ESO
European Organisation for Astronomical Research in the Southern Hemisphere

TOWARDS MODEL RE-USABILITY FOR THE DEVELOPMENT OF TELESCOPE CONTROL SYSTEMS

R. Karban, L. Andolfato, M. Zamparelli
ESO, Munich, Germany

AGENDA

ESO
European Organisation for Astronomical Research in the Southern Hemisphere

- Document centric vs. Model centric approach
- Re-use of Requirements
  - Domain Specific Language
  - Instances of Boilerplates
  - Constrain the Design
- Re-use of Constraints for Interfaces
- Conclusion

ESO
European Organisation for Astronomical Research in the Southern Hemisphere

Document vs. Model centric (1/2)

Document centric

- Specify User Requirements as text
- Requirements analysis
- Write System requirements using text
- Describe level 1 design as text and model
- Write level 2 requirements as text
- Describe level 2 design as text and model

Model centric

- Specify User Requirements as text
- Requirements analysis, capturing key properties with parameters and constraints in the model
- Generate System Requirements document
- Describe level 1 design, formally constrained by requirements parameters
- Generate level 2 requirements document

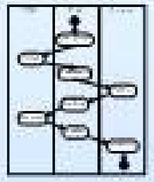
ESO
European Organisation
for Astronomical
Research in the
Southern Hemisphere

Document vs. Model centric (2/2)

Document centric

- Manual verification of Requirements and Design
- Manual creation of test cases
- Manual impact analysis using (inconsistent) requirements and design documentation

Model centric

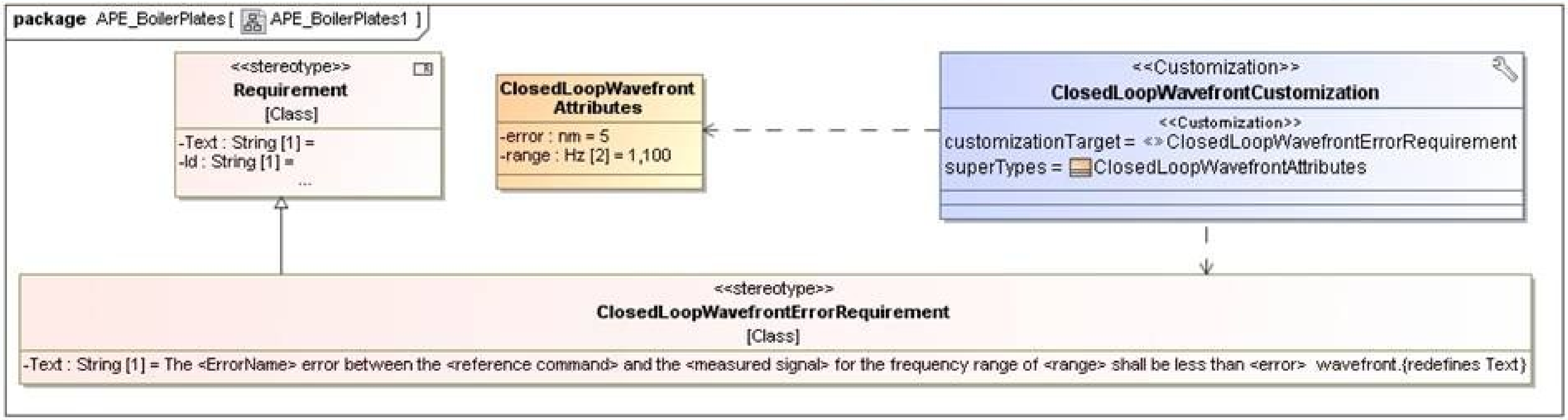


- Automatic validation of requirements and Design
- Automatic impact analysis – bottom up and top down

ESO
European Organisation for Astronomical Research in the Southern Hemisphere

REQUIREMENTS – Doman Specific Language

- Parameterize quantifiable parameters
- Standard text
- Reuse boilerplates for requirements



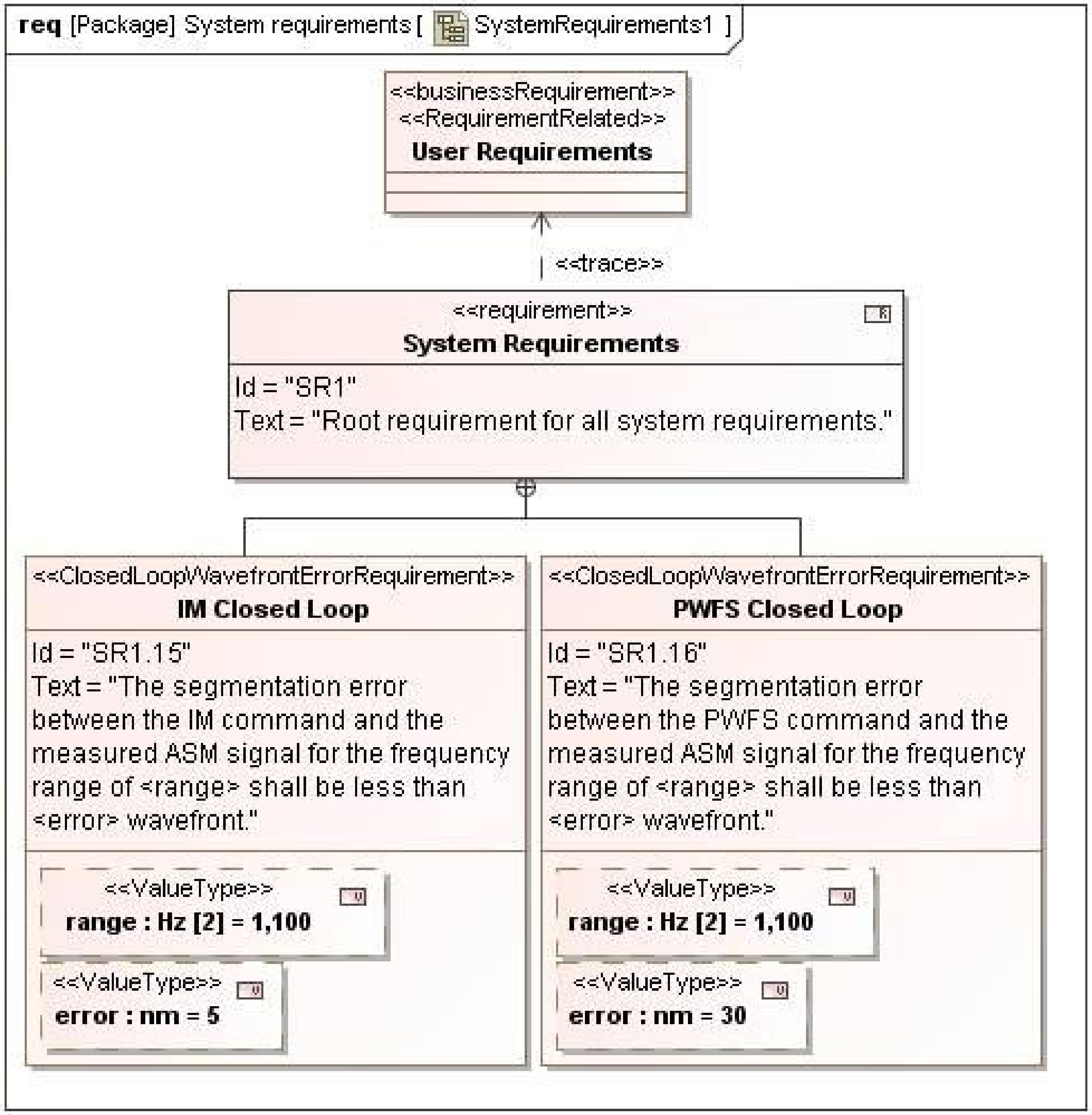
package APE_BoilerPlates [APE_BoilerPlates1]
<<stereotype>>
Requirement
[Class]
-Text : String [1] =
-Id : String [1] =
...
ClosedLoopWavefront Attributes
-error : nm = 5
-range : Hz [2] = 1,100
<<Customization>>
ClosedLoopWavefrontCustomization
<<Customization>>
customizationTarget = ClosedLoopWavefrontErrorRequirement
superTypes = ClosedLoopWavefrontAttributes
<<stereotype>>
ClosedLoopWavefrontErrorRequirement
[Class]

-Text : String [1] = The <ErrorName> error between the <reference command> and the <measured signal> for the frequency range of <range> shall be less than <error> wavefront.{redefines Text}

REQUIREMENTS – Instantiation of Boilerplates

ESO
European Organisation for Astronomical Research in the Southern Hemisphere

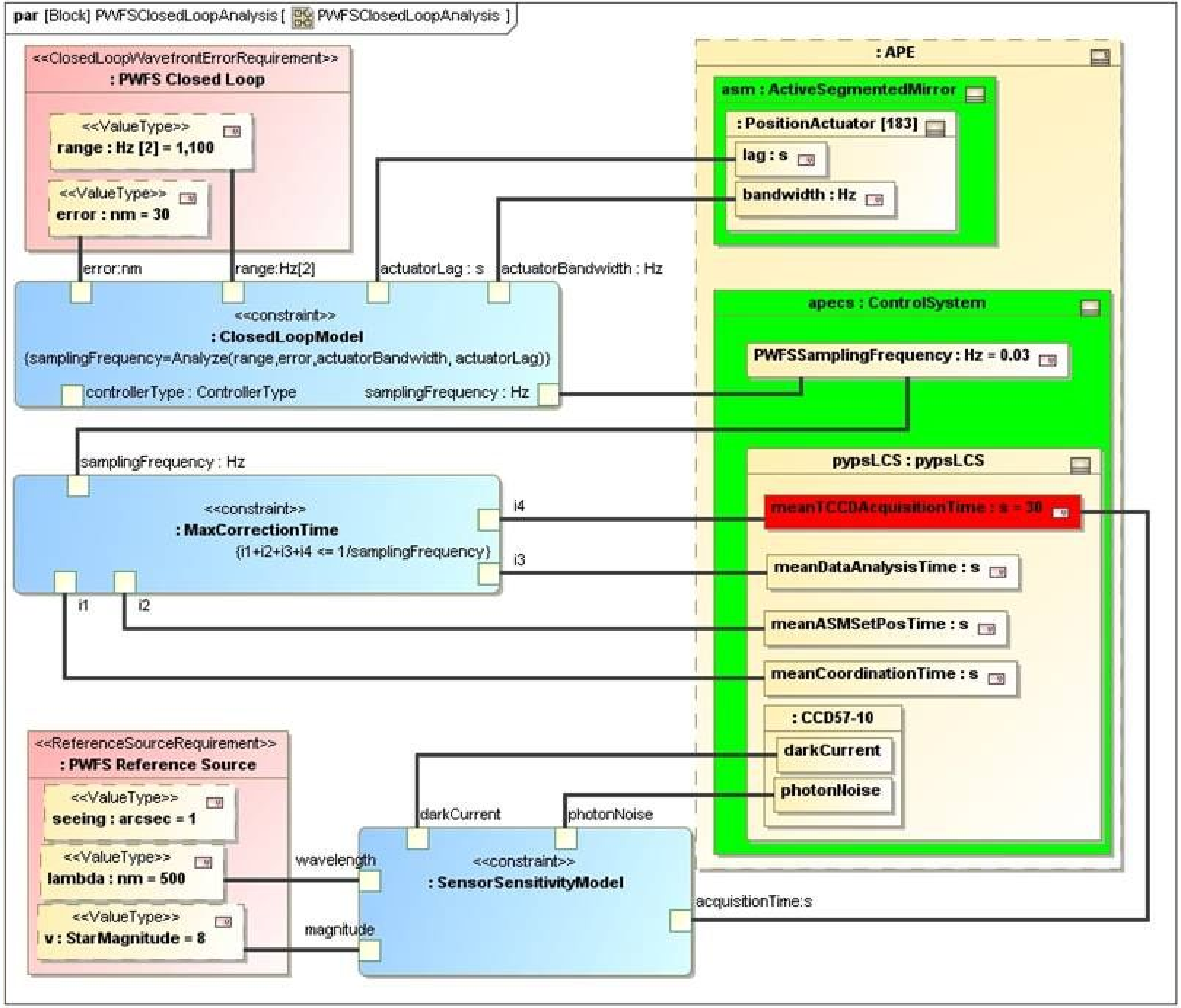
- Replace text parameters
- Redefine default values of quantifiable parameters



System Requirements:
Id = "SR1"
Text = "Root requirement for all system requirements."

IM Closed Loop:
Id = "SR1.15"
Text = "The segmentation error between the IM command and the measured ASM signal for the frequency range of <range> shall be less than <error> wavefront."

PWFS Closed Loop:
Id = "SR1.16"
Text = "The segmentation error between the PWFS command and the measured ASM signal for the frequency range of <range> shall be less than <error> wavefront."



par [Block] PWFSClosedLoopAnalysis [PWFSClosedLoopAnalysis]
<<ClosedLoopWavefrontErrorRequirement>>
: PWFS Closed Loop
<<ValueType>>
range : Hz [2] = 1,100
<<ValueType>>
error : nm = 30
error:nm
range:Hz[2]
actuatorLag : s
actuatorBandwidth : Hz
<<constraint>>
: ClosedLoopModel
{samplingFrequency=Analyze(range,error,actuatorBandwidth, actuatorLag)}
controllerType : ControllerType
samplingFrequency : Hz
samplingFrequency : Hz
<<constraint>>
: MaxCorrectionTime
{i1+i2+i3+i4 <= 1/samplingFrequency}
i4
i3
i1
i2
: APE
asm : ActiveSegmentedMirror
: PositionActuator [183]
lag : s
bandwidth : Hz
apecs : ControlSystem
PWFSSamplingFrequency : Hz = 0.03
pypsLCS : pypsLCS
meanTCCDAcquisitionTime : s = 30
meanDataAnalysisTime : s
meanASMSetPosTime : s
meanCoordinationTime : s
: CCD57-10
darkCurrent
photonNoise
<<ReferenceSourceRequirement>>
: PWFS Reference Source
<<ValueType>>
seeing : arcsec = 1
<<ValueType>>
lambda : nm = 500
<<ValueType>>
v : StarMagnitude = 8
wavelength
magnitude
darkCurrent
photonNoise
<<constraint>>
: SensorSensitivityModel
acquisitionTime:s

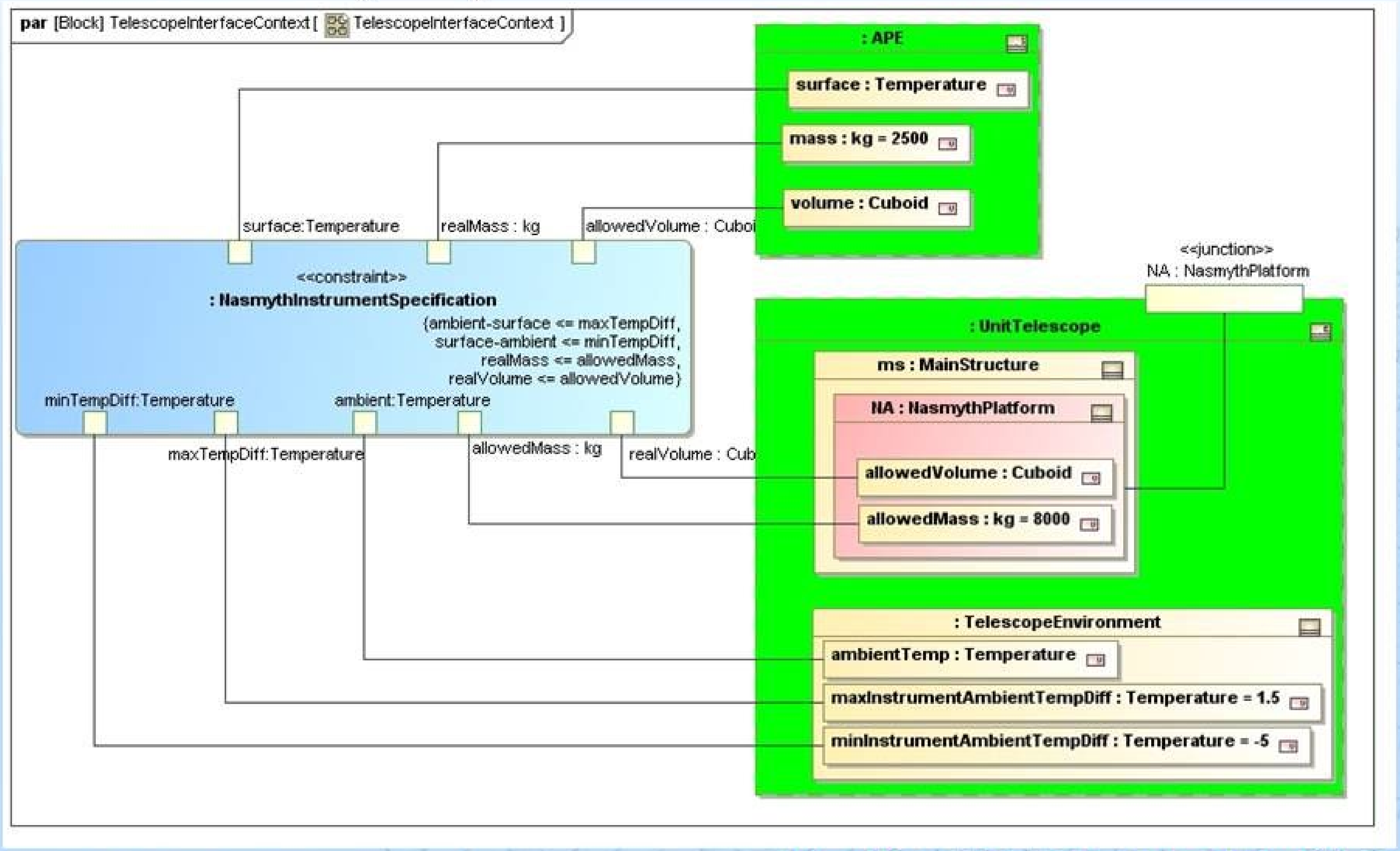
ESO
European Organisation for Astronomical Research in the Southern Hemisphere

Re-use of Constraints for Interfaces (1/2)

- Define quantifiable System Interfaces
- Constrain all participating components
- Propagate down the system hierarchy

Re-use of Constraints for Interfaces (2/2)

ESO
European Organisation for Astronomical Research in the Southern Hemisphere



ESO
European Organisation for Astronomical Research in the Southern Hemisphere

CONCLUSION

- Requirements become more than text
- Bind requirements' parameters to system properties
- Specify system interfaces as executable constraints
- Reusability of model artifacts (requirements, constraints, etc.) -> easier due to higher abstraction level and less dependence on actual implementation
- Requirement validation during design phases
- Consistent and correct documentation via auto-generation
- Consistent impact and trade-off analysis