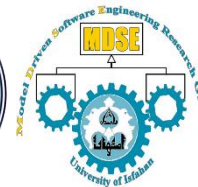# SOLVING THE STATE ELIMINATION CASE STUDY USING EPSILON

Model-Driven Software Engineering Research Group
Department of Software Engineering
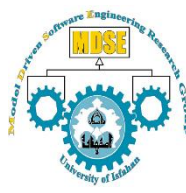University of Isfahan

Mohammadreza Sharbaf
m.sharbaf@eng.ui.ac.ir
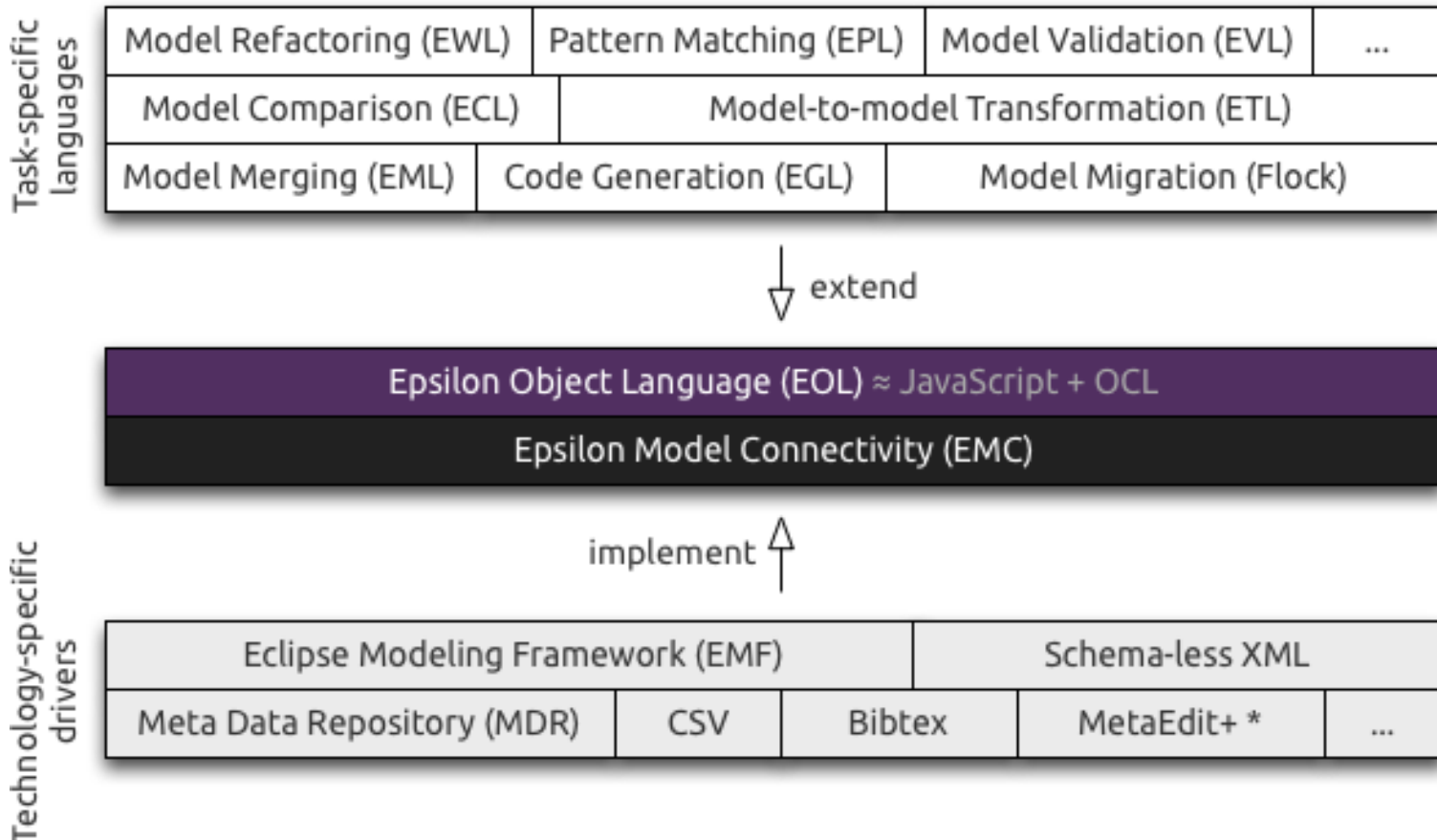**Shekoufeh Kolahdouz-Rahimi**
sh.rahimi@eng.ui.ac.ir
Bahman Zamani
zamani@eng.ui.ac.ir

EPSILON OVERVIEW

- A family of integrated programming languages for managing models.

Task-specific languages	Model Refactoring (EWL)	Pattern Matching (EPL)	Model Validation (EVL)	...
	Model Comparison (ECL)	Model-to-model Transformation (ETL)		
	Model Merging (EML)	Code Generation (EGL)	Model Migration (Flock)	

↓ extend

Epsilon Object Language (EOL) ≈ JavaScript + OCL

Epsilon Model Connectivity (EMC)

implement ↑

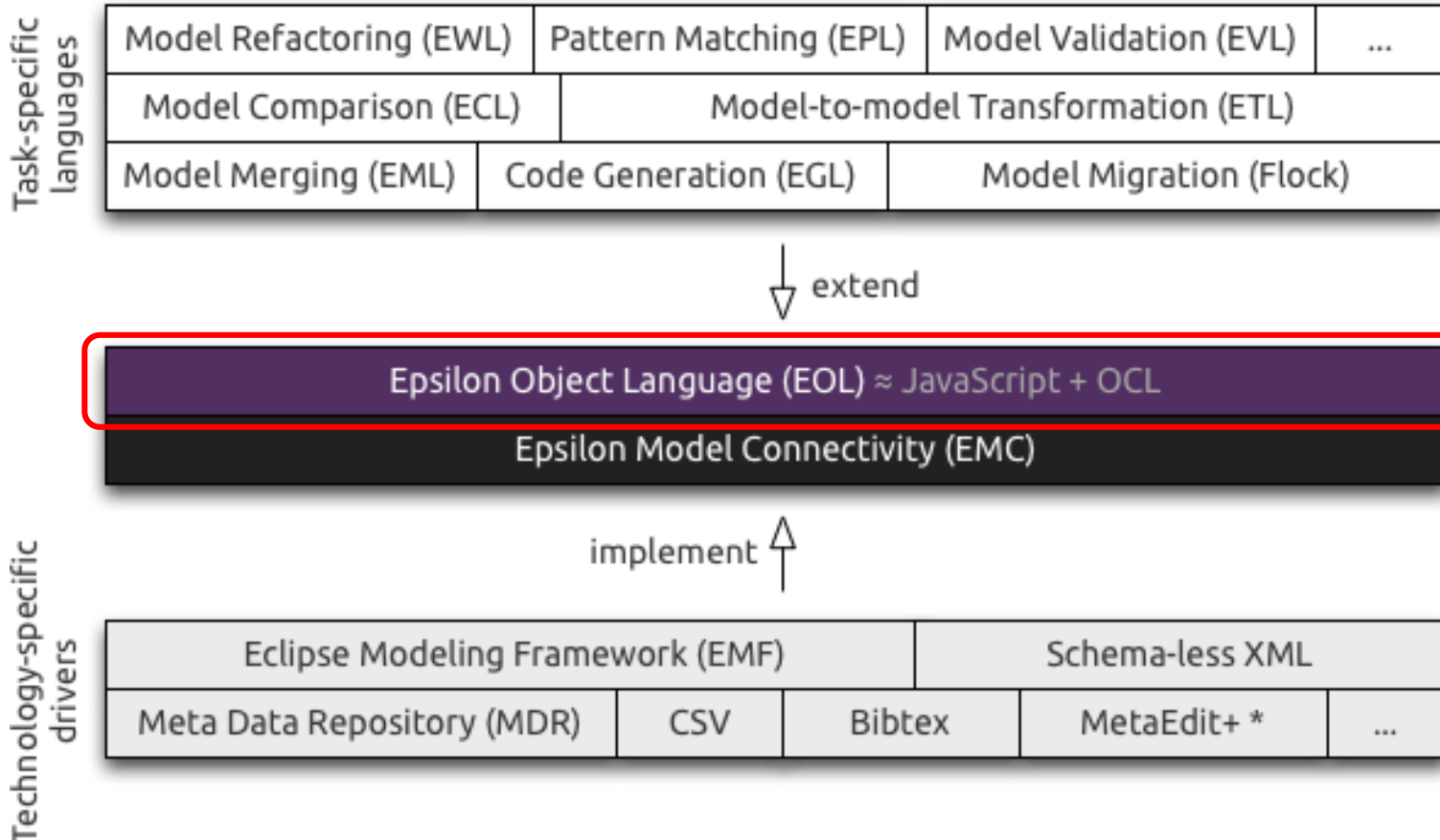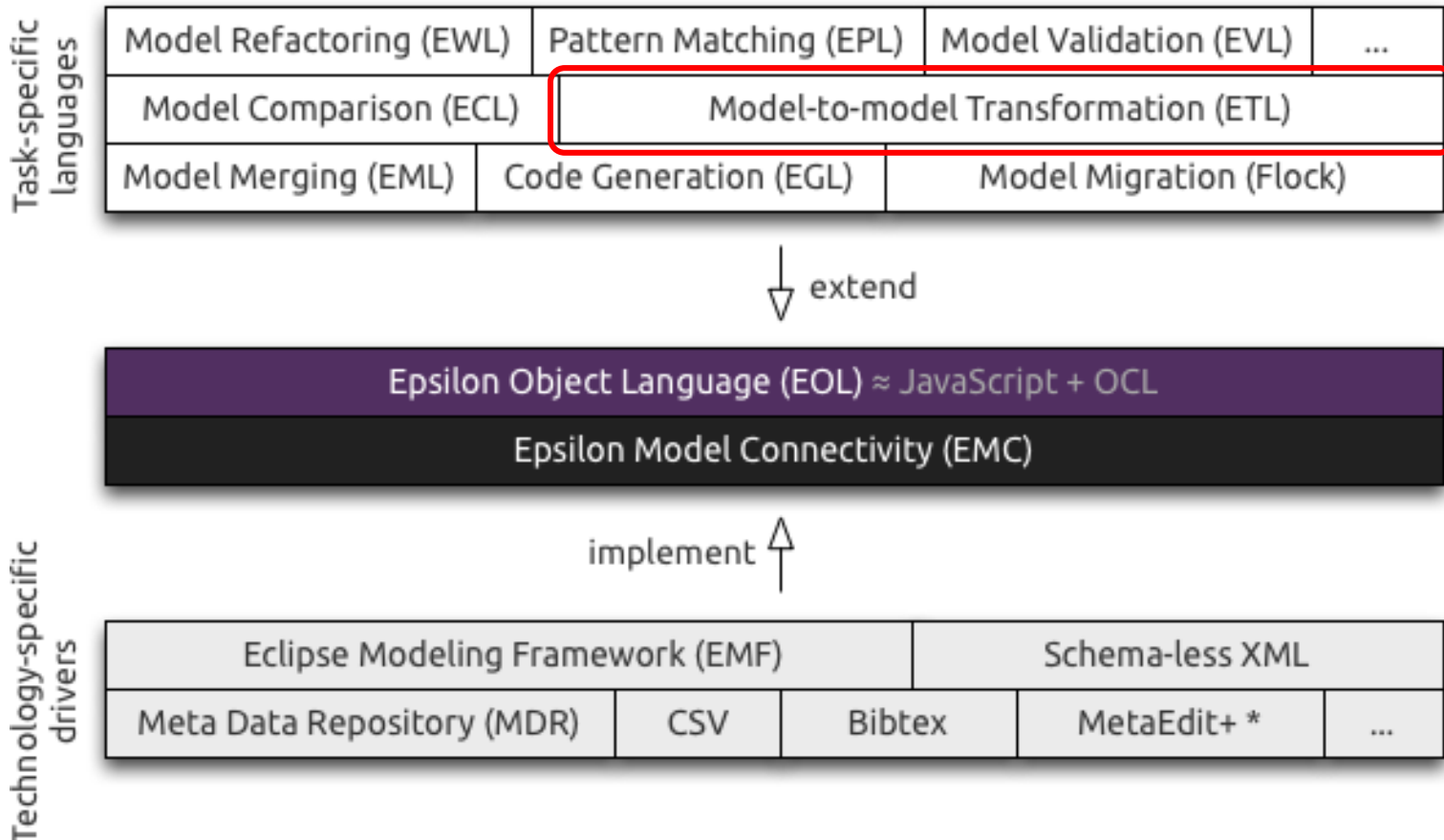| Technology-specific drivers	Eclipse Modeling Framework (EMF)		Schema-less XML	
	Meta Data Repository (MDR)	CSV	Bibtex	MetaEdit+ *	...

EPSILON OVERVIEW

- A family of integrated programming languages for managing models.

EPSILON OVERVIEW

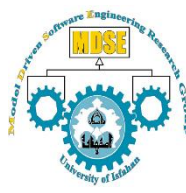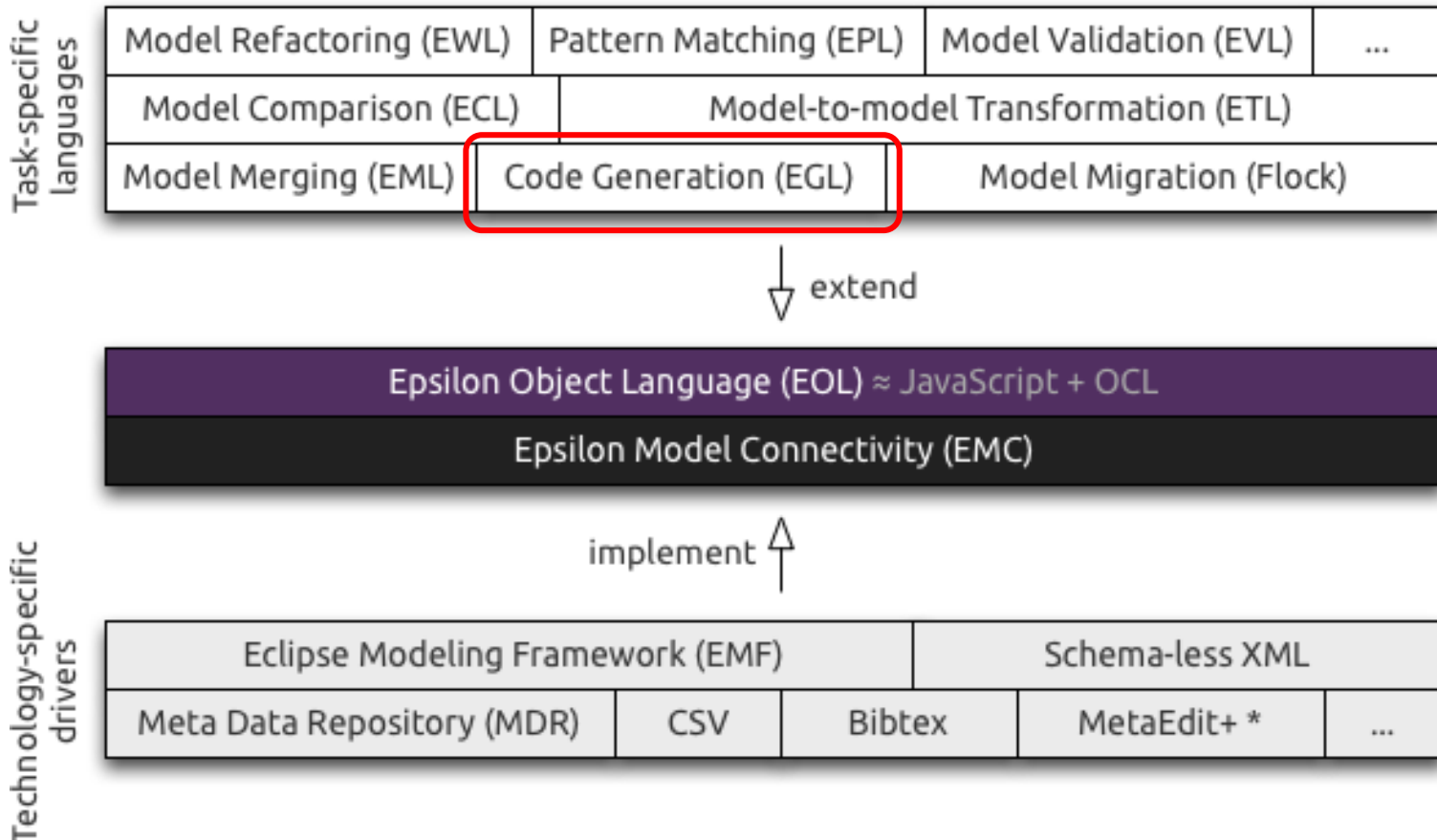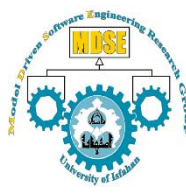- A family of integrated programming languages for managing models.

EPSILON OVERVIEW

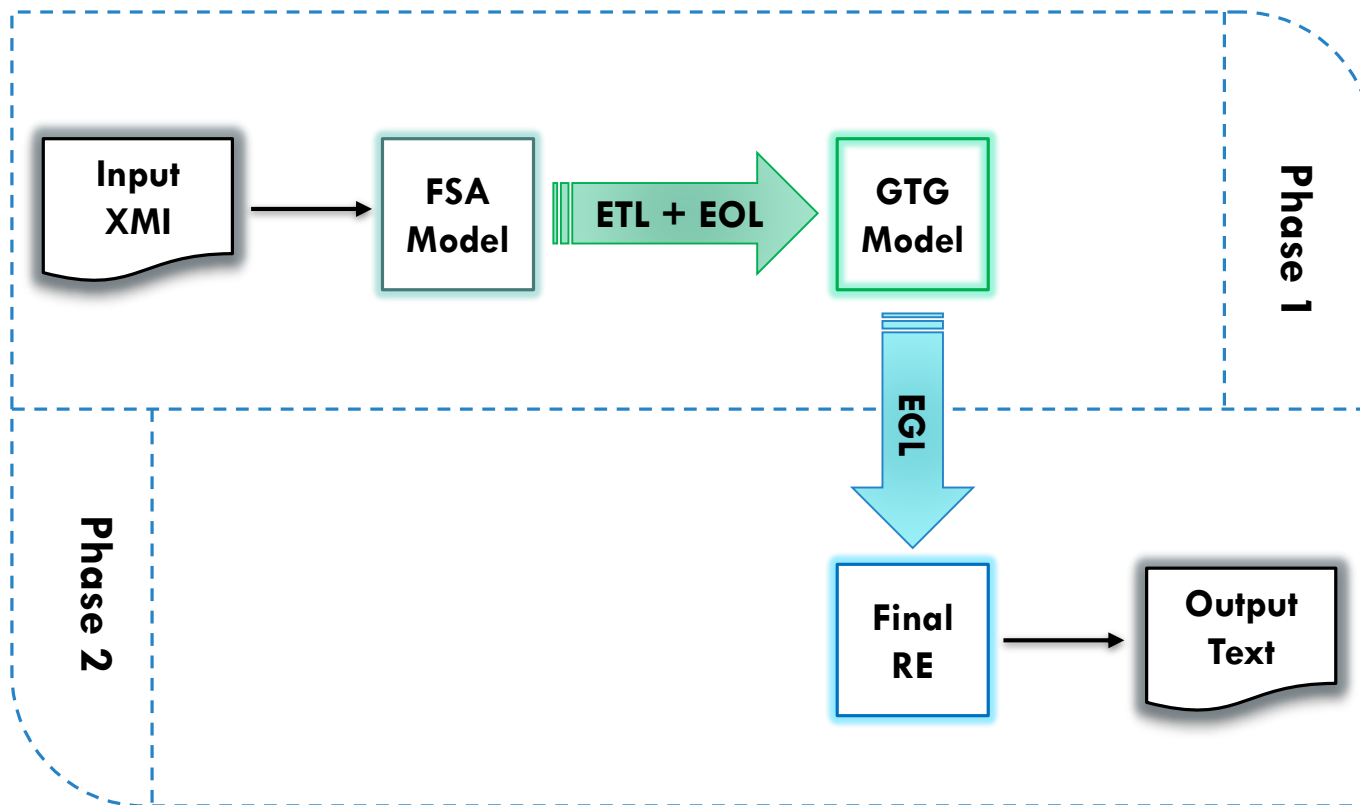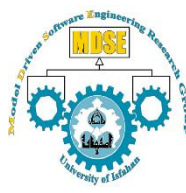- A family of integrated programming languages for managing models.

SOLUTION OVERVIEW - MAIN TASK

https://github.com/MSharbaf/TTC2017-StateElimination

SOLUTION OVERVIEW - MAIN TASK: IMPLEMENTATION

- Phase 1: FSA to GTG

  - Step 1: Creation of GTG

FSA Model	ETL + EOL	GTG Model

```
rule StateAddition
    transform S1: input!State
    to S2: output!State{
    guard : S1.isInitial or S1.isFinal

    S2.id = S1.id ;
    S2.isInitial = S1.isInitial ;
    S2.isFinal = S1.isFinal ;
    if(S1.isInitial){
        trans.source = S2 ;
        S2.outgoing ::= trans ;
    }
    if(S1.isFinal){
        trans.target = S2 ;
        S2.incoming ::= trans ;
    }
}
```
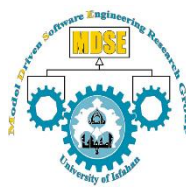
- Phase 1: FSA to GTG

  - Step 1: Creation of GTG

  - Step 2: Deletion of Intermediate states and transitions


```
var flag_lbl_SL = false ;
var SL_Tr = SIncoming.select(tr|tr.source==K and tr.target==K);
while(SL_Tr.size() > 0){
    var tr = SL_Tr.first() ;
    if(tr.label.isEmpty()==false){
        if(flag_lbl_SL==false){
            lbl_SL += "(" ;
            flag_lbl_SL = true ;
        }else
            lbl_SL += "+" ;
        lbl_SL += tr.label ;
    }
    delete tr ;
    SL_Tr.remove(tr)  ;
}
if(lbl_SL.isEmpty()==false)
    lbl_SL += ")*" ;
```
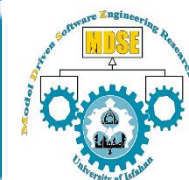
SOLUTION OVERVIEW - MAIN TASK: IMPLEMENTATION

- Phase 1: FSA to GTG

  - Step 1: Creation of GTG

  - Step 2: Deletion of Intermediate states and transitions

  - Step 3: Labelling the unlabeled transition in the GTG



```
label += lbl_SSL ;
if(not lbl_SF.isEmpty()){
    label += "(" ;
    label += lbl_SF ;
    label += ")" ;
}
if(not lbl_FSF.isEmpty() or not lbl_FSL.isEmpty()){
    label += "(" ;
    label += lbl_FSF ;
    if(not lbl_FSF.isEmpty())
        label += "+" ;
    label += lbl_FSL ;
    label += ")*" ;
}
trans.label = label ;
```

SOLUTION OVERVIEW - MAIN TASK: IMPLEMENTATION

- **Phase 1: FSA to GTG**

  - Step 1: Creation of GTG

  - Step 2: Deletion of Intermediate states and transitions

  - Step 3: Labelling the unlabeled transition in the GTG
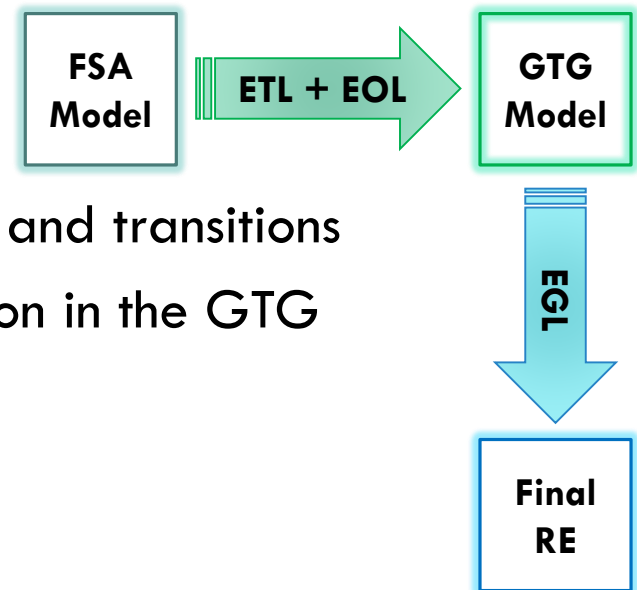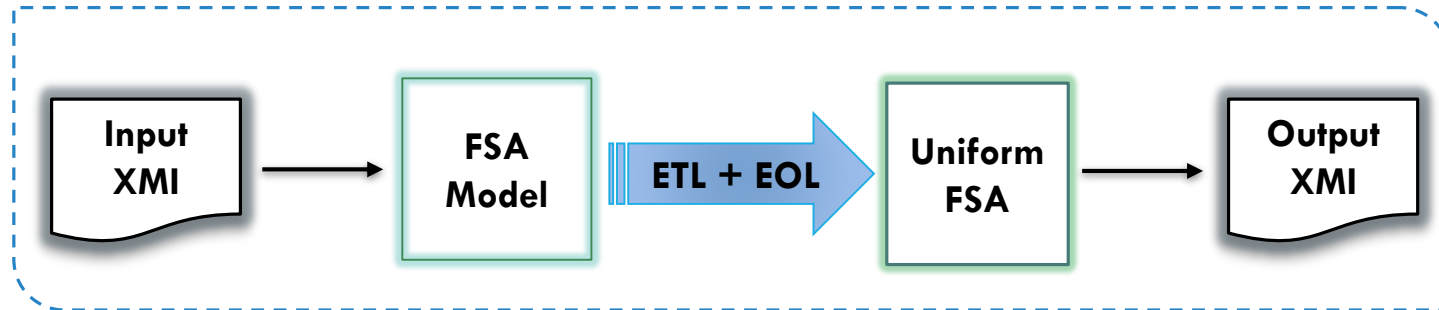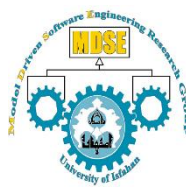
- **Phase 2: GTG to RE**

```
[%
  var sb := new Native("java.lang.StringBuilder");
  sb.append(Transition.allInstances.selectOne
    (s|s.label.isDefined()).label) ;
%][%=sb.toString()%]
```
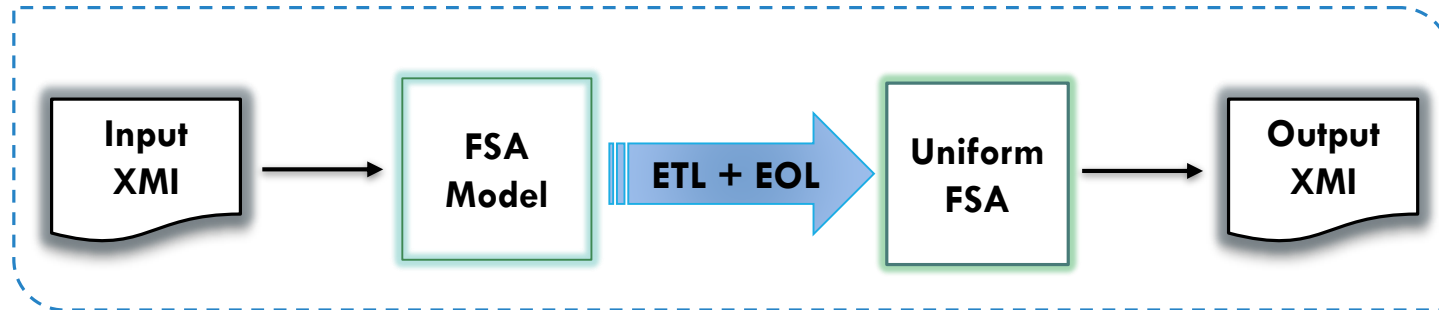
FSA Model → **ETL + EOL** → GTG Model

EGL

Final RE

SOLUTION OVERVIEW - EXTENSION 1

SOLUTION OVERVIEW – EXTENSION 1


```
pre{
    Uniforming() ;
}
rule TransitionGraph
    transform S1: input!TransitionGraph
    to S2: output!TransitionGraph{
        S2.states ::= S1.states ;
        S2.transitions ::= S1.transitions ;
}
operation Uniforming()
{
    var States = input!State.all().asSequence() ;
    var SS = States.select(st|st.isInitial==true);
    if(SS.size()>1)
    {
        var NStart : new input!State ;
        TG.states.add(NStart) ;
```
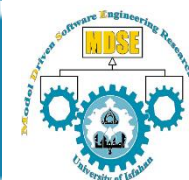
RESULTS

Model name (number of states)	Correct	Execution Time (s)	Scalability
leader3_2 (26)	yes	0.1476	
leader4_2 (61)	yes	0.1617	
leader3_3 (69)	yes	0.1769	
leader5_2 (141)	yes	0.2252	
leader3_4 (147)	yes	0.2398	
leader3_5 (273)	yes	0.3877	
leader4_3 (274)	yes	0.3639	
leader6_2 (335)	yes	0.4528	
leader3_6 (459)	yes	0.71	
leader4_4 (812)	yes	1	
leader5_3 (1050)	yes	2	
leader3_8 (1059)	yes	2	
leader4_5 (1933)	yes	7	
leader6_3 (3759)	yes	23	
leader4_6 (3962)	yes	29	
leader5_4 (4244)	yes	31	
leader5_5 (12709)	yes	341.6	
leader6_4 (20884)	yes	920.8	
leader6_5 (78784)	yes	9834.34	Leader6_5

All runs on java version "1.8.0", Windows 7 PC, 3,6 GHz Intel Core i7
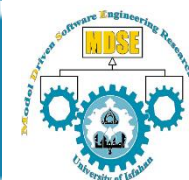
ABSTRACTION LEVEL

- Level of abstraction of transformation language:
  - EOL is Low
  - ETL is High
  - EGL is High

Element		Abstraction level
Main Task		
Phase 1	Transformation of a Uniform FSA to a GTG	Medium
Phase 2	Transformation of a GTG to a final regular expression	High
Overall solution		Medium
Extension 1		
Overall solution		Medium

CONCLUSION

- An Epsilon-based solution to the State Elimination case

- Uses a transformation chain comprises:
  - Model-to-Model transformation
  - Model-to-text transformation

- A scalable solution with reasonably good performance

https://github.com/MSharbaf/TTC2017-StateElimination