CS793 Project: Code Clones Visual Representation

Yu Sun, Frédéric Jouault

{yu.sun.cs, f.jouault}@gmail.com

12/5/2007

Goal

- Realize an uniform visual representation for analysis results of different code clone detection tools by using Model-Driven Engineering

Sub-Objectives & Main Steps

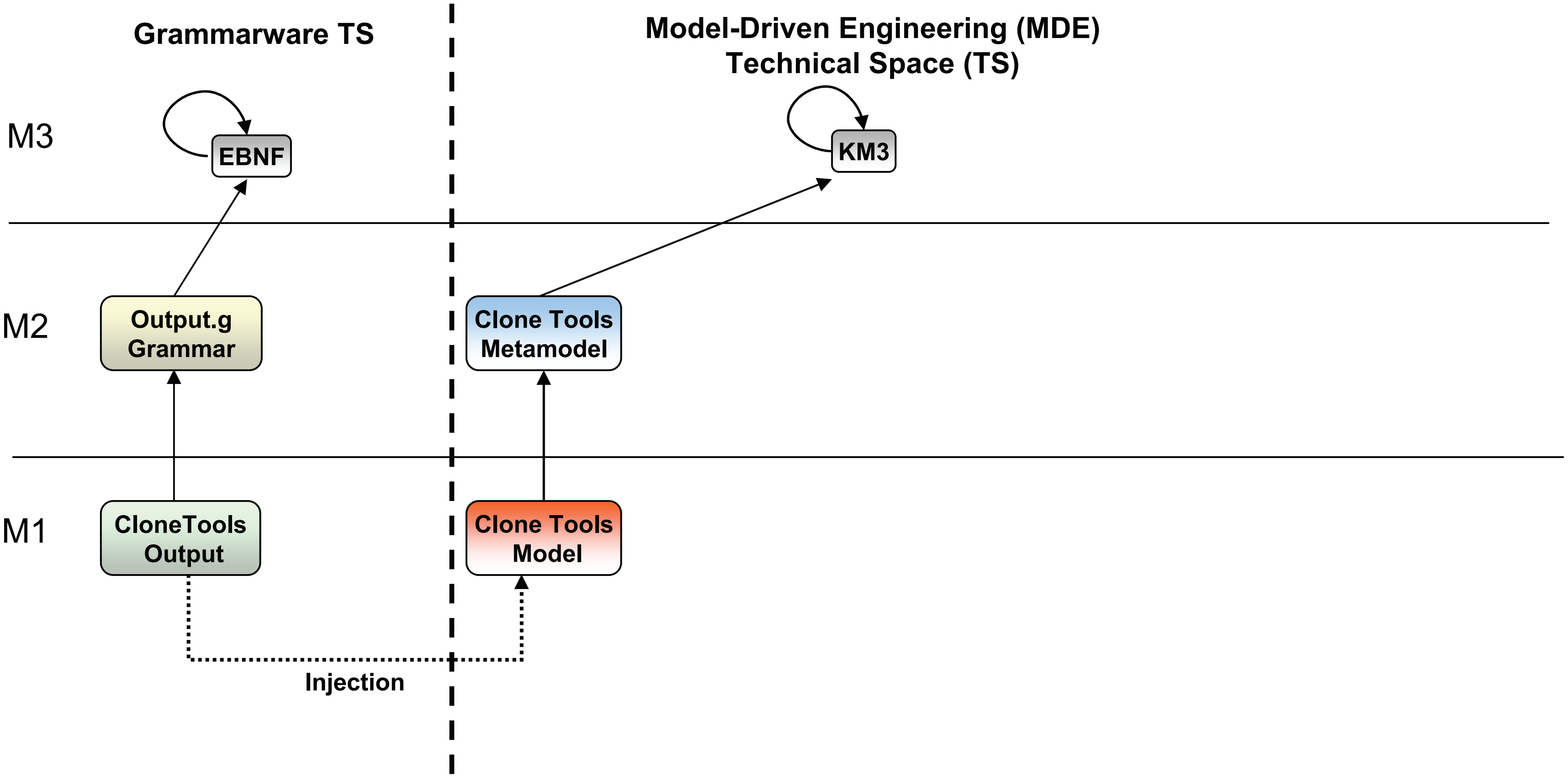
- Define specific DSLs for various kinds of code clone analysis results generated from different code clone detection tools, such as CCFinder, ClondDr, Simian, SimScan etc.
- Define a generic DSL to represent code clones
- Realize the transformation from specific code clone DSLs to the generic DSL
- Transform the generic code clone DSL to Scalable Vector Graphics (SVG)

Step 1 - Preparation

- Try some code clone detection tools to better understand this domain
- Read a tutorial about SVG
- Finish the project proposal

Step 2 – a DSL for a simple tool

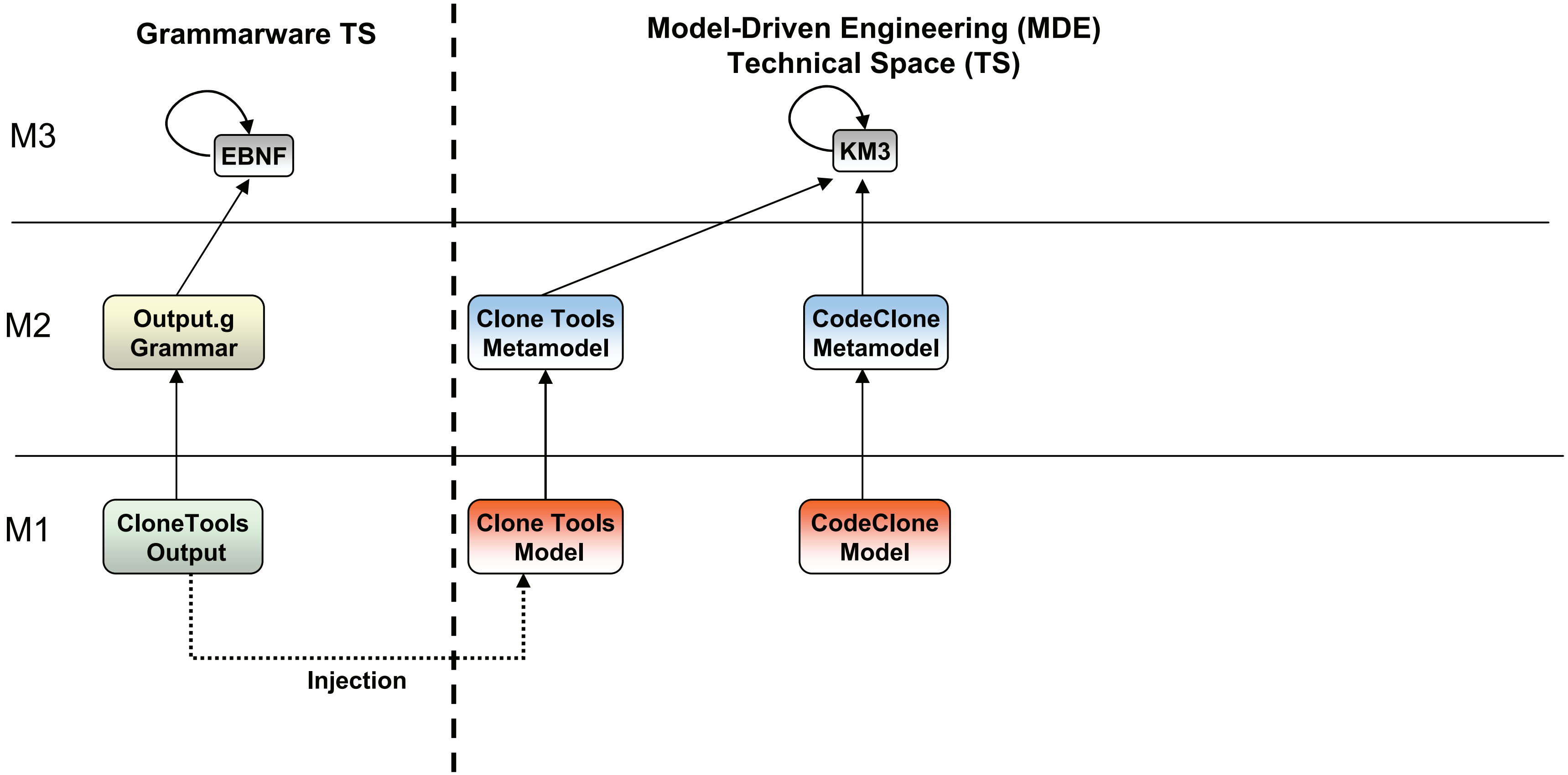
- Choose to first implement Simian
- Understand the result
- Define its KM3 Metamodel
- Define its TCS syntax
- Test Samples



M1 = terminal model level
M2 = metamodel level
M3 = meta-metamodel level

Step 3 – Define generic Code Clone DSL

- Analyze the Simian and other tools again, get the generic concepts for the DSL
- Define its KM3 metamodel



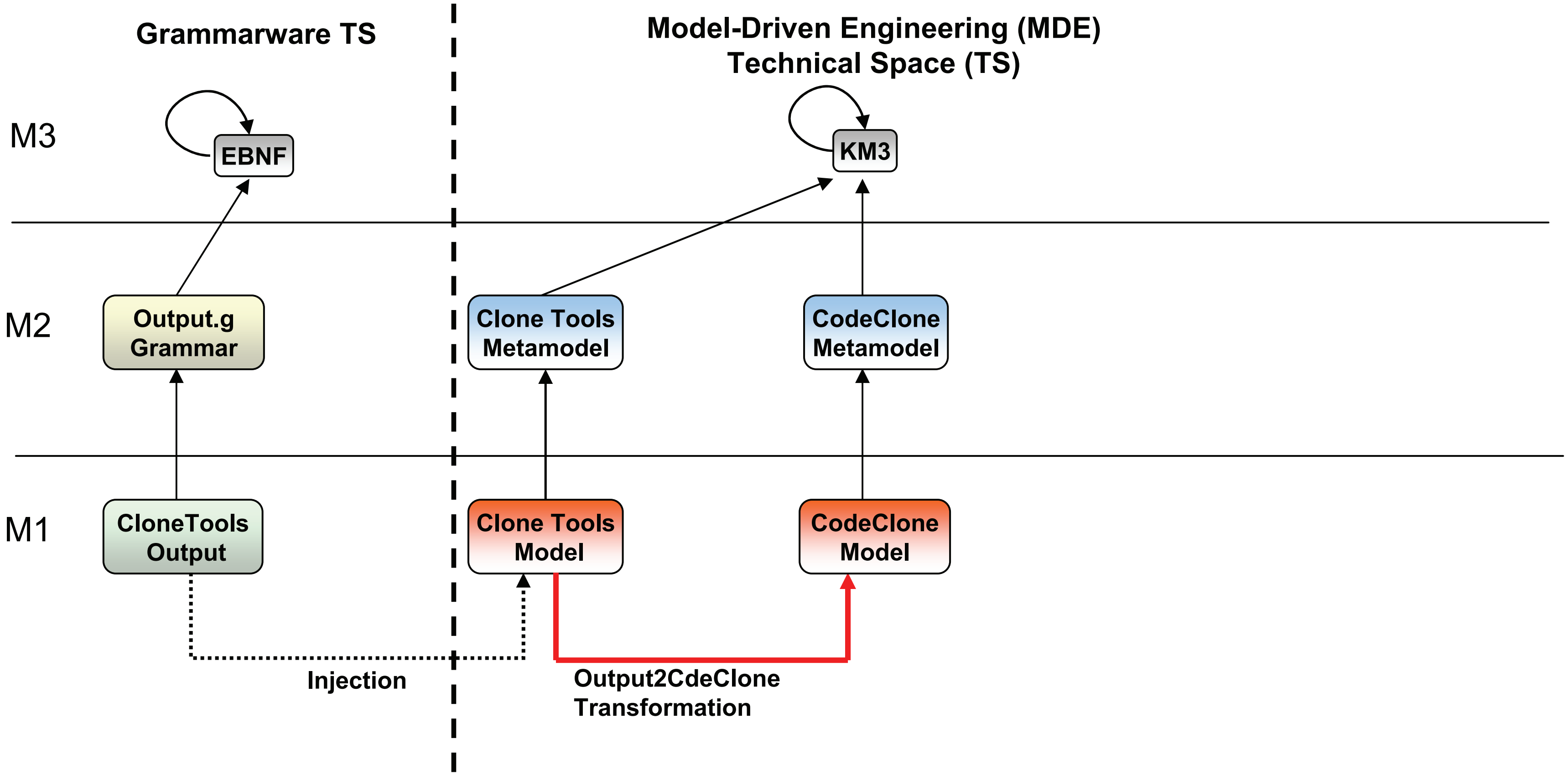
M1 = terminal model level

M2 = metamodel level

M3 = meta-metamodel level

Step 4 – Transformation to generic DSL

- Most of the mappings are very direct
- Getting familiar with ATL and its debug



M1 = terminal model level

M2 = metamodel level

M3 = meta-metamodel level

Step 5 – Understand later part transformation

- Referred to two samples
  - Table to SVGPieChart
  - Table to SVGBarChart

- http://www.eclipse.org/m2m/atl/atlTransformations/

Step 6 – Learning SVG and XML

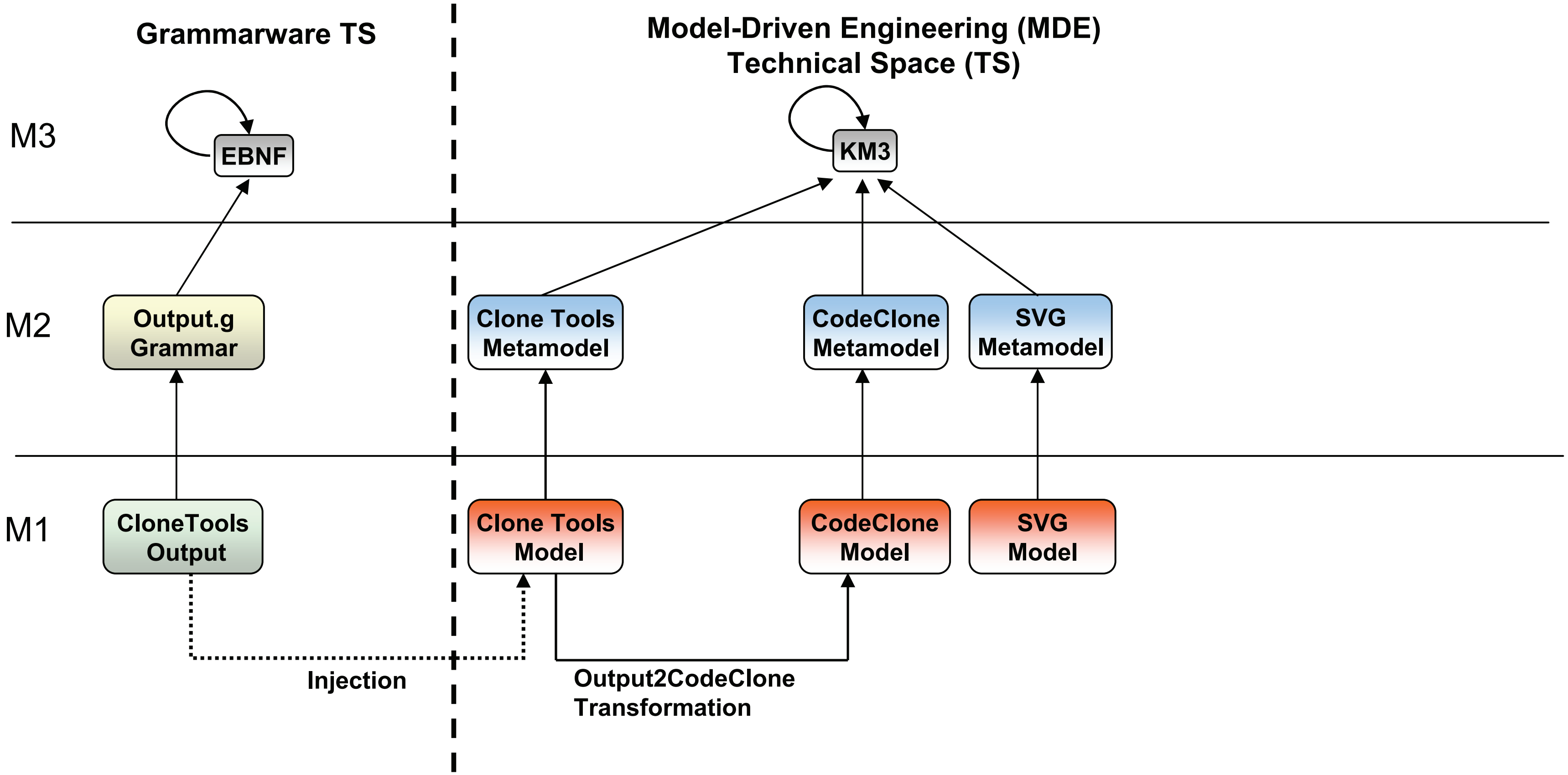
- Introduction to XML *Doug Tidwell IBM*
- Some online tutorials about SVG and examples

Step 7 – Design my SVG representation for code clone

- Try to make the representation concise and clear
- Write the SVG code for a specific clone group
- Calculate the location for each element

Step 8 – Define SVG

- Define SVG KM3 Metamodel
- An already defined metamodel in AM3 Zoo
- But it is too big, not necessary
- Define SVG metamodel by myself



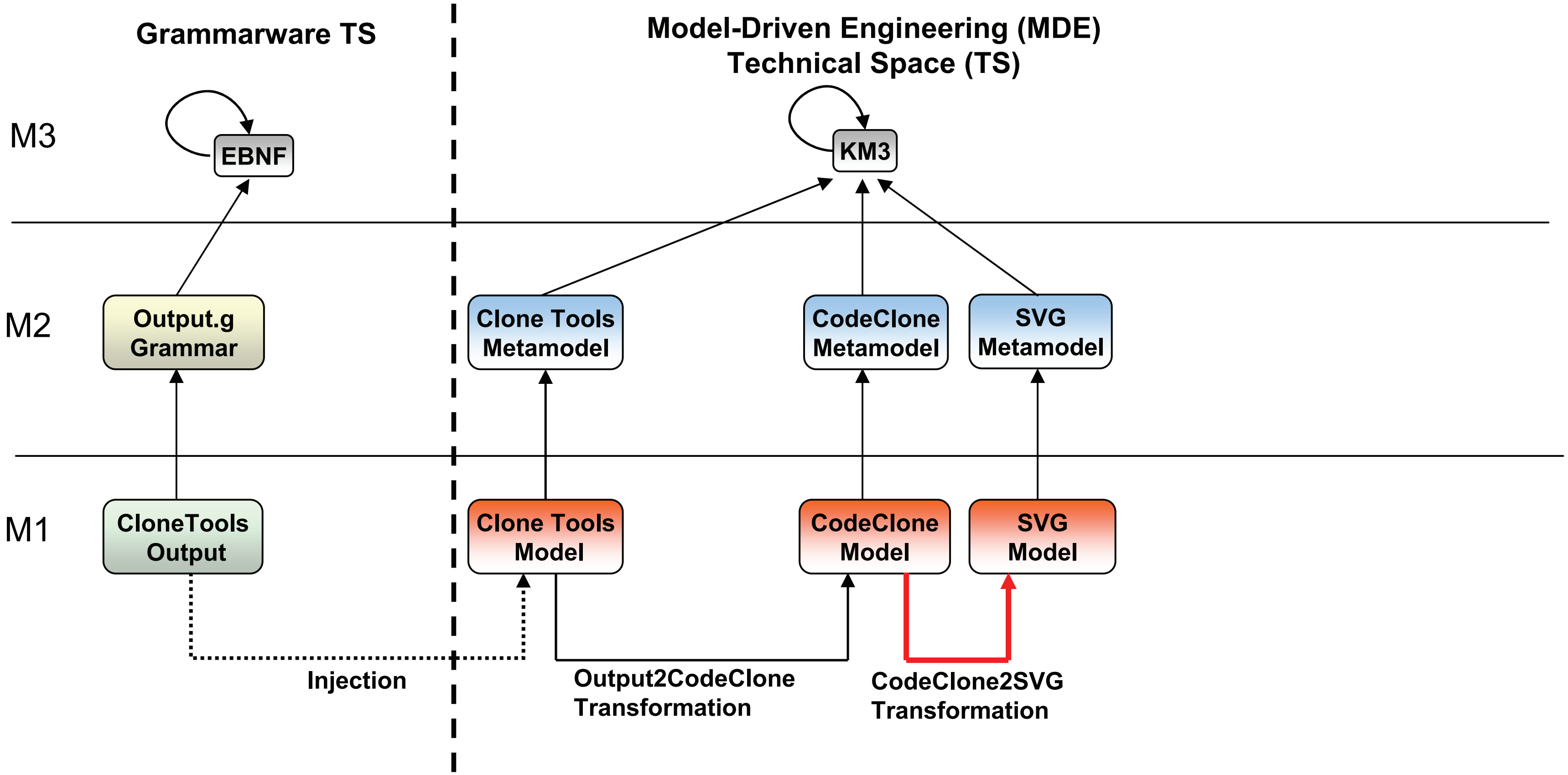
M1 = terminal model level

M2 = metamodel level

M3 = meta-metamodel level

Step 9 – Transformation from CodeClone to SVG

- Most difficult and time-consuming part of the project
- Two different domains
- Difficulties in using ATL
- Referred to Table to SVG example
- Problem: Total number of lines is not available



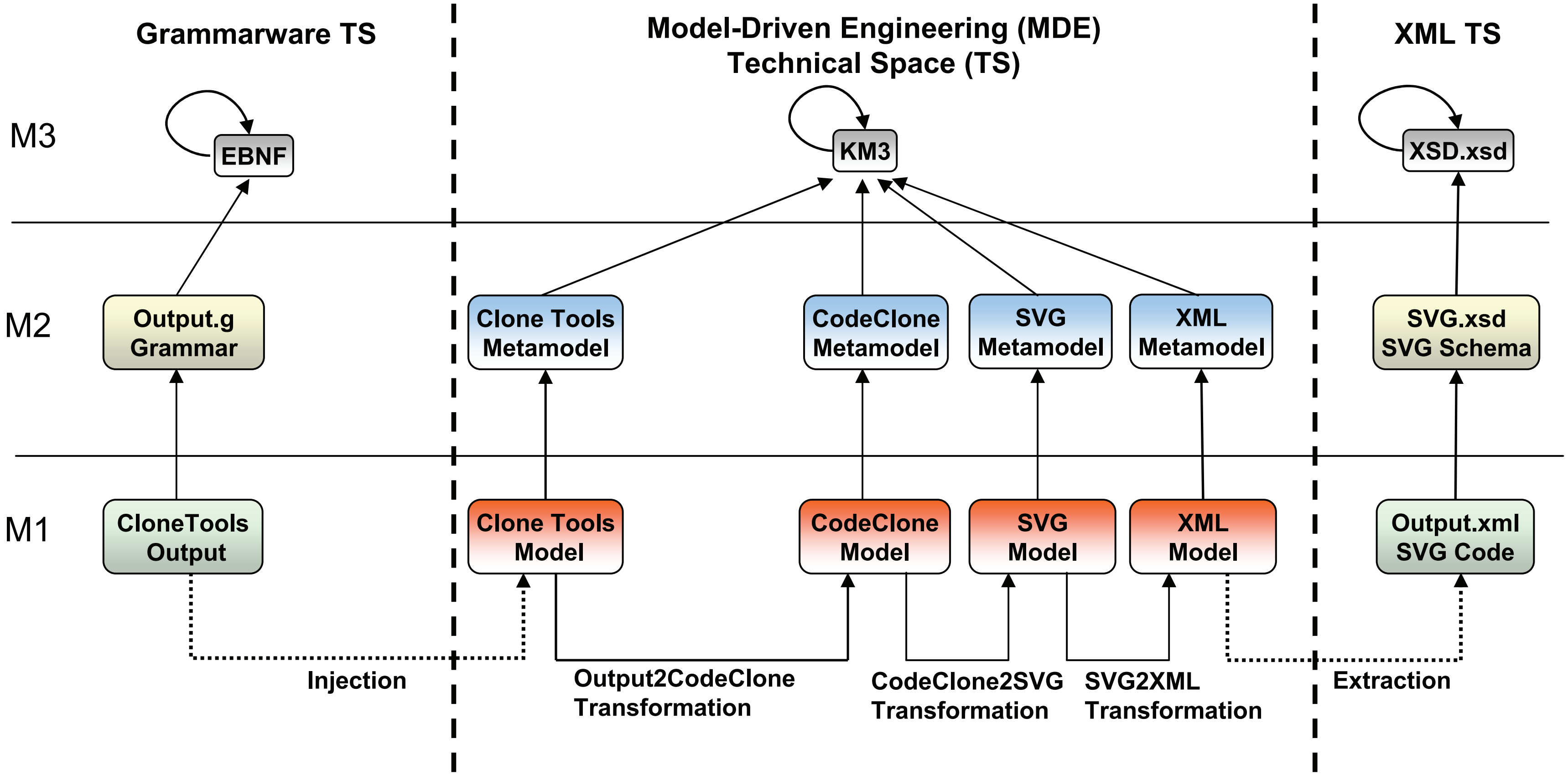
M1 = terminal model level

M2 = metamodel level

M3 = meta-metamodel level

Step 10 – Transformation from SVG to XML

- XML Metamodel is already available
- Referred to an example SVG to XML



M1 = terminal model level

M2 = metamodel level

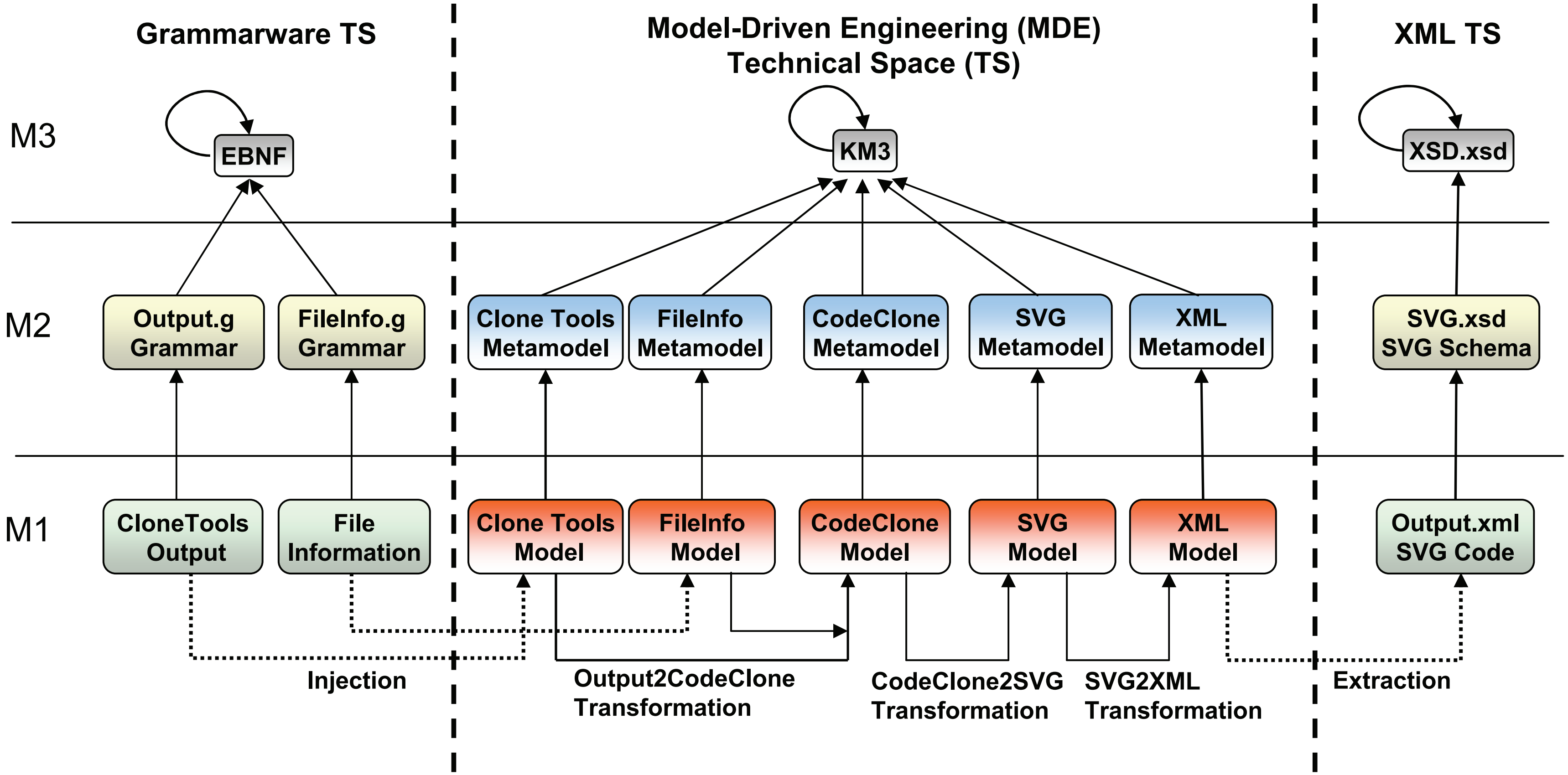
M3 = meta-metamodel level

Step 11 – Test the whole process

- Tried different input model to verify the correctness of transformations
- Modified the positions of the elements in the final graph
- Modified the Metamodel and TCS for Simian to add some file information in the end of the input

Step 12 – Adding FileInfo DSL

- Define KM3 and TCS for FileInfo
- Make Simian & FileInfo as the input to the CodeClone Transformation



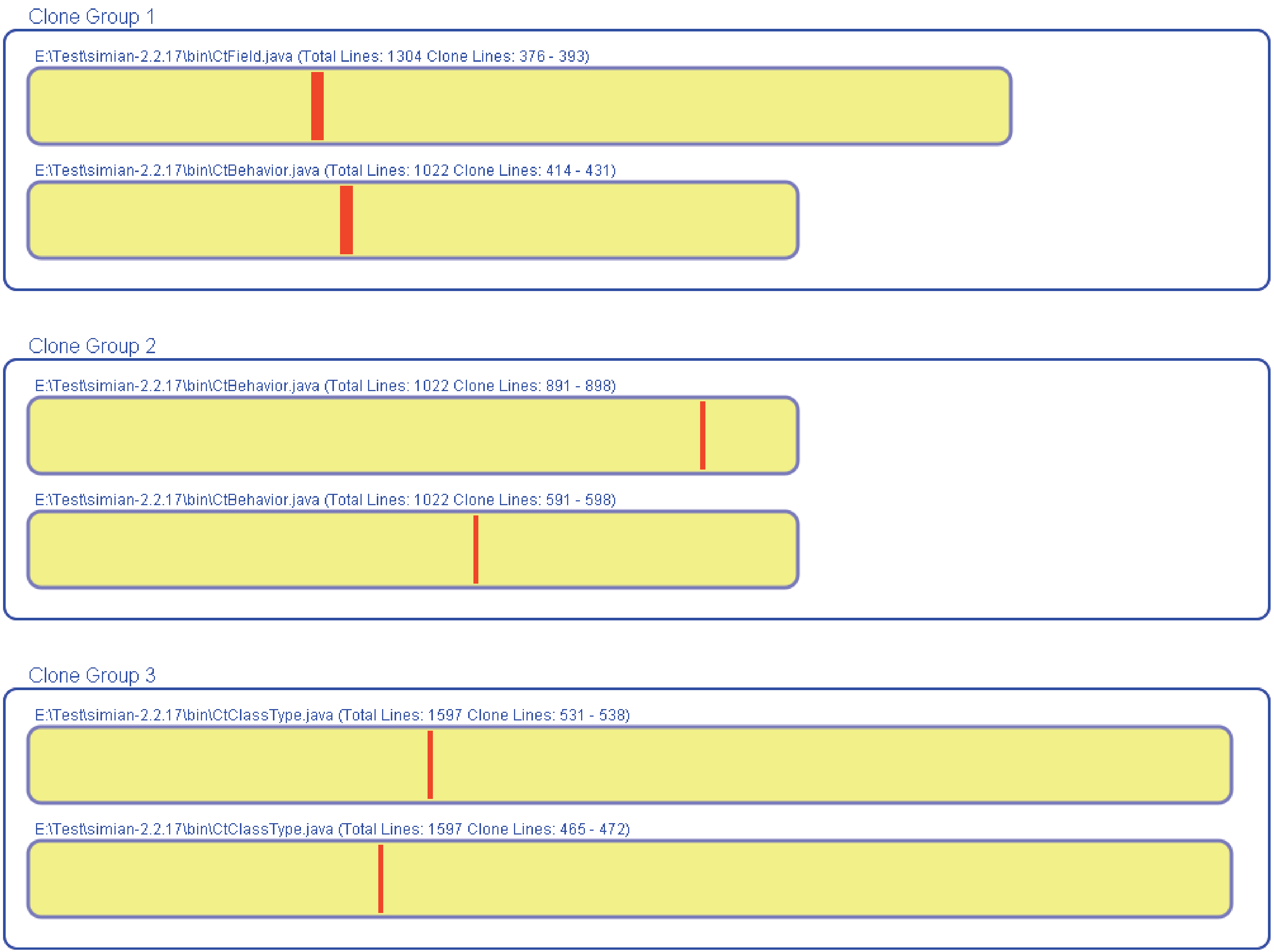
M1 = terminal model level
M2 = metamodel level
M3 = meta-metamodel level

Step 13 – Support a new clone tool SimScan

- Define the KM3 and TCS
- Transformation to CodeClone
- Test

Step 14 – Another kind of Representation

- Design the graph and calculate the position
- Write the ATL transformation CodeClone2SVG2



Clone Group 1

E:\Test\simian-2.2.17\bin\CtField.java (Total Lines: 1304 Clone Lines: 376 - 393)

E:\Test\simian-2.2.17\bin\CtBehavior.java (Total Lines: 1022 Clone Lines: 414 - 431)

Clone Group 2

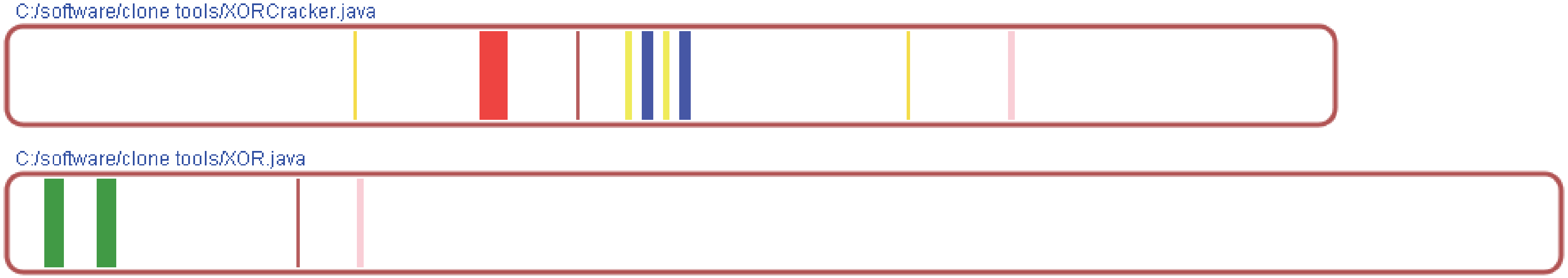
E:\Test\simian-2.2.17\bin\CtBehavior.java (Total Lines: 1022 Clone Lines: 891 - 898)

E:\Test\simian-2.2.17\bin\CtBehavior.java (Total Lines: 1022 Clone Lines: 591 - 598)

Clone Group 3

E:\Test\simian-2.2.17\bin\CtClassType.java (Total Lines: 1597 Clone Lines: 531 - 538)

E:\Test\simian-2.2.17\bin\CtClassType.java (Total Lines: 1597 Clone Lines: 465 - 472)

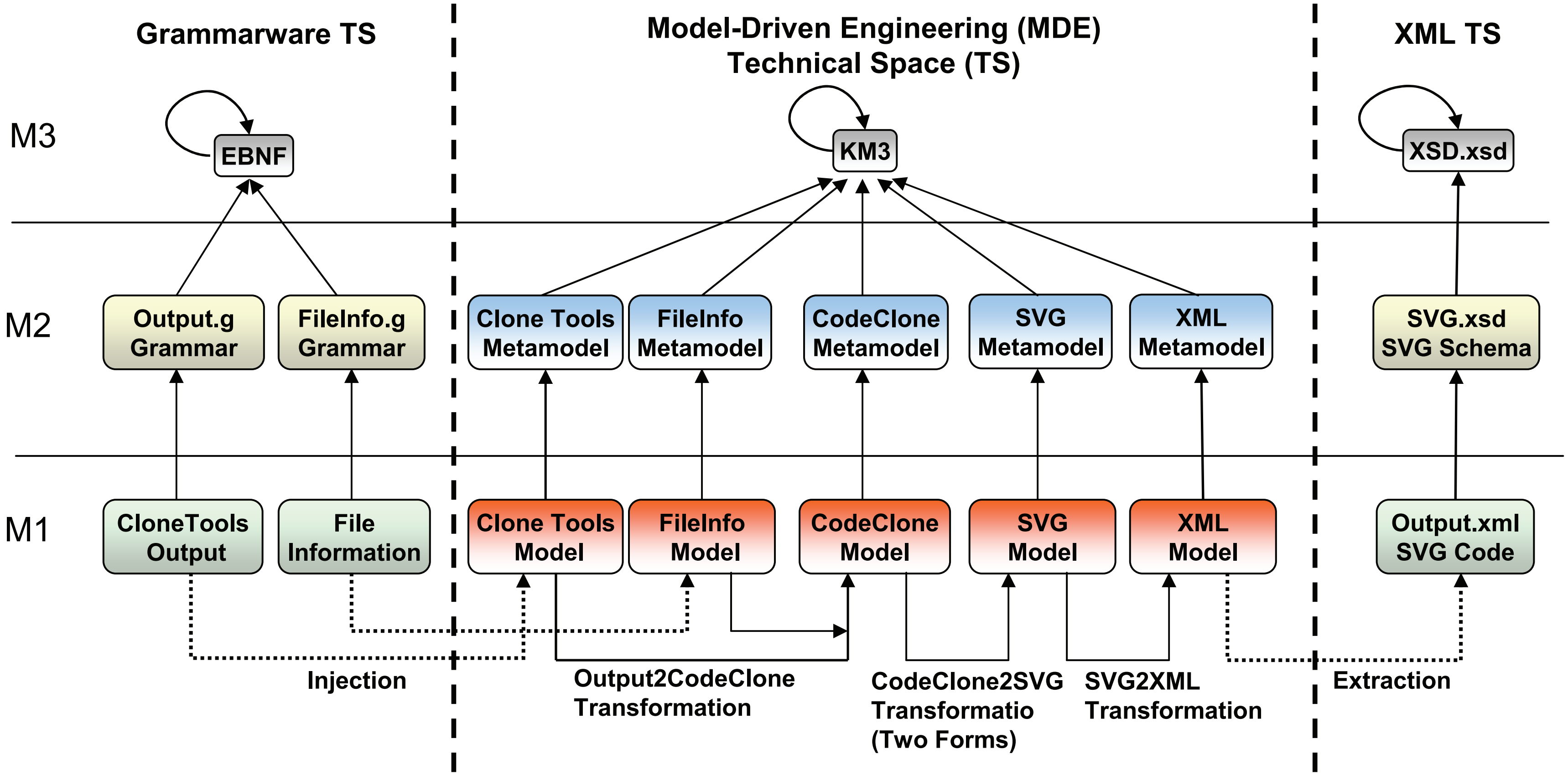


C:/software/clone tools/XORCracker.java

C:/software/clone tools/XOR.java

Step 15 – Megamodel Graph

- Better understand the whole picture and structure



M1 = terminal model level

M2 = metamodel level

M3 = meta-metamodel level

Step 16 – Ant scripts

- Simian2Visual
- SimScan2Visual

Step 17 – Support CloneDr

- A new version of CloneDr offers a good output file for this project
- Finished its KM3, TCS, ATL

Future work

- Improve the parsing of the input result
  - Some parts of the input is deleted
  - Some inputs are too complex to parse
- Improve the details of the visual representation
- Step 19 – Write the paper

Thank you!!!