From MODEL-BASED REQs to CONTRACTUAL REQS

2019 Acquisition Research Symposium

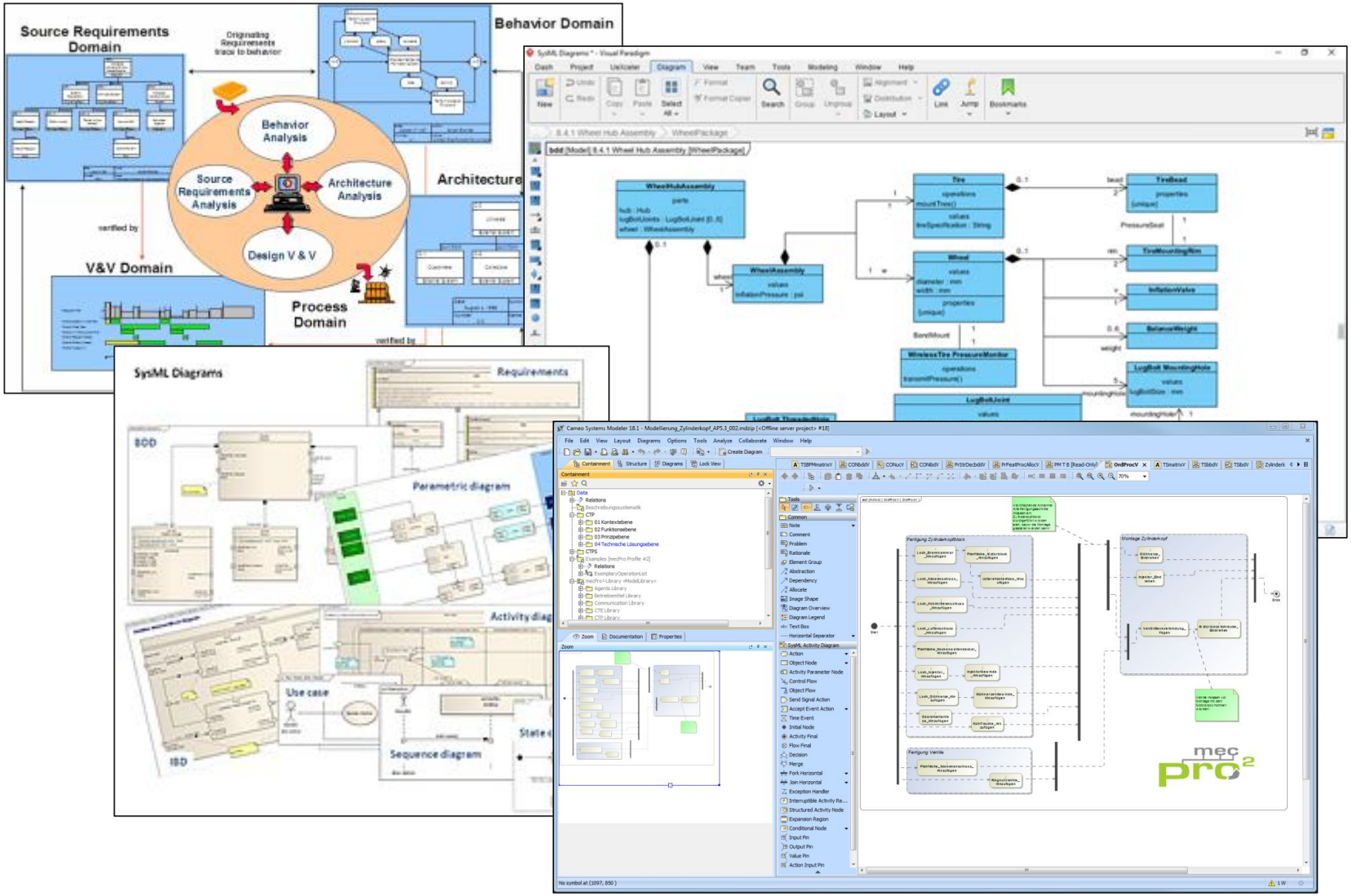
Dr. **Alejandro Salado** *Assistant Professor*

Paul Wach *Doctoral Student*

Grado Department of Industrial & Systems Engineering

VIRGINIA TECH™

MBSE is awesome



VIRGINIA TECH

The system shall sdgsgsrgsregwergwerg.

The system shall aojaionsfiusduin.

The system shall aojnaradfv.

The system shall adgaergsrth.

The system shall wergwergwergw.

The system shall sadfasdcasdc.

The system shall wergwergwerg.

The system shall wefvwefv.

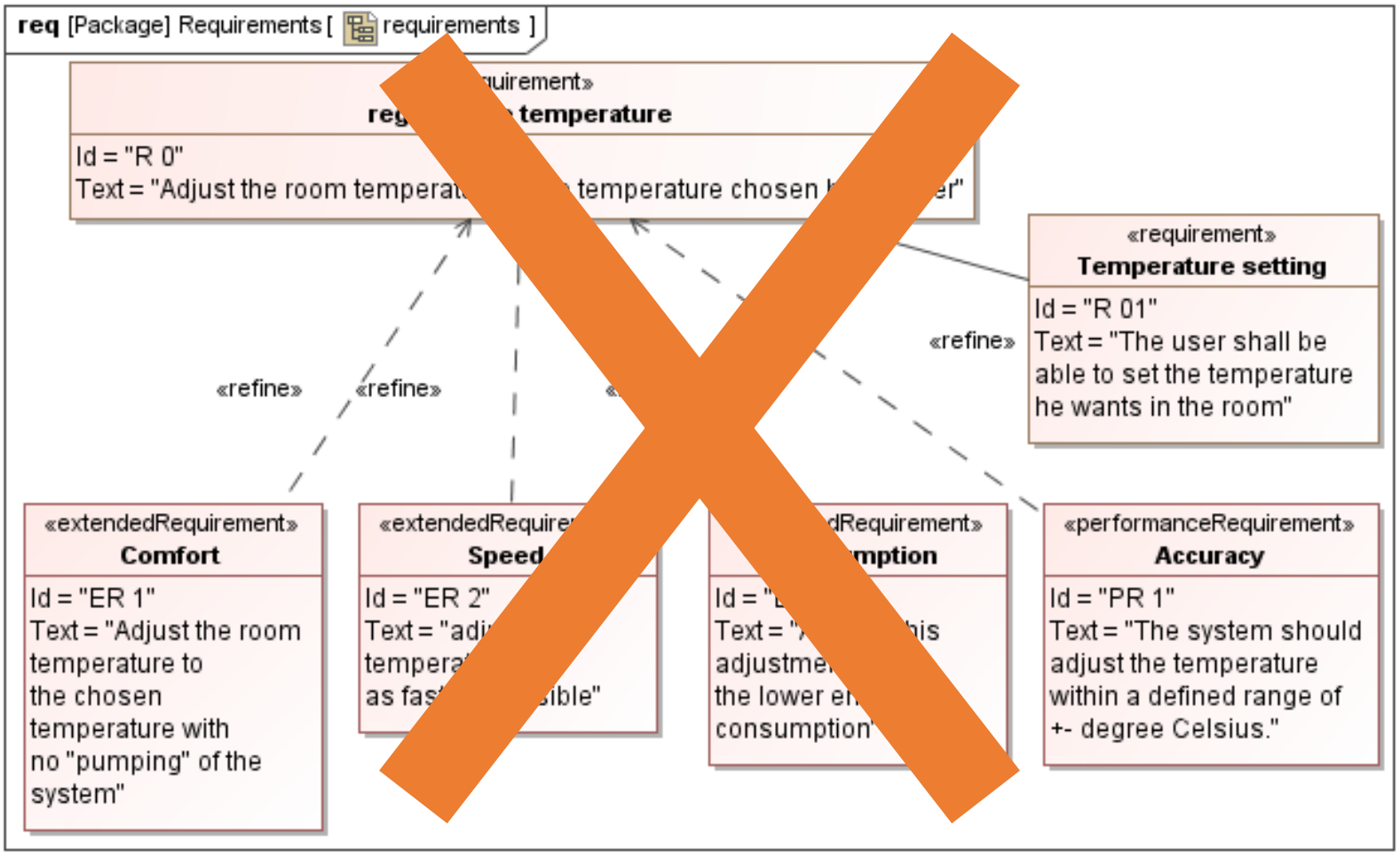
The system shall sdfgsdg.

The system shall kuiyklyu.

The system shall sdfvsrgrtg.

The system shall bfgnfghmy.

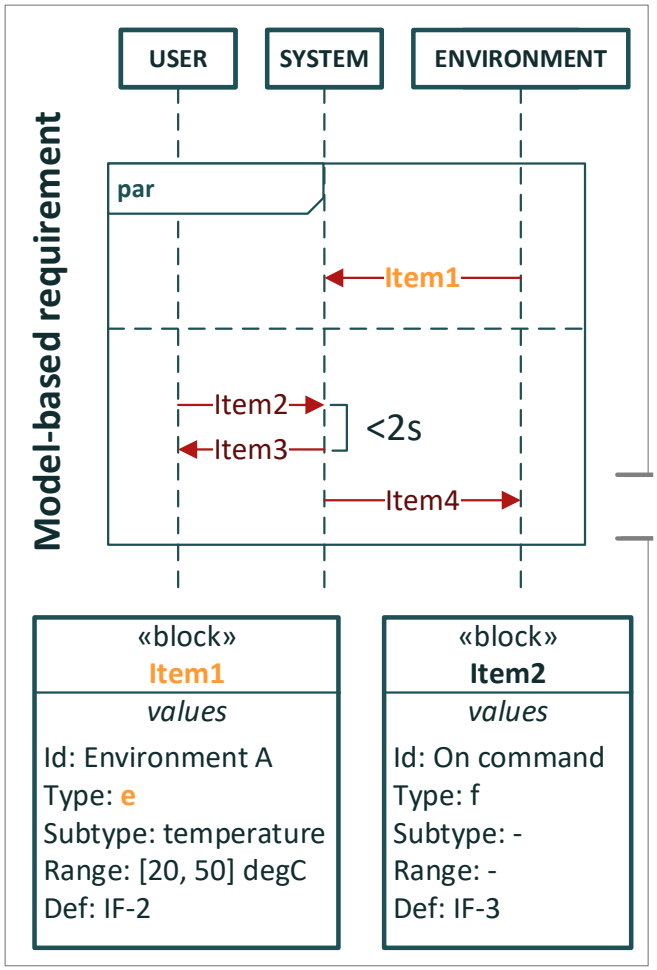
The system shall asdasdgaergwer.



Source: https://docs.nomagic.com/display/SYSMLP182/Requirement+Diagram

TOWARDS TRUE REQUIREMENT MODELS

VIRGINIA TECH



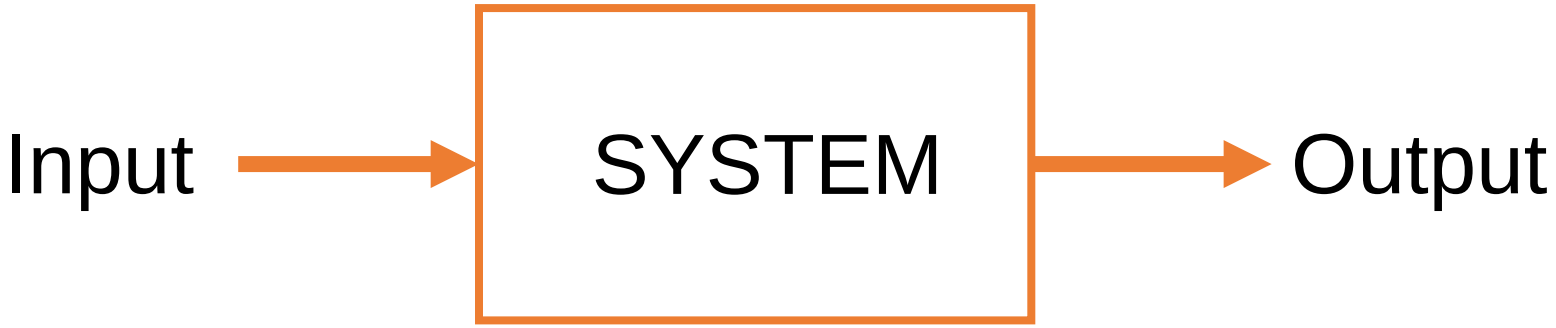
4 FOUNDATIONS

MATHEMATICAL CONSTRUCT Wymore, 1993

INTERFACE TYPOLOGY Various authors

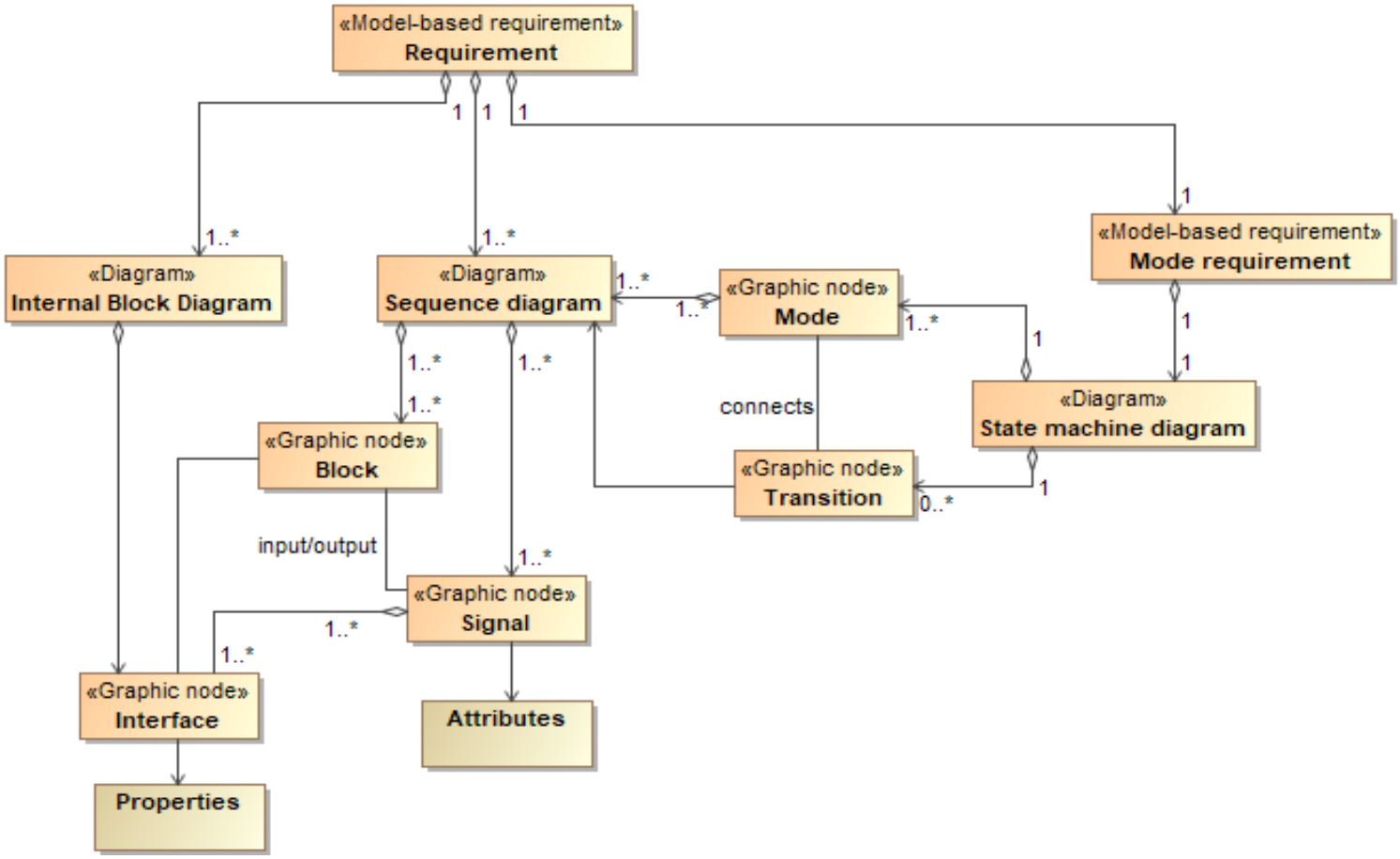
TEXTUAL CONSTRUCT INCOSE, 2012

REQUIREMENT TYPOLOGY Salado & Nilchiani, 2014



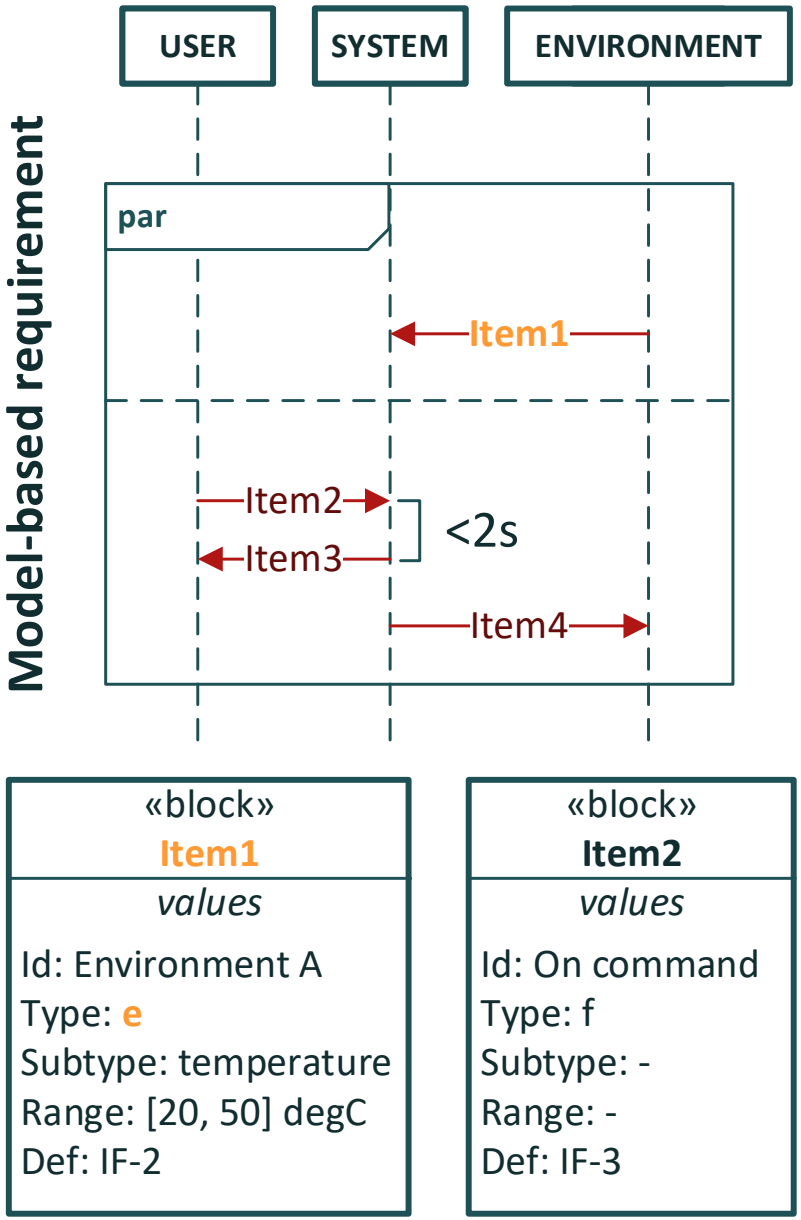
A system transforms inputs into outputs

A set of requirements defines a minimal transformation set of inputs into outputs



The system shall provide a signal through a physical interface.

VIRGINIA TECH



2 classes of seq diagrams

TYPESof interfaces

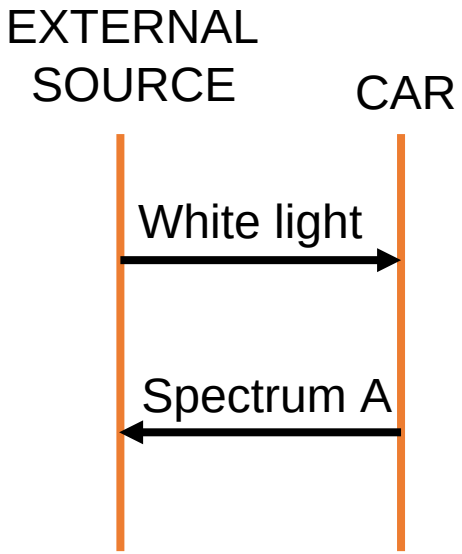
- Physical connection
- Energy flow
- Mass flow
- Information flow

Which ones are usually captured with sequence or state diagrams?

Can Wymore capture them all?

WHAT	Functional transformation
HOW WELL	Performance of the transformation
WITH WHAT	Resources employed
WHERE	Environment where transformation occurs

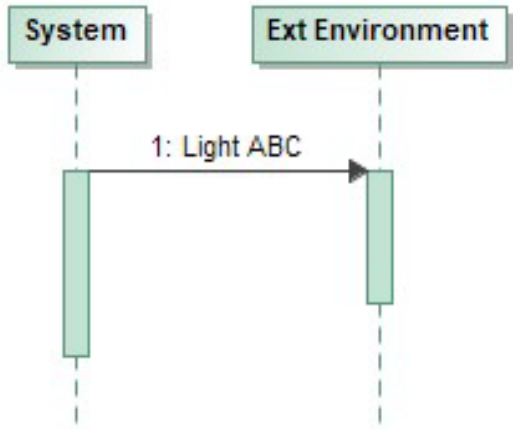
What about color?

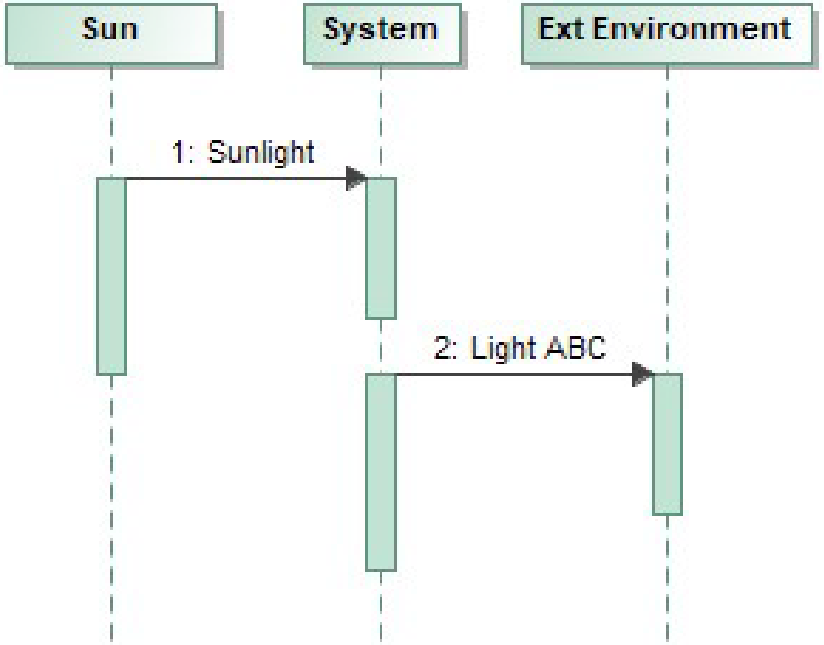


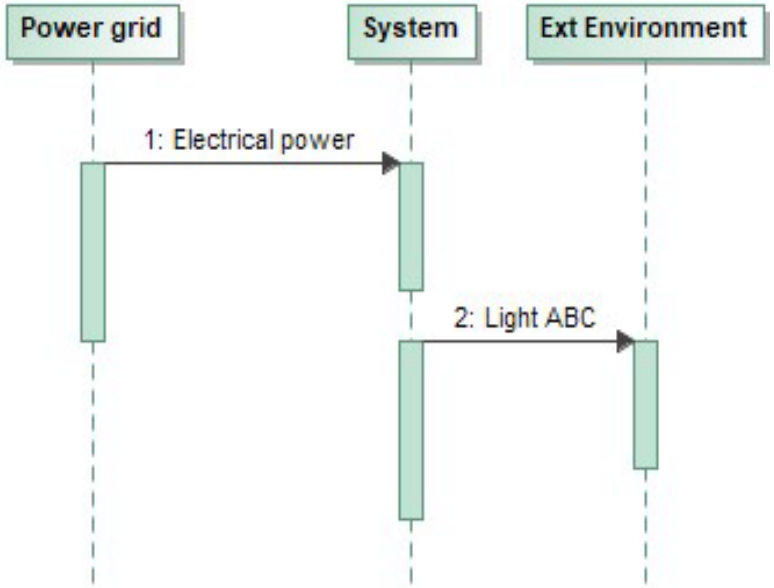
The system shall exhibit color ABC.

The system shall accept input light XYZ.

The system shall reflect light ABC when receiving input light XYZ.







The system shall <action> <object> <modifier 1> <modifier 2> <modifier 3> … <modifier n>.

Algorithm (sample)

for each element in Diagram 1

  createreq(element)

end

Algorithm (sample)

```
for each element in Diagram 1
  createreq(element)
end
```

```
createreq(Item1):
  if item1.type=e &syst.direction(item1)=in
    <req1.action>=”operate”
    <req1.object>=”in”&item1.subtype &
     “range”&item1.range
    <req1.modifier1>=”when fulfilling
     requirements”
    <req1.modifier2>=reqs(Diagram1)
    <req1.modifier 3>=”Note: Interface defined
     in”&item1.Def
  if item1.type=f &syst.direction(item1)=in
    <req1.action>=”accept”
    <req1.object>=item1.Id
    <req1.modifier>=”according to IF defined
     in”&item1.Def
  if item1.type=f &syst.direction(item1)=out
    <req1.action>=”provide”
    ...
```

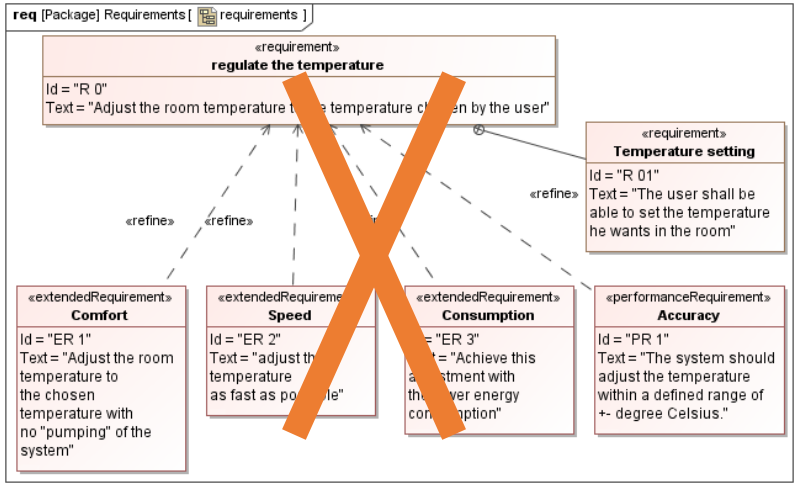
BENEFITS

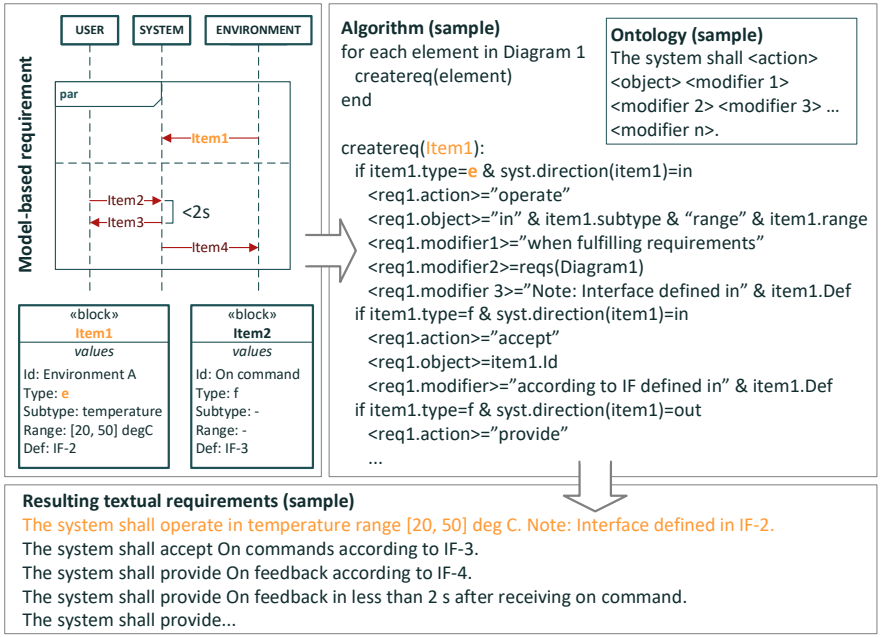
Consistent grammar

Requirement coverage

Requirement inconsistency

WRAPPINGUP





VIRGINIA TECH

This material is based upon work supported by the Acquisition Research Program under HQ0034-18-1-0006. The views expressed in written materials or publications, and/or made by speakers, moderators, and presenters, do not necessarily reflect the official policies of the Department of Defense nor does mention of trade names, commercial practices, or organizations imply endorsement by the U.S. Government.

THANK YOU

asalado@vt.edu

VIRGINIA TECH