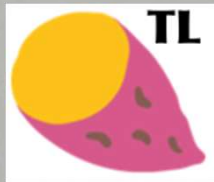




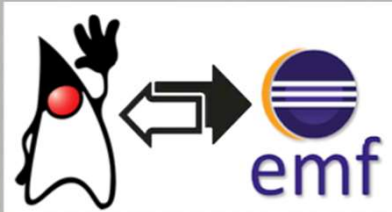
UNIVERSITY OF  
**LEICESTER**

15<sup>th</sup> Transformation Tool Contest  
TTC 2023, Leicester, UK

# YAMTL+EMF-Syncer Solution Containers to MiniYAML case

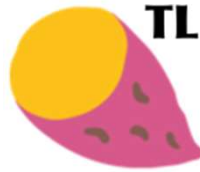


+



Artur Boronat, University of Leicester, UK

# YAMTL



- Declarative M2M trafos in Xtend and now **Groovy**
  - **Unidirectional trafos (out-place)**
    - Pattern matching
    - Object resolution strategy
    - Rules with multiple inheritance
    - Module composition
- Execution modes
  - Batch
  - **Incremental** for rule applications
    - Dependency tracking
    - Forward propagation
- Transformation semantics: source is the ground truth



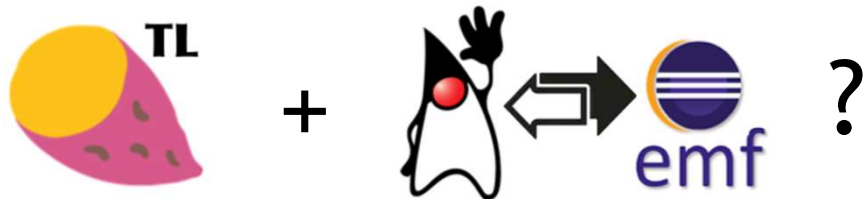
- Declarative M2M trafos in Xtend and now **Groovy**
  - **Unidirectional trafos (out-place)**
    - Pattern matching
    - Object resolution strategy
    - Rules with multiple inheritance
    - Module composition
- Execution modes
  - Batch
  - **Incremental** for rule applications
    - Dependency tracking
    - Forward propagation
- Transformation semantics: source is the ground truth



- Syncing between Java program snapshots and EMF model instances
  - Infers mappings between a Java program and an Ecore domain model **at feature level**
  - **Bidirectional trafos**
- Execution modes
  - Batch: eager/lazy
  - Incremental forward and backward
- Syncing semantics: merging

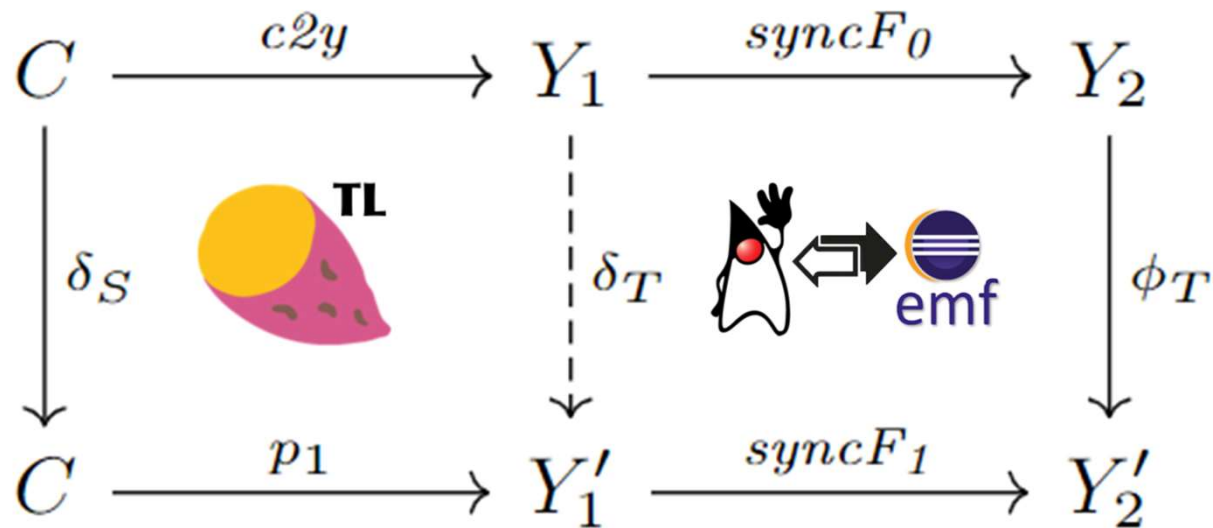
# Integration of YAMTL and EMF-Syncer: attempt 1

- Attempt 1
  - Use EMF adapters
- Problem 1: spurious changes
  - In small models, this is akin to a batch transformation
- Problem 2: information loss in the target
  - YAMTL semantics considers that the target is a projection of the source



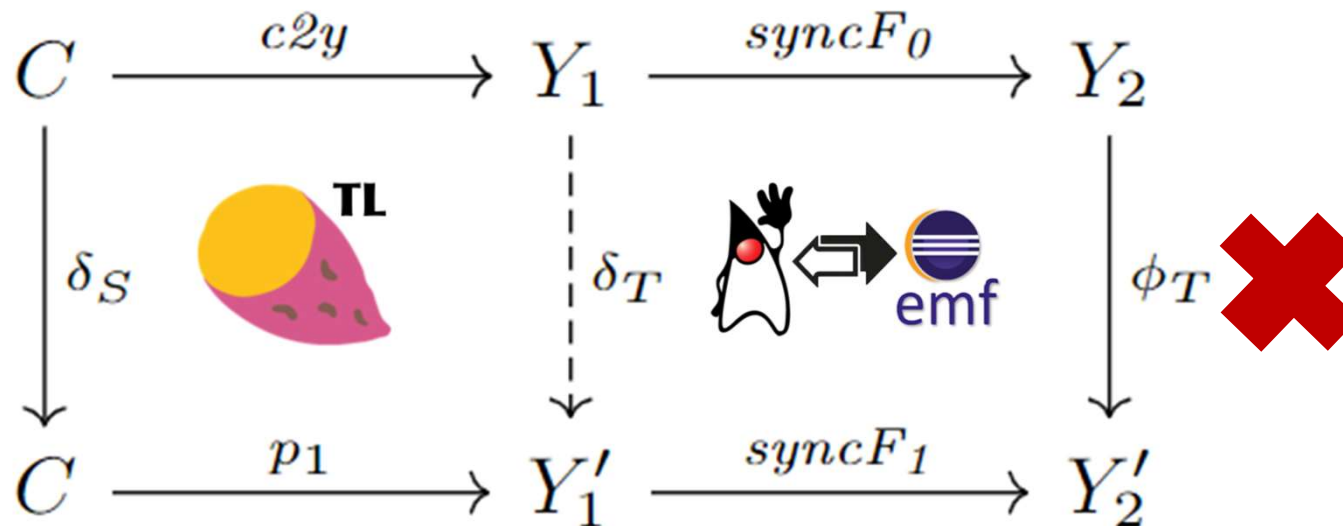
# Integration of YAMTL and EMF-Syncer: attempt 1

- Use EMF adapters
- Problem 1: spurious changes
  - In small models, this is akin to a batch transformation
- Problem 2: information loss in the target
  - YAMTL semantics considers that the target is a projection of the source



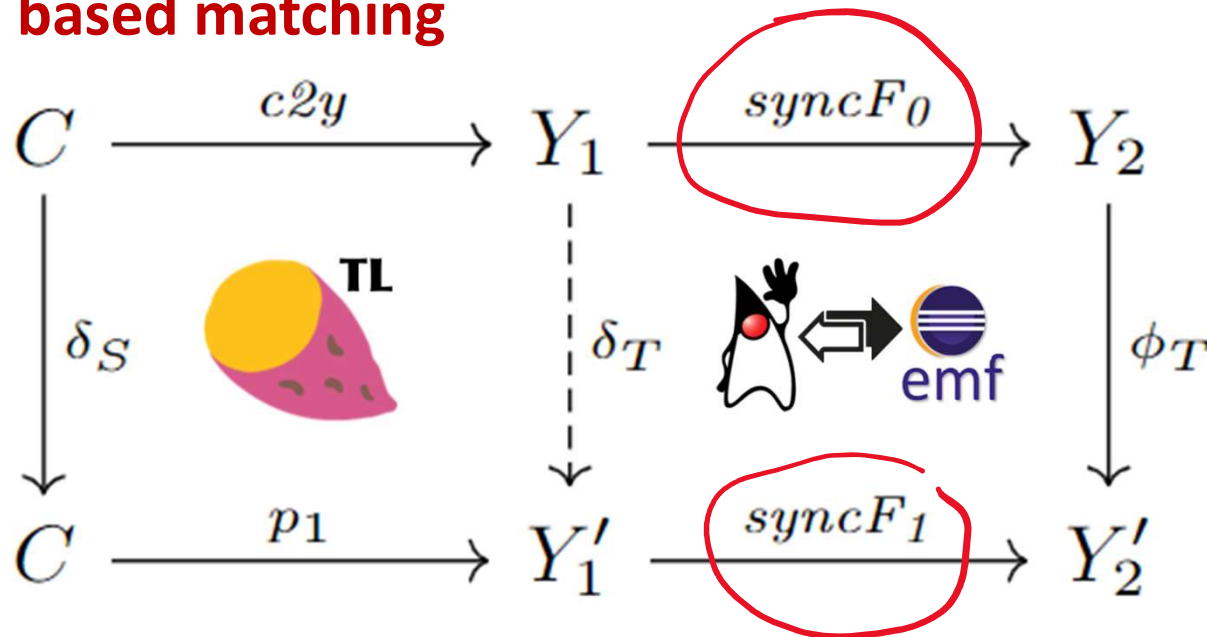
# Integration of YAMTL and EMF-Syncer: attempt 1

- Use EMF adapters
- Problem 1: spurious changes
  - In small models, this is akin to a batch transformation
- Problem 2: information loss in the target
  - YAMTL semantics considers that the target is a projection of the source



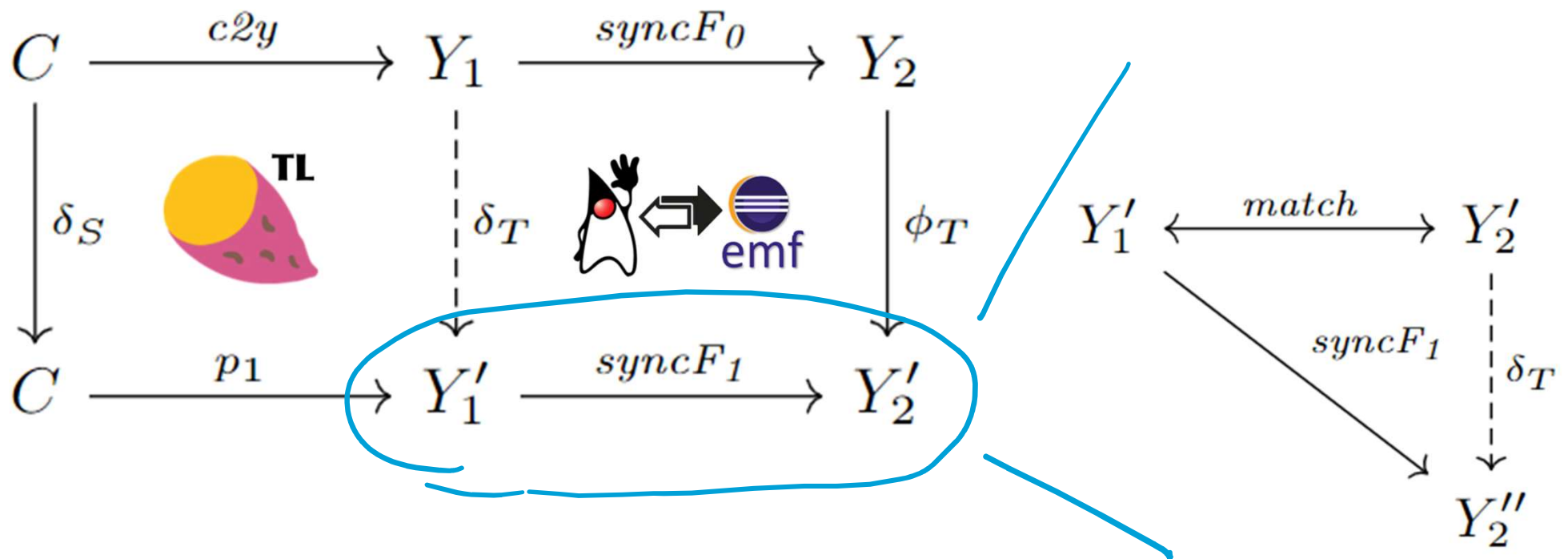
# Integration of YAMTL and EMF-Syncer: attempt 2

- Attempt 2: decouple YAMTL and EMF-Syncer, via incremental **state-based matching**



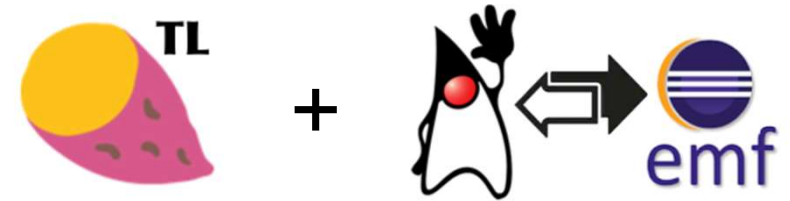
# Integration of YAMTL and EMF-Syncer: attempt 2

- Attempt 2: decouple YAMTL and EMF-Syncer, via incremental state-based matching

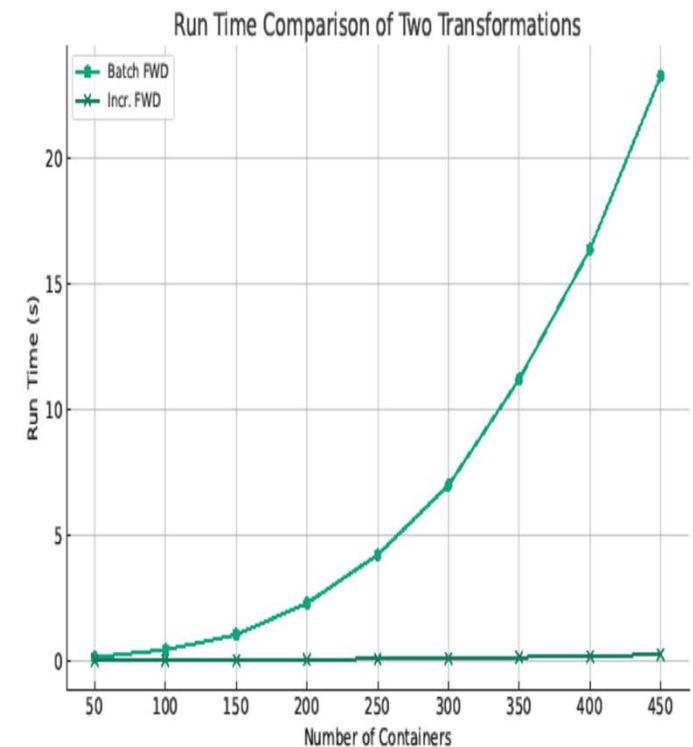




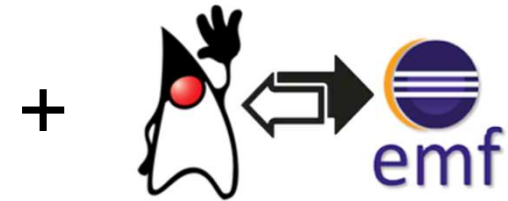
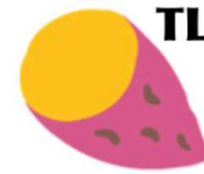
# Evaluation



- **Conciseness**
- Correctness: tie with the best approaches
- Scalability Batch FWD and Incr FWD
  - **Incrementality** is not impacted by composition of YAMTL and EMF-Syncer
- Limitation: interpreter does have an impact on **run-time** performance



# Conclusions



- Aims
  - Experiment with Groovy as a host language for YAMTL: conciseness
  - Experiment with the integration of YAMTL and EMF-Syncer
  - Experiment with state-based matching in EMF-Syncer
- Achievements
  - All of YAMTL features work from Groovy
    - Concise transformation
  - Full solution: YAMTL for transformations and EMF-Syncer for reconciliation
    - Both are incremental
  - Glue code is minimal
- Limitations
  - As with all other solutions: order is not considered in incremental forward
  - Scalability test (batch backward): fails at 250 (stackoverflow exception)