MoDELCi

A Trial on

Model-Driven Approach for Electrical Circuit Modeling

CS587 Model-Driven Software Development

by
Hanifi Güneş

May'09
Bilkent University

Agenda

- Motivation
- Domain Analysis
- Meta-Modeling
- Model Transformations
  - Model2Text, Model2XML
  - Model2Model, ECircuit2ECircuit
- Final Thoughts and Lessons Learned
- Demonstration

Motivation

- Quite hard stuff to implement complicated electronic circuits
  - Low-level design procedures:
    - Slash down **productivity** and **reusability**
    - Multiply the cost and invested resources (time, labor force etc...)
  - Plenty of divergent tools out there!
    - Not interoperable! Decreases portability and requires too much effort to grasp either of them!

Motivation

- Economical aspects
- MDA over Electronics comrises a huge niche
  - It requires a multi/inter-disciplinary approach.
  - This project is among very first efforts.
- And last but not least, I am still an Electrical and Electronics Engineer!!!

So…

- How to address productivity, reusability, interoperability and portability?
- MDA can better our hand! ☺

Domain Analysis

- An **Electronic circuit** is a **closed path** formed by the **interconnection** of **electronic components** through which an electric current can flow. [1]

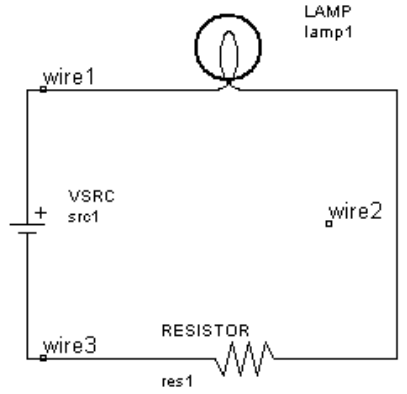
- An **electrical network** is an **interconnection of electrical elements** such as resistors, inductors, capacitors, transmission lines, voltage sources, current sources, and switches. [2]

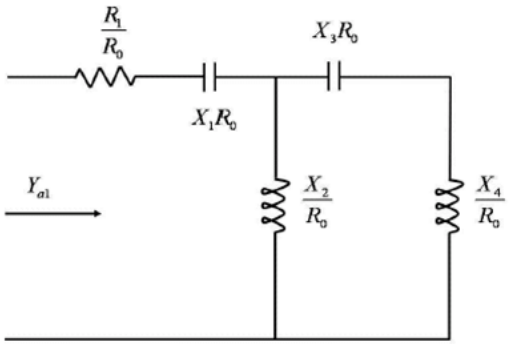
- An **electrical circuit** is a **network** that has a closed loop, giving a return path for the current. A **network** is a **connection of two or more components**, and may not necessarily be a circuit. [2]

- A **network** that also contains **active electronic components** is known as an electronic circuit. [2]

Extracting terms

- Circuit: A closed loop combined by electronic components.
- Network: Connection of two or more electronic elements not having an active component!
- Electronic component: Basic electronic element with two or more connecting leads.
- Active comp: Have gain and/or directionality e.g. semiconductor devices.[3]
- Passive comp: Have neither! Electrical elements [3]

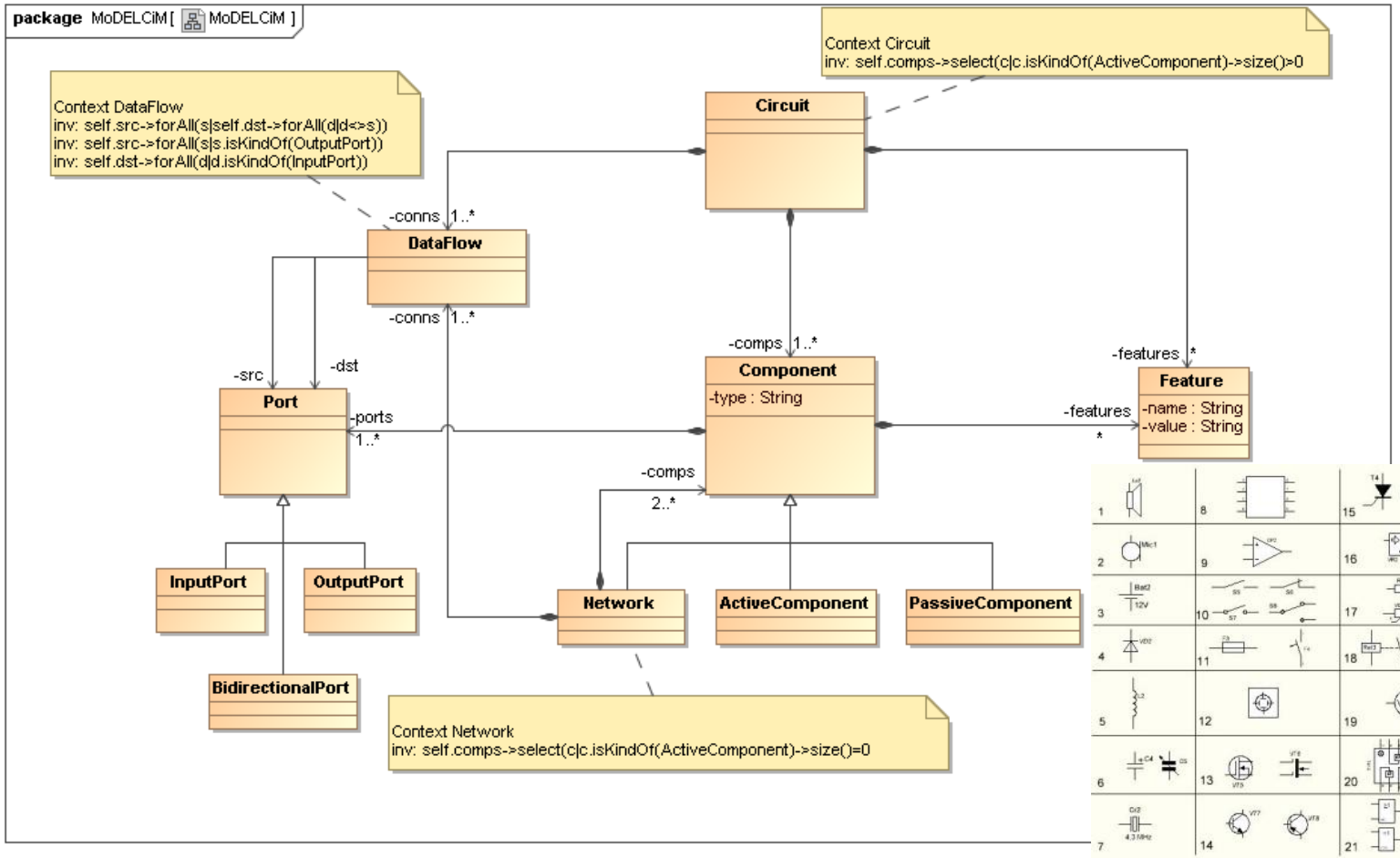




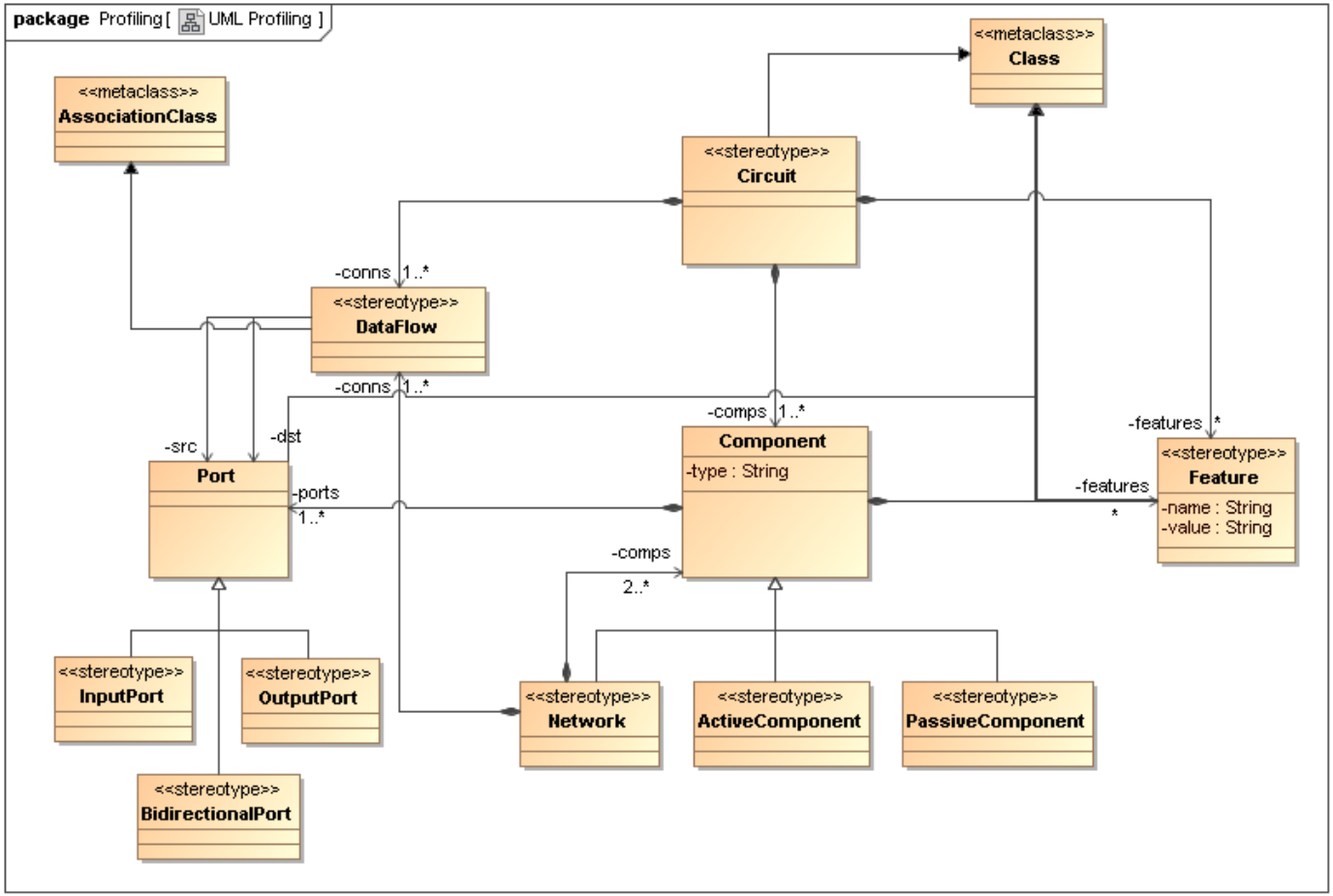
Mapping concepts to grammar

- Circuit :: Components Connection
- Components :: {Component}[1..*]
- Connection :: {DataFlow}[1..*]
- Component :: Network | ActiveComp | PassiveComp
- Network:: PassiveComponents Connection
- PassiveComponents :: {PassiveComp}[2..*]
- Connection :: {DataFlow}[1..*]
- DataFlow :: SourcePort DestinationPort
- SourcePort :: Port
- DestinationPort :: Port
- Port :: Markers| InputPort | OutputPort
- Markers :: {Marker}*
- Marker :: VoltageMarker | CurrentMarker | GainMarker

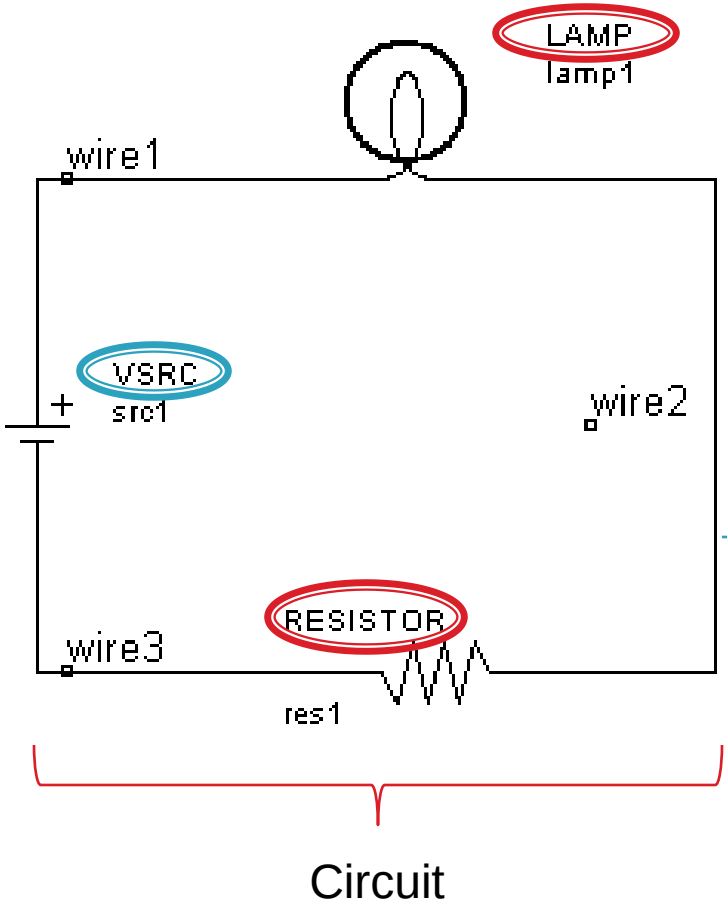
Meta-model built upon MOF



Meta-model using UML Profiling

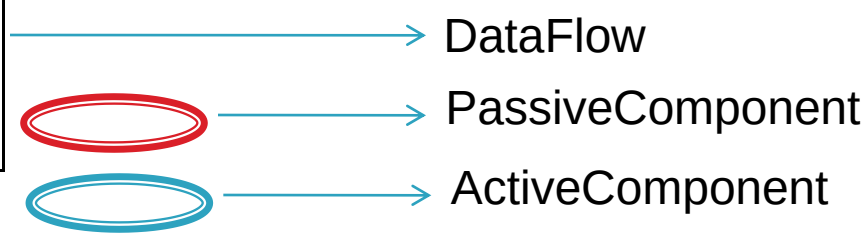


Take an example model...

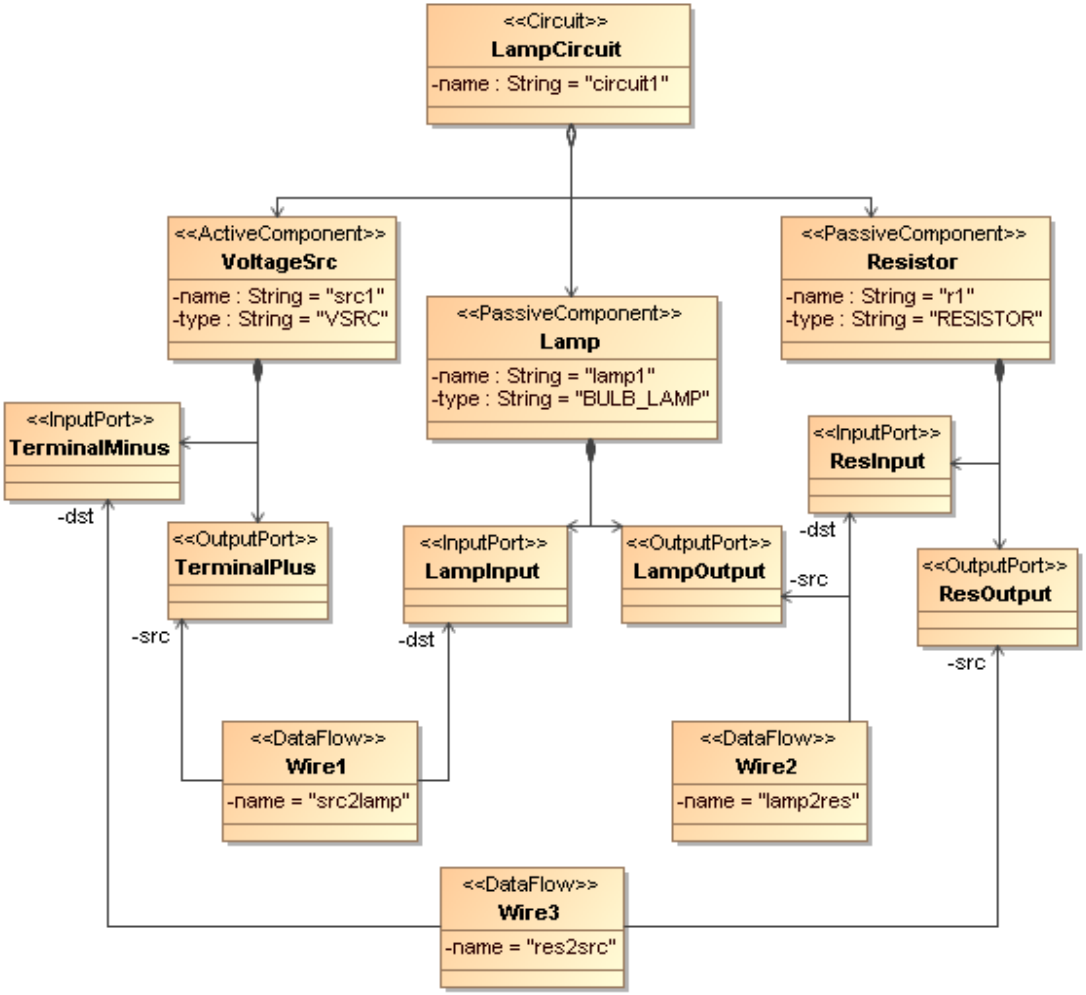


Each component:

- ✓ has a unique name
- ✓ has two ports(in, out)
- ✓ connected to its adjacent via a wire



Model using profiling

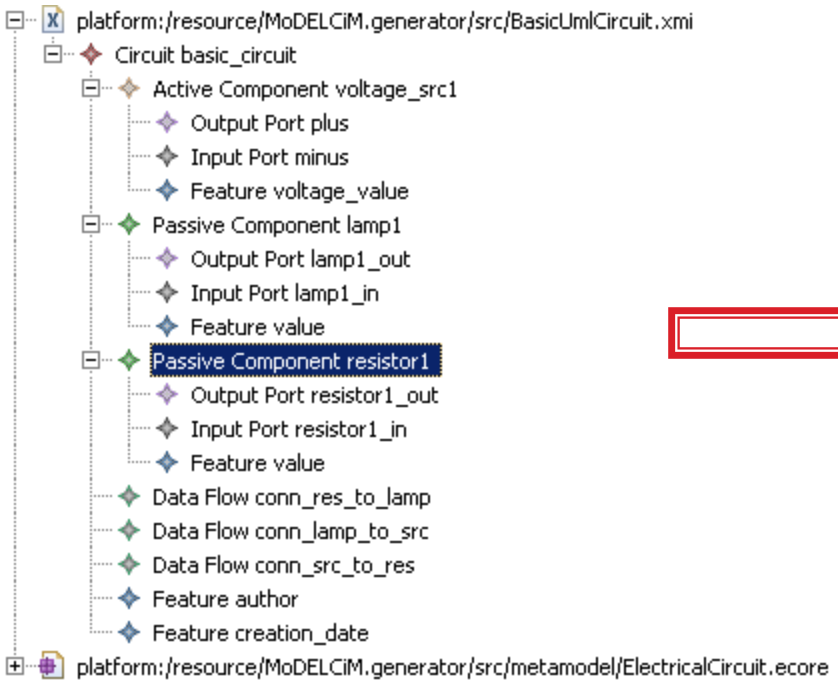


Model Transformations

- Why?
  - To assure portability, interoperability and reusability
- Why XML?
  - It is a powerful standard right away
  - Highly extensible and has a stable meta-model supported by narrowly all set of modeling tools
  - Readable and easy-to-update

M2XML

- Will be demonstrated soon but



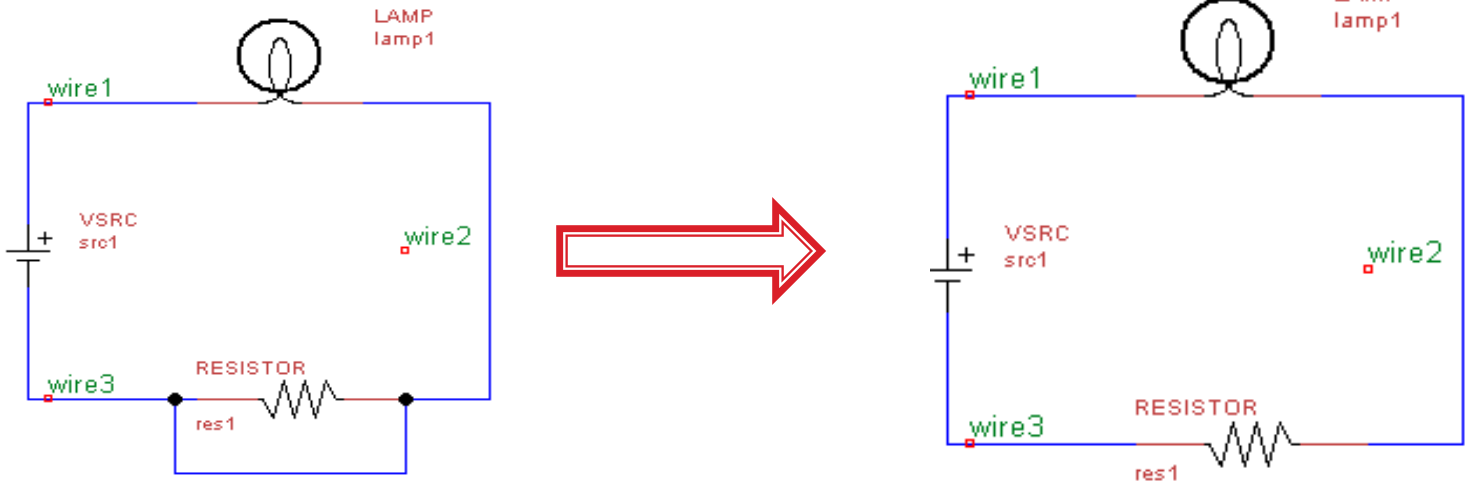

```
<?xml version="1.0" encoding="UTF-8"?>
<circuit name="network_circuit">
  <properties>
    <property name="author" value="Hanifi Gunes"/>
    <property name="creation_date" value="15/05/2009 16:38:14"/>
  </properties>
  <components>
    <component name="voltage_src1">
      <properties>
        <property name="voltage_value" value="10"/>
      </properties>
      <ports>
        <port type="out" name="plus"/>
        <port type="in" name="minus"/>
      </ports>
    </component>
    <component name="lamp1">
      <properties>
        <property name="value" value="10K"/>
      </properties>
      <ports>
        <port type="out" name="lamp1_out"/>
        <port type="in" name="lamp1_in"/>
      </ports>
    </component>
```

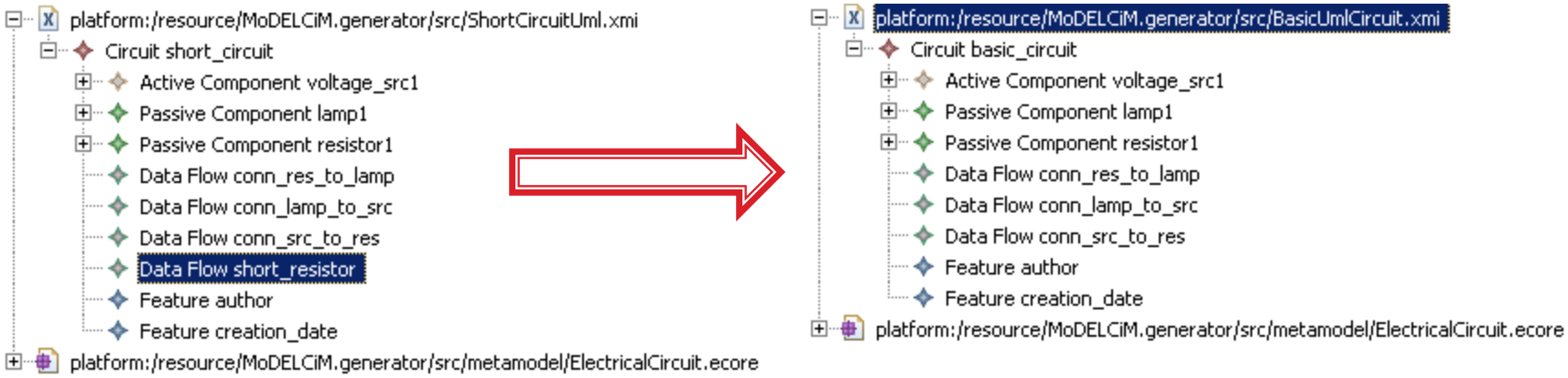
Model2Model

- Why?
  - To better a model, refine it

- An example that aligns a real-world problem with MDA can be refining an existent circuit
- Refining = simplifying
- A criteria based simplification may be of beneficial
- Reduces cost and automate circuit modeling

Short circuit remover





Technically speaking

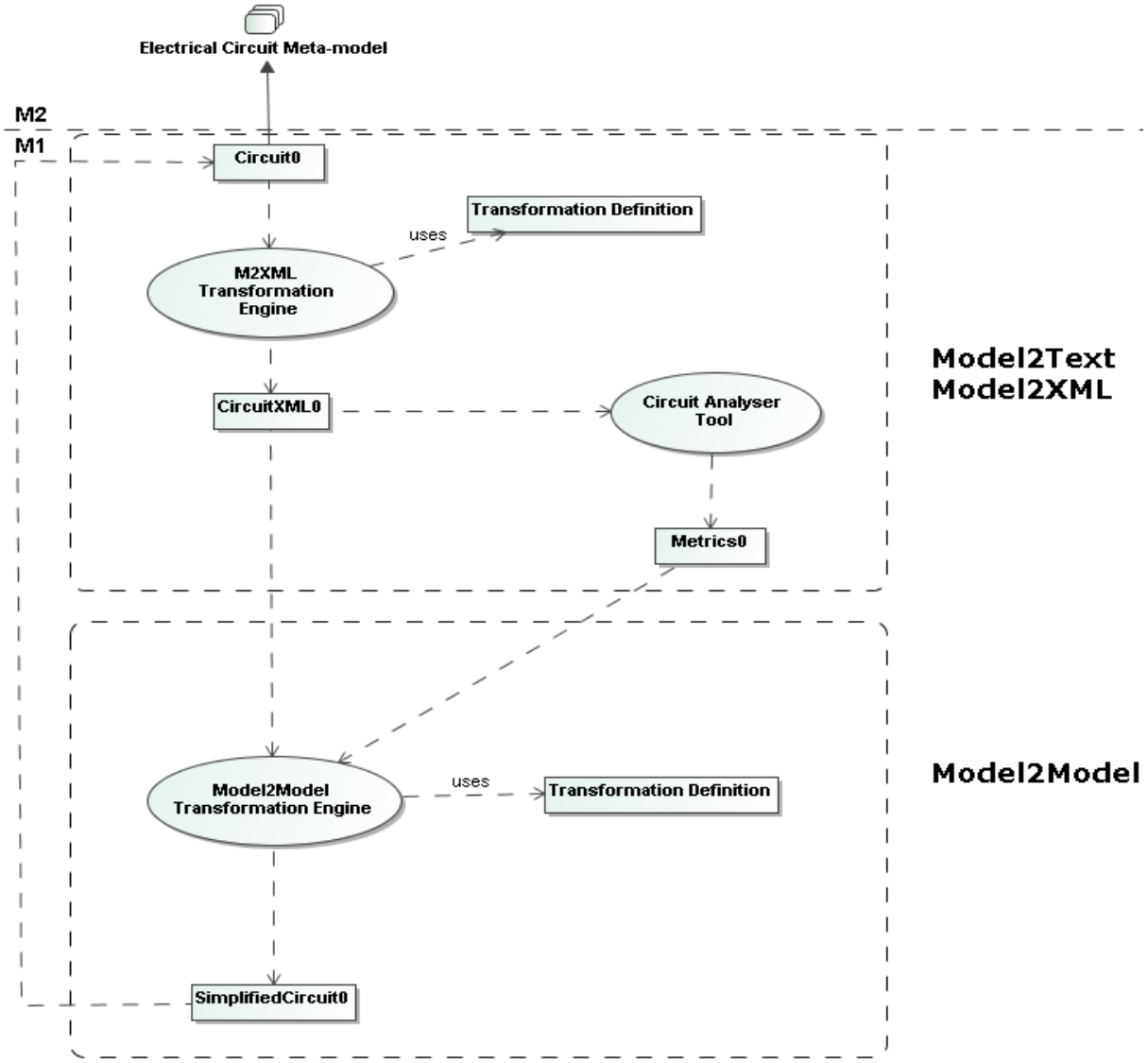
```
module CircuitSimplificator;
create OUT : ElectricalCircuit refining IN : ElectricalCircuit;
helper context Sequence(ElectricalCircuit!DataFlow) def : getSimplifiedSeq :
    Sequence (ElectricalCircuit!DataFlow) =
    self -> select(c|
        c.src.substring(1, c.src.lastIndexOf('/'))
        <> c.dst.substring(1, c.dst.lastIndexOf('/')));
rule Circuit {
    from
        s : ElectricalCircuit!Circuit
    to
        t : ElectricalCircuit!Circuit (
            name <- s.name,
            features <- s.features,
            comps <- s.comps,
            conns <- s.conns.getSimplifiedSeq()
        )
}
rule Network {
    from
        s : ElectricalCircuit!Network
    to
        t : ElectricalCircuit!Network (
            name <- s.name,
            features <- s.features,
            comps <- s.comps,
            conns <- s.conns.getSimplifiedSeq()
        )
}
```

A detailing question

- Can we fully automate the circuit simplification process soon?

- With the help of MDA, it seems like possible☺
  - However requires sufficient and capable tooling!

How to automate?



Final thoughts

- Well seperation of concerns and thus levels are crucial.
- Electronic circuits are good examples that of having well discriminated concerns
- Be as generic as possible while defining a meta-model.
- Meta modeling seems to be quite helpful in electronics engineering where we focus on systems rather than models!

Final thoughts

- Model transformations expedite the way to re/produce models and thus, are of paramount importance
- In this case, increases productivity, portability, interoperability, reusability dramatically
- Thanks to MDA, circuit generation can be fully automated soon
- Current MDSD tools still leave significant room for improvement!

Demonstration

- Now to the live examples

Thanks and questions?

References

- [1] http://en.wikipedia.org/wiki/Electronic_circuit
- [2] http://en.wikipedia.org/wiki/Electrical_network
- [3] Khalil, Hassan (2001). *Nonlinear Systems (3rd Edition)*. Prentice Hall. ISBN 0130673897.
- J. White, Douglas C. Schmidt, A. Nechypurenko and E. Wuchner, “Domain-Specific Intelligence Frameworks for Assisting Modelers in Combinatorically Challenging Domains”