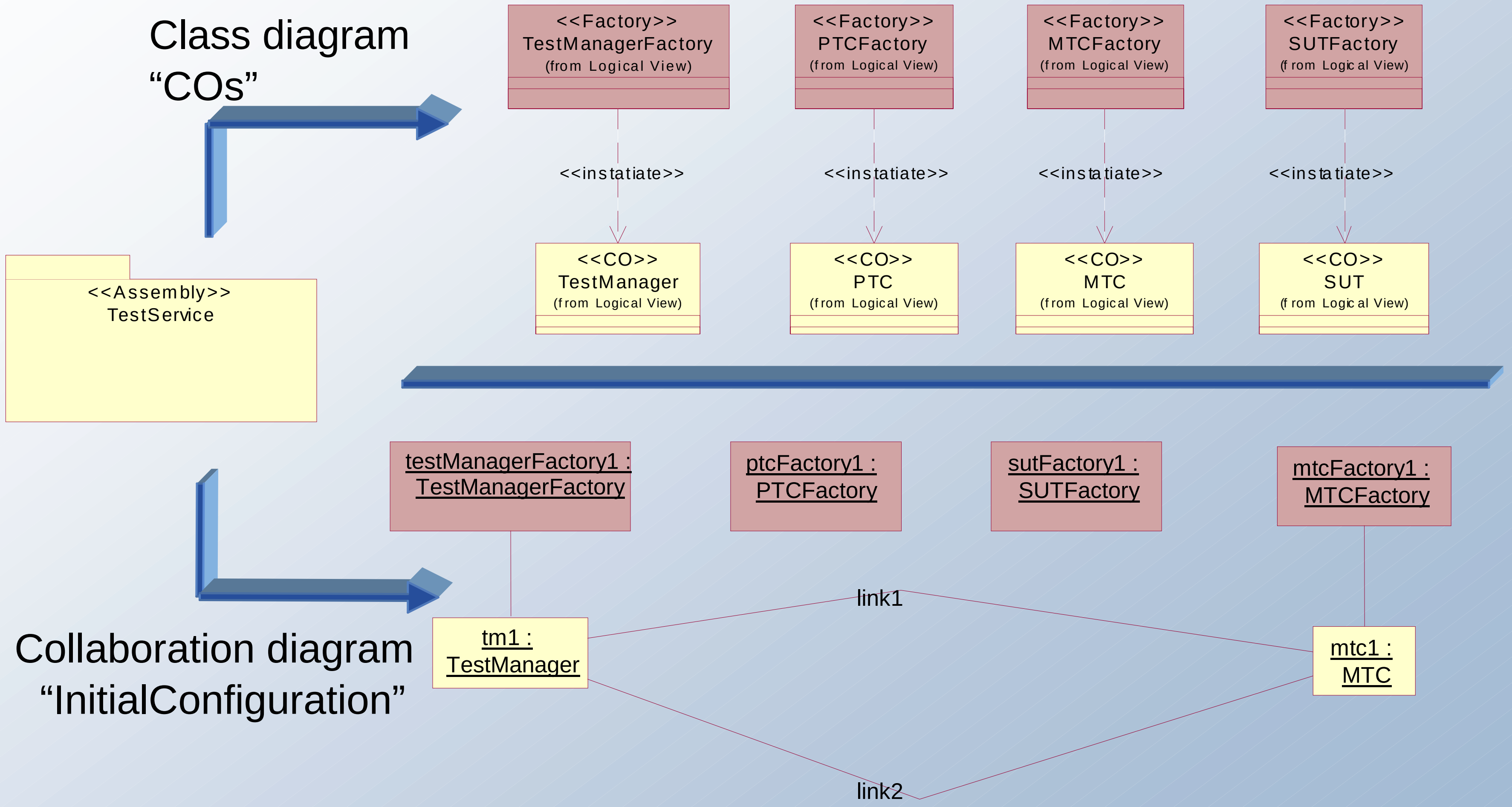
<<Interface>>
i_TestManagerNotify

submittTestResult()

<<Interface>>
i_MTCNotify

submittTestResult()

# Initial Configuration



Class diagram
“COs”

<<Factory>>
TestManagerFactory
(from Logical View)

<<Factory>>
PTCFactory
(from Logical View)

<<Factory>>
MTCFactory
(from Logical View)

<<Factory>>
SUTFactory
(from Logical View)

<<instatiate>>

<<instatiate>>

<<instatiate>>

<<instatiate>>

<<CO>>
TestManager
(from Logical View)

<<CO>>
PTC
(from Logical View)

<<CO>>
MTC
(from Logical View)

<<CO>>
SUT
(from Logical View)

<<Assembly>>
TestService

testManagerFactory1 : TestManagerFactory

ptcFactory1 : PTCFactory

sutFactory1 : SUTFactory

mtcFactory1 : MTCFactory

tm1 : TestManager

mtc1 : MTC

link1

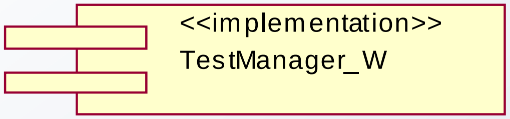
link2

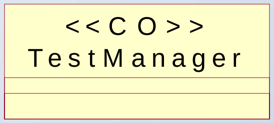
Collaboration diagram
“InitialConfiguration”

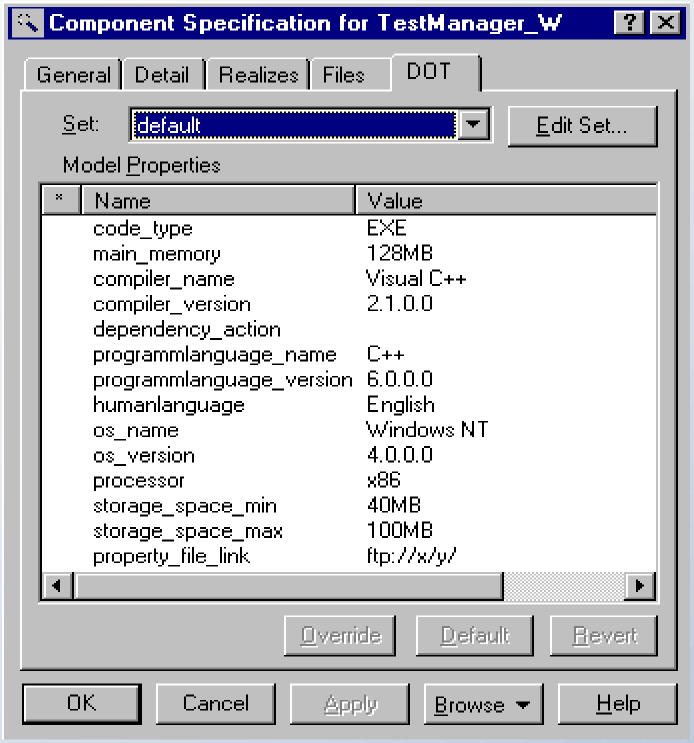
# Tagged Values (1)

- DCL specific keyword pairs: property name and value
- For stereotypes <<CO>> and <<Implementation>> tagged values are defined in the profile
- Source: CORBA Components Descriptors
  - Software Package
  - CORBA Component
  - Component Assembly

# Tagged Values (2)





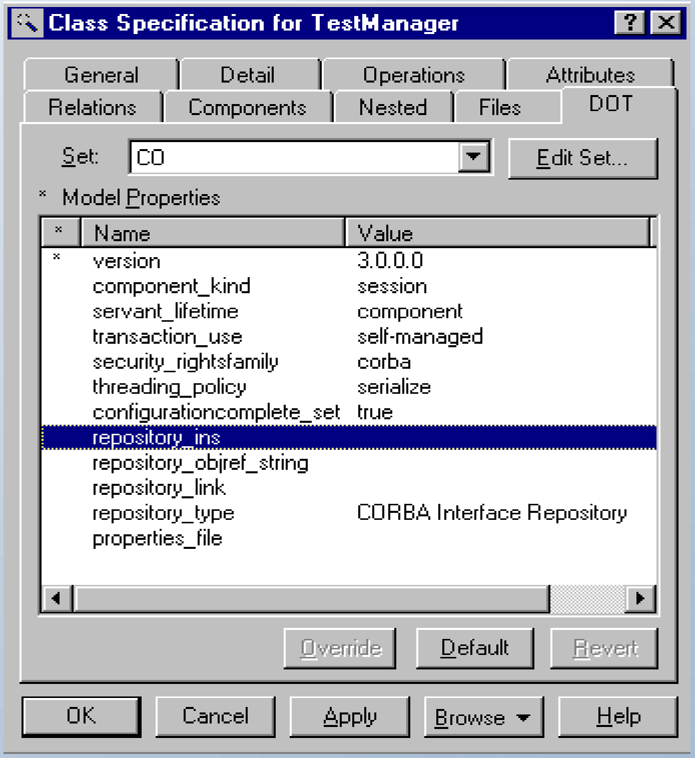


Component Specification for TestManager_W

Set: default

Model Properties

| * | Name | Value |
|---|---|---|
| | code_type | EXE |
| | main_memory | 128MB |
| | compiler_name | Visual C++ |
| | compiler_version | 2.1.0.0 |
| | dependency_action | |
| | programmlanguage_name | C++ |
| | programmlanguage_version | 6.0.0.0 |
| | humanlanguage | English |
| | os_name | Windows NT |
| | os_version | 4.0.0.0 |
| | processor | x86 |
| | storage_space_min | 40MB |
| | storage_space_max | 100MB |
| | property_file_link | ftp://x/y/ |



Class Specification for TestManager

Set: CO

* Model Properties

| * | Name | Value |
|---|---|---|
| * | version | 3.0.0.0 |
| | component_kind | session |
| | servant_lifetime | component |
| | transaction_use | self-managed |
| | security_rightsfamily | corba |
| | threading_policy | serialize |
| | configurationcomplete_set | true |
| | repository_ins | |
| | repository_objref_string | |
| | repository_link | |
| | repository_type | CORBA Interface Repository |
| | properties_file | |

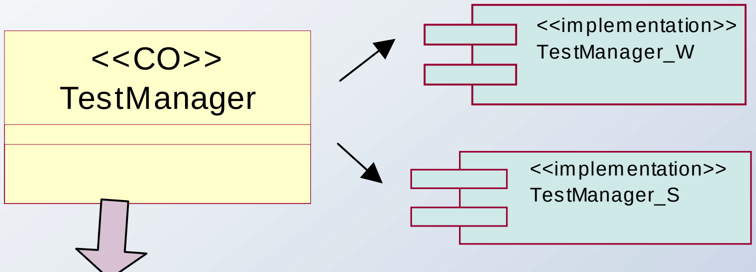
# Constraints

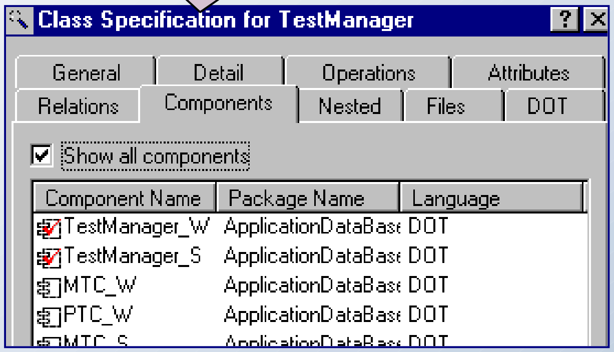
- <<assembly>> must have two diagrams: “COs” and “InitialConfiguration” diagram
- a class diagram of <<assembly>> package contains only classes with <<CO>> and <<Factory>>
- Generalisation: all elements of the same stereotype (e.g. <<CO>>)
- Valid association stereotype combinations:

| To: <br> From: | <<CO>> | <<Factory>> | <<Interface>> |
|---|---|---|---|
| <<CO>> | | | <<uses>><br><<provides>> |
| <<Factory>> | <<instantiate>> | | |
| <<Interface>> | | | |

# Realization with UML tool Rational Rose

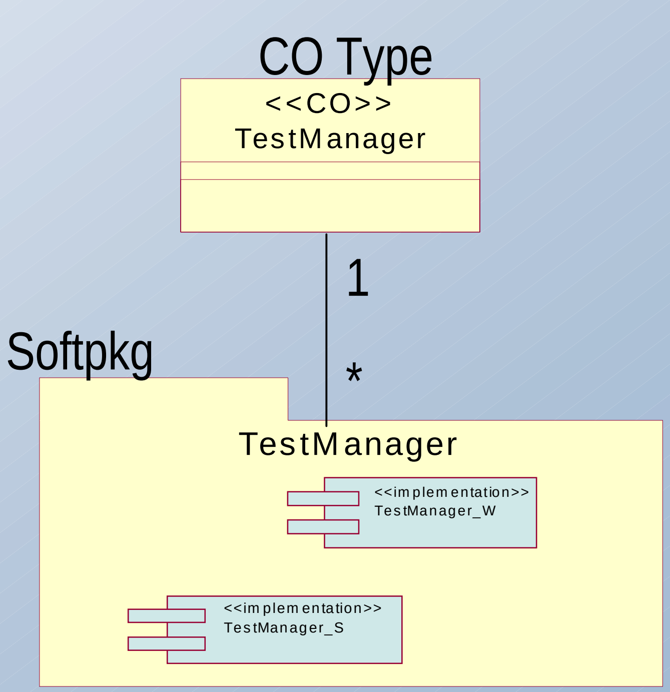
- Rational Rose provides
  - graphical support for UML
  - automatic generation of XML-Descriptors using Rose Extensibility Interface (REI)



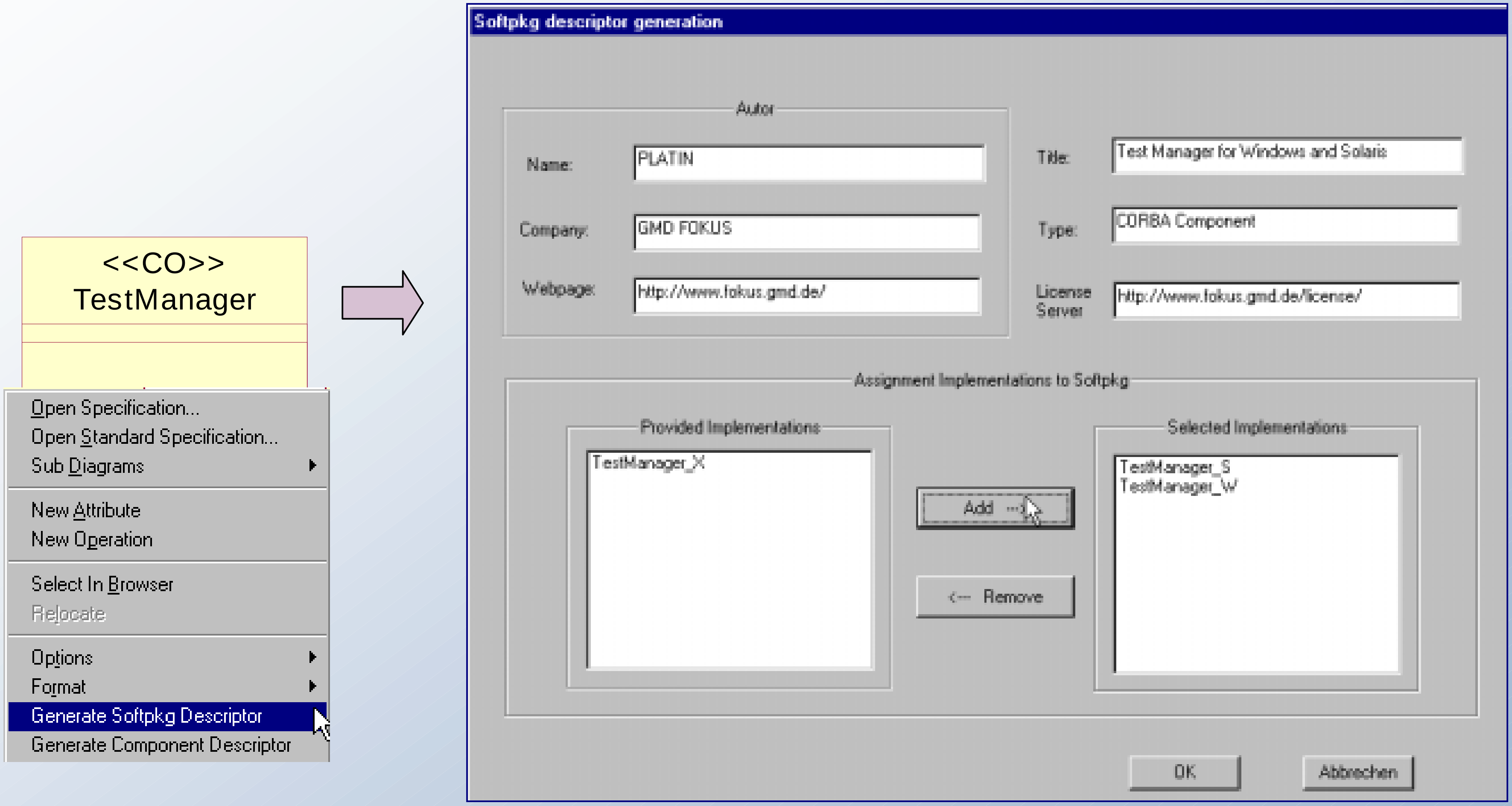


☑ Show all components

| Component Name | Package Name | Language |
|---|---|---|
| TestManager_W | ApplicationDataBas | DOT |
| TestManager_S | ApplicationDataBas | DOT |
| MTC_W | ApplicationDataBas | DOT |
| PTC_W | ApplicationDataBas | DOT |
| MTC_S | ApplicationDataBas | DOT |



# Generation of softpkg descriptor (1)



Softpkg descriptor generation

Autor

Name: PLATIN

Company: GMD FOKUS

Webpage: http://www.fokus.gmd.de/

Title: Test Manager for Windows and Solaris

Type: CORBA Component

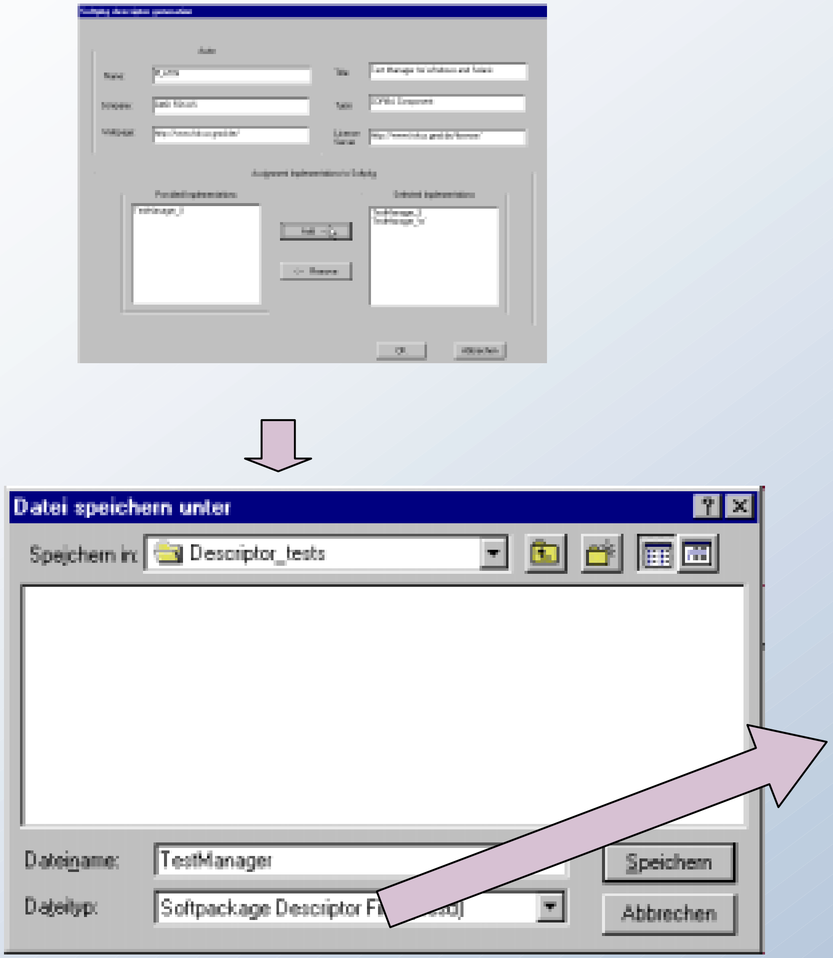
License Server: http://www.fokus.gmd.de/license/

Assignment Implementations to Softpkg

Provided Implementations: TestManager_X

Selected Implementations: TestManager_S, TestManager_W

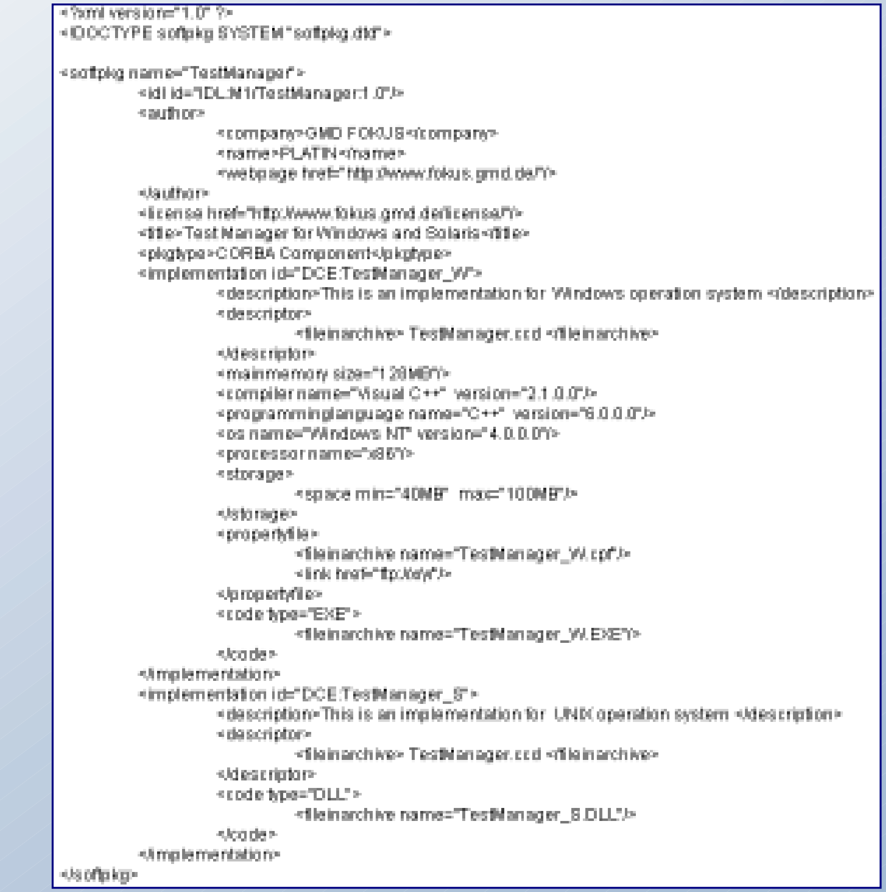
# Generation of Softpkg descriptor (2)



Speichern in: Descriptor_tests

Dateiname: TestManager

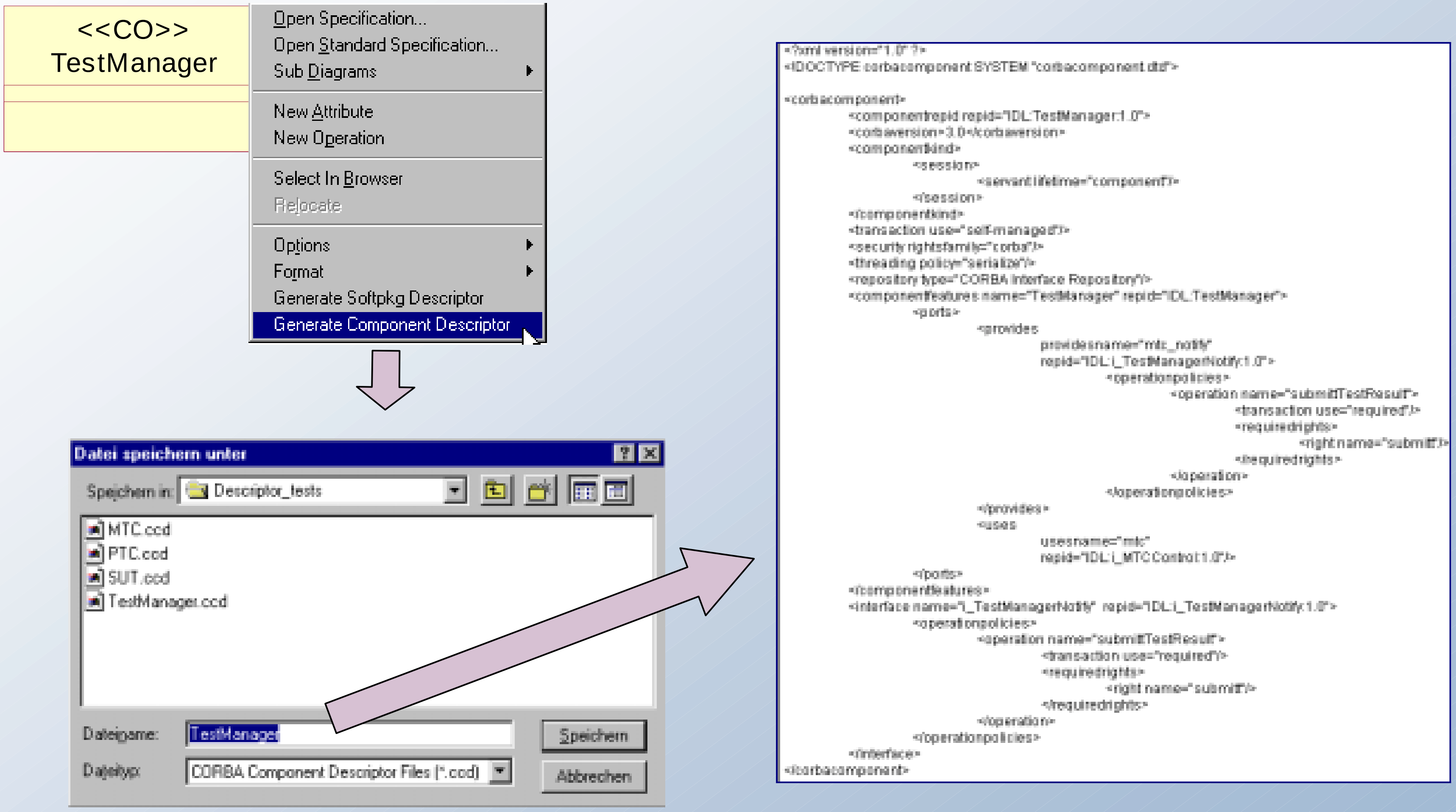
Dateityp: Softpackage Descriptor Fi[illegible]sd)



```
<?xml version="1.0" ?>
<!DOCTYPE softpkg SYSTEM"softpkg.dtd">

<softpkg name="TestManager">
        <idl id="IDL:M1/TestManager:1.0"/>
        <author>
                <company>GMD FOKUS</company>
                <name>PLATIN</name>
                <webpage href="http://www.fokus.gmd.de/"/>
        </author>
        <license href="http://www.fokus.gmd.de/license/"/>
        <title>Test Manager for Windows and Solaris</title>
        <pkgtype>CORBA Component</pkgtype>
        <implementation id="DCE:TestManager_W">
                <description>This is an implementation for Windows operation system </description>
                <descriptor>
                        <fileinarchive> TestManager.ccd </fileinarchive>
                </descriptor>
                <mainmemory size="128MB"/>
                <compiler name="Visual C++"  version="2.1.0.0"/>
                <programminglanguage name="C++"  version="6.0.0.0"/>
                <os name="Windows NT" version="4.0.0.0"/>
                <processor name="x86"/>
                <storage>
                        <space min="40MB"  max="100MB"/>
                </storage>
                <propertyfile>
                        <fileinarchive name="TestManager_W.cpf"/>
                        <link href="ftp://x/y/"/>
                </propertyfile>
                <code type="EXE">
                        <fileinarchive name="TestManager_W.EXE"/>
                </code>
        </implementation>
        <implementation id="DCE:TestManager_S">
                <description>This is an implementation for  UNIX operation system </description>
                <descriptor>
                        <fileinarchive> TestManager.ccd </fileinarchive>
                </descriptor>
                <code type="DLL">
                        <fileinarchive name="TestManager_S.DLL"/>
                </code>
        </implementation>
</softpkg>
```

# Generation of Component descriptor



Speichern in: Descriptor_tests

MTC.ccd
PTC.ccd
SUT.ccd
TestManager.ccd

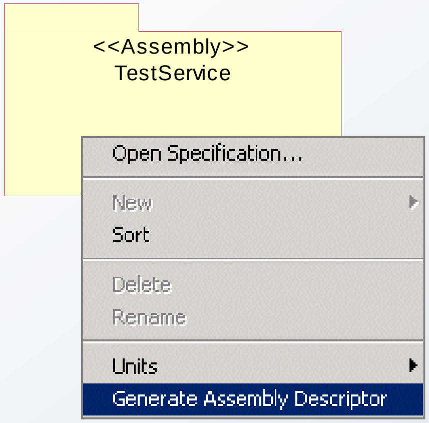
Dateiname: TestManager

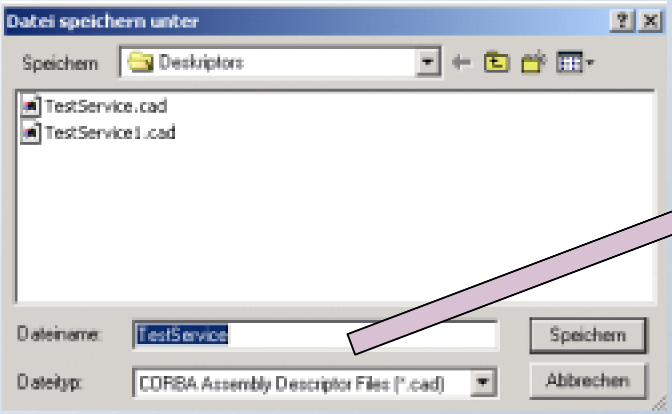
Dateityp: CORBA Component Descriptor Files (*.ccd)

```
<?xml version="1.0" ?>
<!DOCTYPE corbacomponent SYSTEM "corbacomponent.dtd">

<corbacomponent>
        <componentrepid repid="IDL:TestManager:1.0">
        <corbaversion>3.0</corbaversion>
        <componentkind>
                <session>
                        <servant lifetime="component"/>
                </session>
        </componentkind>
        <transaction use="self-managed"/>
        <security rightsfamily="corba"/>
        <threading policy="serialize"/>
        <repository type="CORBA Interface Repository"/>
        <componentfeatures name="TestManager" repid="IDL:TestManager">
                <ports>
                        <provides
                                providesname="mtc_notify"
                                repid="IDL:i_TestManagerNotify:1.0">
                                        <operationpolicies>
                                                <operation name="submitTestResult">
                                                        <transaction use="required"/>
                                                        <requiredrights>
                                                                <right name="submit"/>
                                                        </requiredrights>
                                                </operation>
                                        </operationpolicies>
                        </provides>
                        <uses
                                usesname="mtc"
                                repid="IDL:i_MTCControl:1.0"/>
                </ports>
        </componentfeatures>
        <interface name="i_TestManagerNotify"  repid="IDL:i_TestManagerNotify:1.0">
                <operationpolicies>
                        <operation name="submitTestResult">
                                <transaction use="required"/>
                                <requiredrights>
                                        <right name="submit"/>
                                </requiredrights>
                        </operation>
                </operationpolicies>
        </interface>
</corbacomponent>
```

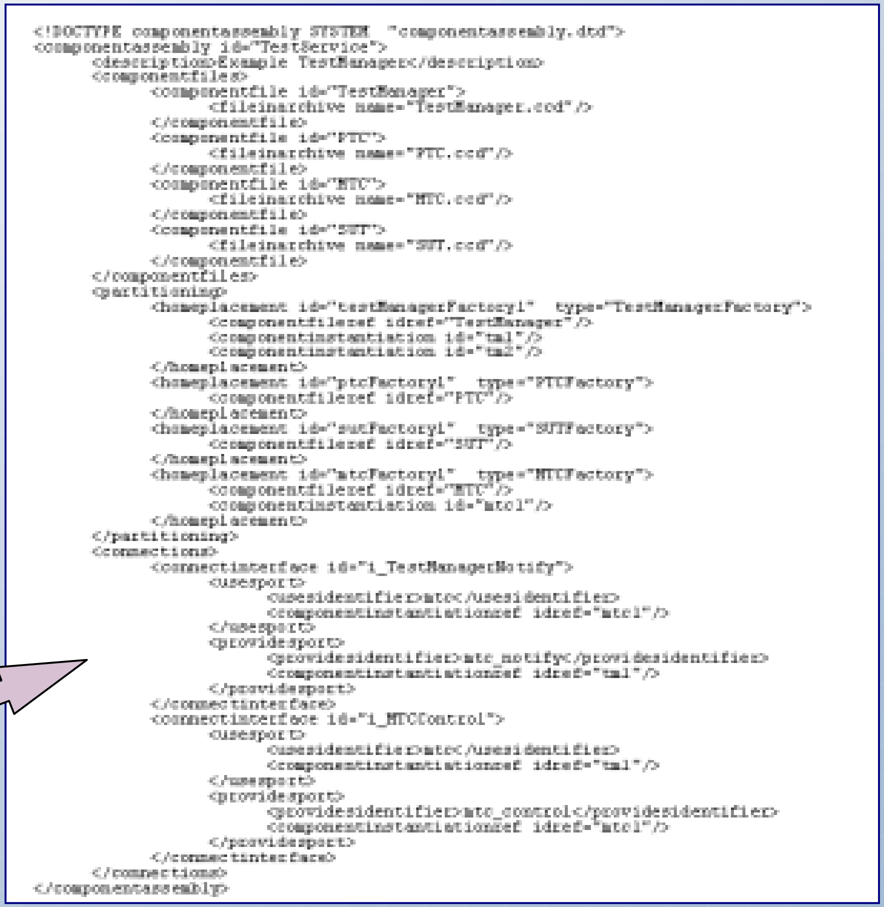
# Generation of Assembly descriptor





Dateiname: TestService

Dateityp: CORBA Assembly Descriptor Files (*.cad)



```
<!DOCTYPE componentassembly SYSTEM  "componentassembly.dtd">
<componentassembly id="TestService">
        <description>Example TestManager</description>
        <componentfiles>
                <componentfile id="TestManager">
                        <fileinarchive name="TestManager.ccd"/>
                </componentfile>
                <componentfile id="PTC">
                        <fileinarchive name="PTC.ccd"/>
                </componentfile>
                <componentfile id="MTC">
                        <fileinarchive name="MTC.ccd"/>
                </componentfile>
                <componentfile id="SUT">
                        <fileinarchive name="SUT.ccd"/>
                </componentfile>
        </componentfiles>
        <partitioning>
                <homeplacement id="testManagerFactory1"  type="TestManagerFactory">
                        <componentfileref idref="TestManager"/>
                        <componentinstantiation id="tm1"/>
                        <componentinstantiation id="tm2"/>
                </homeplacement>
                <homeplacement id="ptcFactory1"  type="PTCFactory">
                        <componentfileref idref="PTC"/>
                </homeplacement>
                <homeplacement id="sutFactory1"  type="SUTFactory">
                        <componentfileref idref="SUT"/>
                </homeplacement>
                <homeplacement id="mtcFactory1"  type="MTCFactory">
                        <componentfileref idref="MTC"/>
                        <componentinstantiation id="mtc1"/>
                </homeplacement>
        </partitioning>
        <connections>
                <connectinterface id="i_TestManagerNotify">
                        <usesport>
                                <usesidentifier>mtc</usesidentifier>
                                <componentinstantiationref idref="mtc1"/>
                        </usesport>
                        <providesport>
                                <providesidentifier>mtc_notify</providesidentifier>
                                <componentinstantiationref idref="tm1"/>
                        </providesport>
                </connectinterface>
                <connectinterface id="i_MTCControl">
                        <usesport>
                                <usesidentifier>mtc</usesidentifier>
                                <componentinstantiationref idref="tm1"/>
                        </usesport>
                        <providesport>
                                <providesidentifier>mtc_control</providesidentifier>
                                <componentinstantiationref idref="mtc1"/>
                        </providesport>
                </connectinterface>
        </connections>
</componentassembly>
```

# Conclusion

- Graphical support for automated deployment process
  - Specification of component-based software systems requirements
  - Specification of Initial Configuration of component instances
  - Automatic generation of XML-Descriptors for the Deployment process
- Under development:
  - Automatic generation of UML Deployment diagram from  the XML Description of target environment

# Thank you!