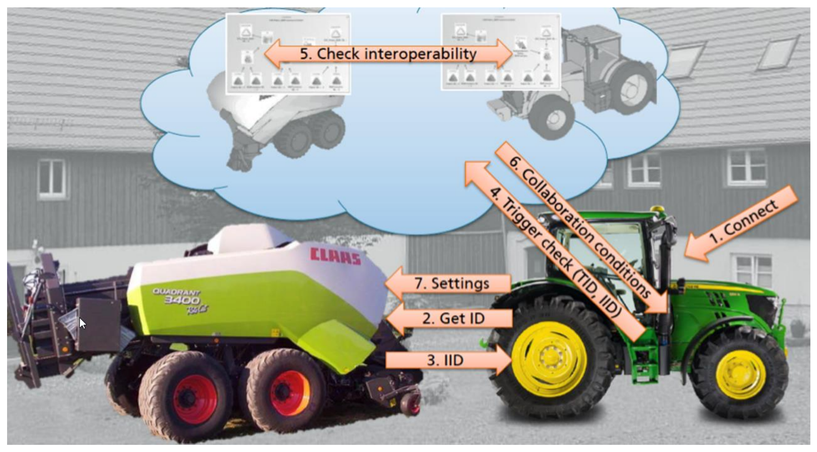
Designing Cyber-Physical Systems using aDSL: a Domain-Specific Language and Tool Support

Freek van den Berg[ut], Vahid Garousi[wur], Bedir Tekinerdogan[wur], and Boudewijn R. Haverkort[ut].

Context & Problem statement

- Cyber-physical systems (CPSs)
  - Integrate computation, software, networking and physical processes[1]
  - A CPS model extends an embedded system model with networking, time synchronization, and **interoperability**[2].
  - we propose aDSL, a DSL with tool support for the interoperability of CPSs[3]



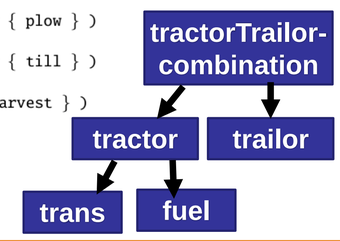
- aDSL objectives:
  - provide a formal language that enables unambiguous definition and reasoning
  - enable the definition of requirements that constraint operation modes to reason about **interoperability**
  - **automatically** evaluate design alternatives

An aDSL instance

The system designer uses the Eclipse IDE for modeling[4]:
1) A **CPS** comprises systems and parts, both with operation spaces
2) A **requirement** constraints these operation spaces

An aDSL system

```
Section system
 Top-level System tractorTrailorCombination OperationSpace () {
   System tractor OperationSpace () {
     System trans OperationSpace () {
      DesAlt (transmission){
        unsynchronized Part transUnsynchronized OperationSpace
            ( driverSkills {advanced moderate easy} continousOperation {no} gears [1 24] )
        doubleClutch Part transDoubleClutch OperationSpace
            ( driverSkills { moderate easy } gears[ 1 2 4])
        CVT Part transCVT OperationSpace
            ( driverSkills {easy} efficiency { frictionLoss } gears [ 1 1000 ] )
      }
     }
     System fuel OperationSpace () {
       DesAlt (engineFuel){
         steam Part fuelSteam OperationSpace
             (pollution [5 10] speed [0 30] )
         diesel Part fuelDiesel OperationSpace
             (pollution [4 8] fuelConsumption [3 5] speed [0 40] price { medium high } )
         gasoline Part fuelGasoline OperationSpace
             (pollution [2 5] fuelConsumption [4 10] speed [0 50] price { medium high })
         electric Part fuelElectric OperationSpace
             (pollution [0 4] fuelConsumption [8 12] speed [0 55] price { high } )
      }
     }
     DesAlt(trailor) {
        chiselPlow Part TrailorChiselPlow OperationSpace
          ( speed [0 25] agility { low veryLow } activity { plow } )
        trailorTiller Part TrailorTiller OperationSpace
          ( speed [0 25] agility { low veryLow } activity { till } )
        chaserBin Part ChaserBin OperationSpace
          ( speed [0 15] agility { veryLow } activity { harvest } )
        notrailor Part NoTrailor OperationSpace
          ( load [0 0] activity { none } )
    }
   }
```

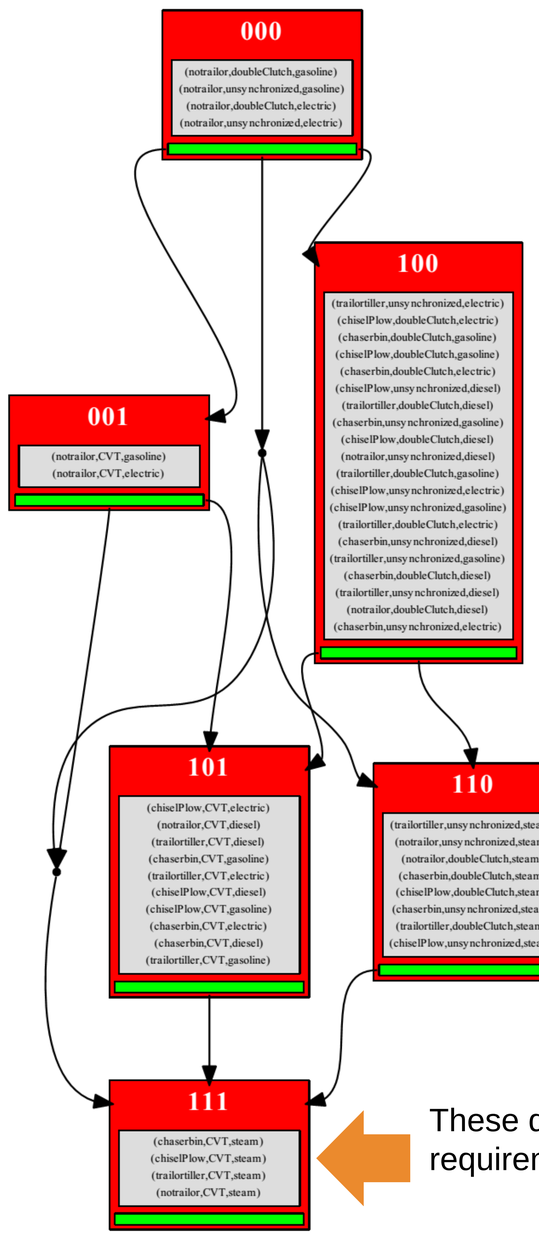


aDSL requirements

```
Section requirements
 Legal Requirement speedRange
   minimum OperationSpace ( speed [5 15] )
   maximum OperationSpace ( speed [0 45] )
 Business Requirement fuelAndPurchaseCosts
   maximum OperationSpace
      ( fuelConsumption [0 20] price {medium})
 Design Requirement operability
   maximum OperationSpace ( driverSkills easy )
```

Pareto optimal designs

We have satisfied the three requirements of aDSL, as mentioned in the objectives



aDSL determines, for each design, the requirements it meets and generates a partial ordering.

These designs meet requirement 1 and 2 but not 3

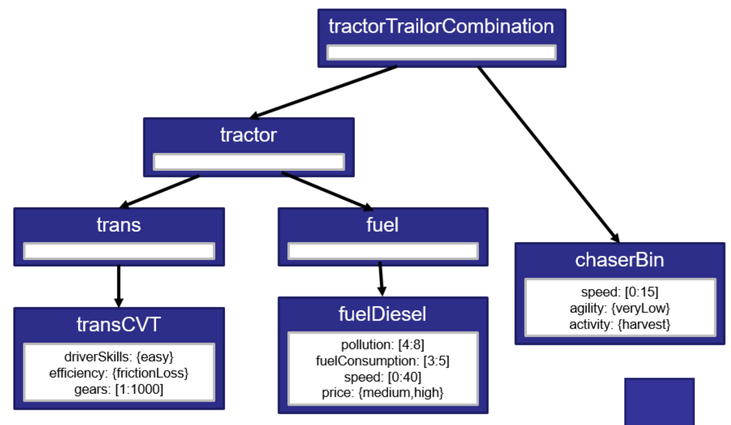
These designs meet all requirements

Evaluating operation spaces

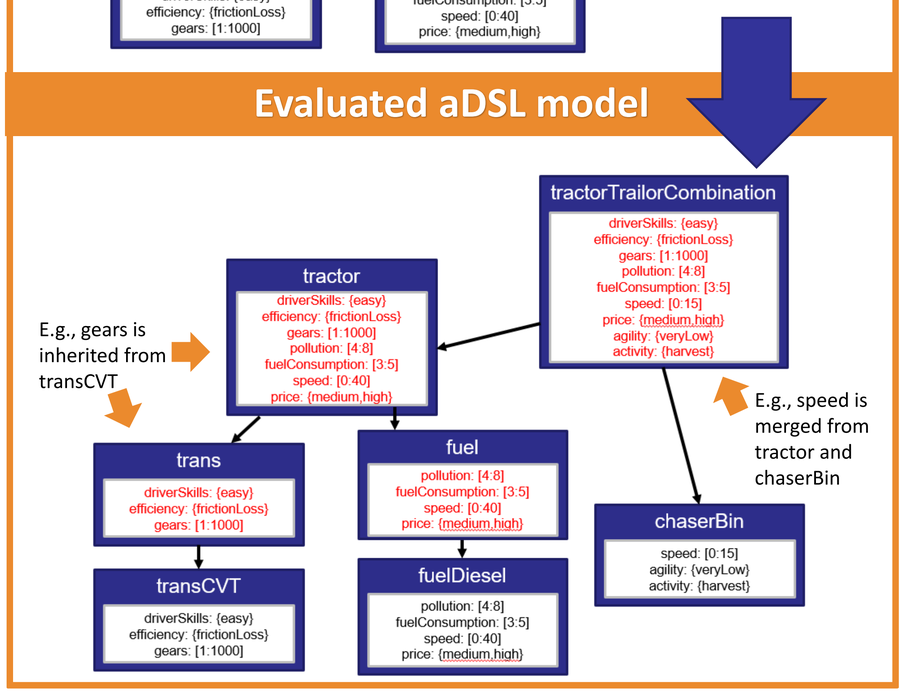
aDSL automatically derives the operation space of each subsystem in a **bottom up** fashion; the result depends on its operation space and the operation spaces of its children.

When operation spaces have overlapping dimensions, the intersection of their values is taken. Otherwise, the dimension and its values are simply copied to a higher level.

Unevaluated aDSL model



Evaluated aDSL model



IEEE SoSE 2018

Poster session

http://www.a-dsl.org

[tu] TU Delft Delft University of Technology

[tue] TU/e Technische Universiteit Eindhoven University of Technology

[ut] UNIVERSITY OF TWENTE.

[wur] WAGENINGEN UNIVERSITY & RESEARCH

[1] E. A. Lee, “Cyber-physical systems--Are computing foundations adequate?”, Position Paper for NSF Workshop on CPS, 2006.
[2] P. Miller, “Interoperability: What is it and why should I want it?” Ariadne, no. 24, 2000.
[3] F. van den Berg, V. Garousi, B. Tekinerdogan, and B.R. Haverkort, "Designing Cyber-Physical Systems with aDSL: a Domain-Specific Language and Tool Support", 13th System of Systems Engineering Conference, 2018
[4] Eclipse IDE for Java and DSL Developers, http://www.eclipse.org/downloads/packages/eclipse-ide-java-and-dsldevelopers/junosr2