

29 August 2008
MSDL Research Presentations
Montréal, Canada

Domain-Specific Modelling

... only transform ...

Raphael Mannadiar

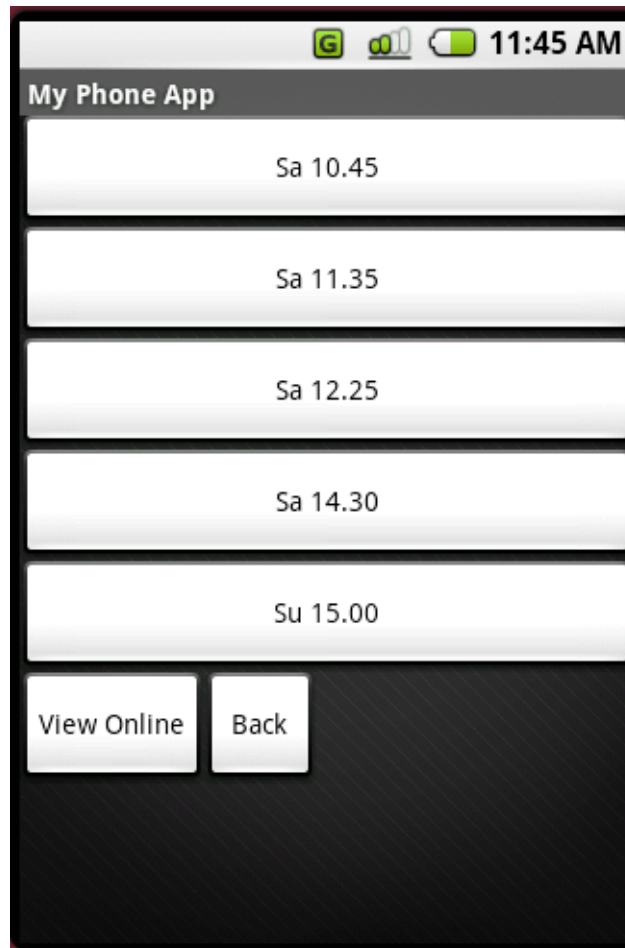


Modelling, Simulation and Design Lab (MSDL)
School of Computer Science, McGill University, Montréal, Canada

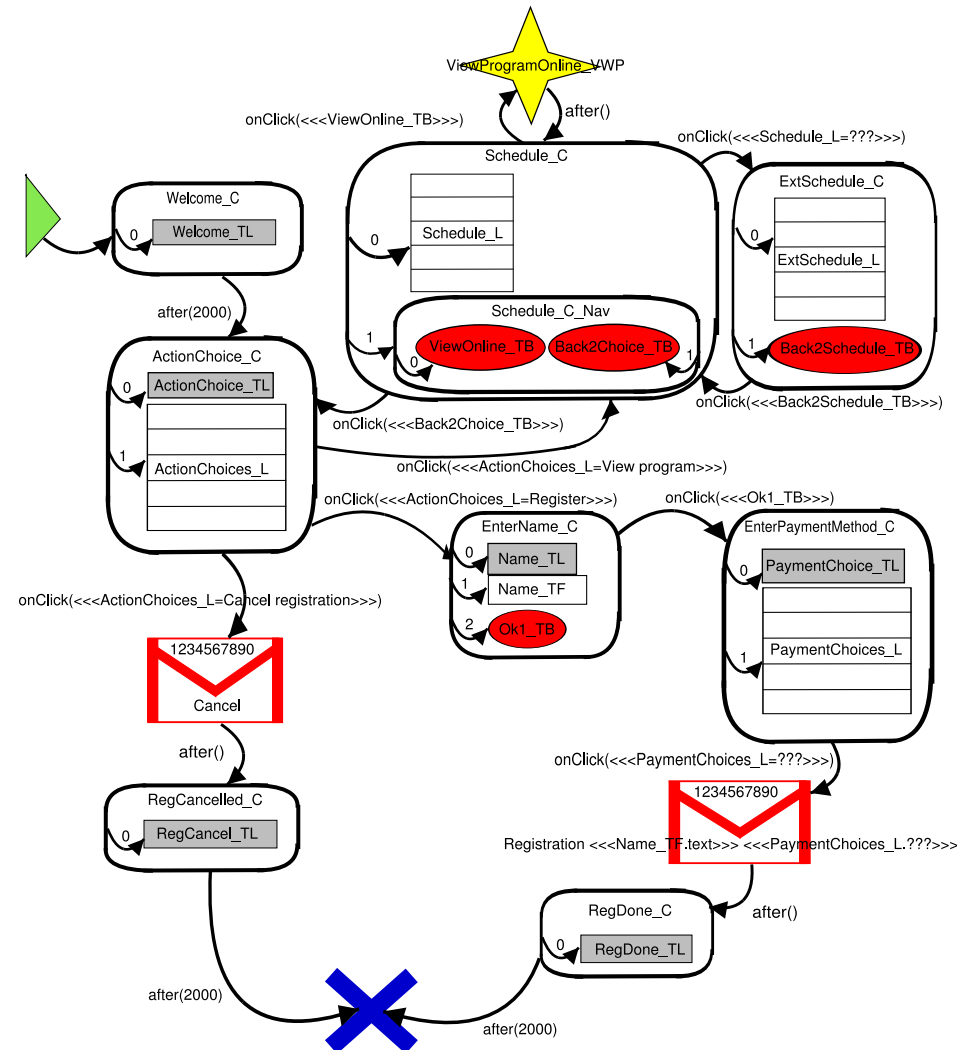
Overview

1. Domain-Specific Modelling
2. Beyond Syntax: Transformation Lattice
3. Conclusions and Future Work

DS(V)M example application: conference registration (Google Android)



DS(V)M example application, the PhoneApps Domain-Specific model



Why DS(V)M ? (as opposed to General Purpose modelling)

- **match the user's mental model** of the problem domain
- **maximally constrain** the user (to the problem at hand)
 - ⇒ easier to learn
 - ⇒ avoid errors
- **separate** domain-expert's work
from analysis/transformation expert's work

Anecdotal evidence of 5 to 10 times speedup

Steven Kelly and Juha-Pekka Tolvanen.

Domain-Specific Modeling: Enabling Full Code Generation. Wiley 2008.

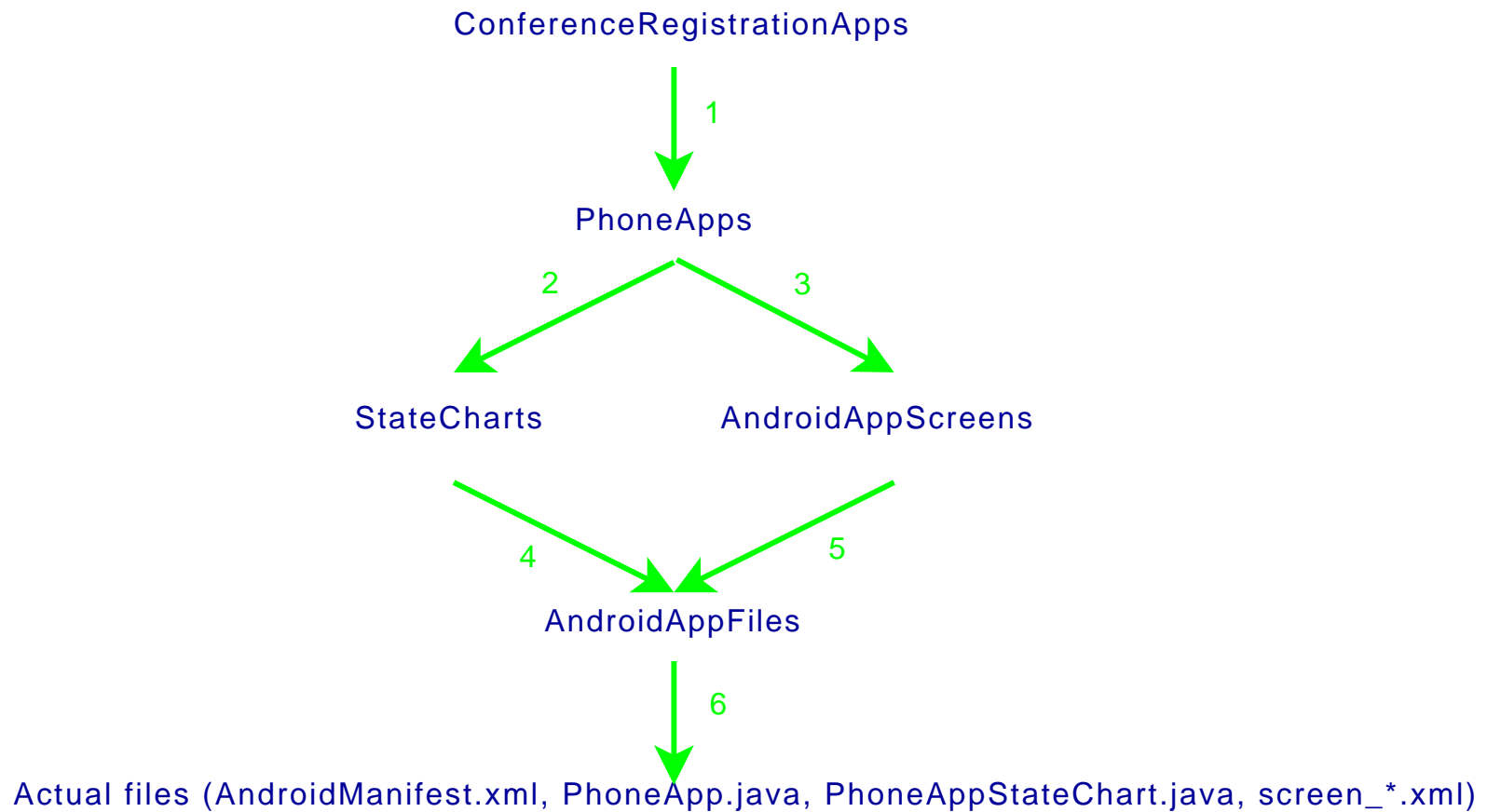
Building DS(V)M Tools Effectively . . .

- **development cost** of DS(V)M Tools may be prohibitive !
- we want to effectively (rapidly, correctly, re-usably, . . .)
 1. Specify DS(V)L **syntax**:
 - **abstract** \Rightarrow **meta-modelling**
 - **concrete** (textual/visual)
 2. Specify DS(V)L **semantics**:
transformation
 3. Model (and analyze and execute) model **transformations**:
 \Rightarrow **graph rewriting**

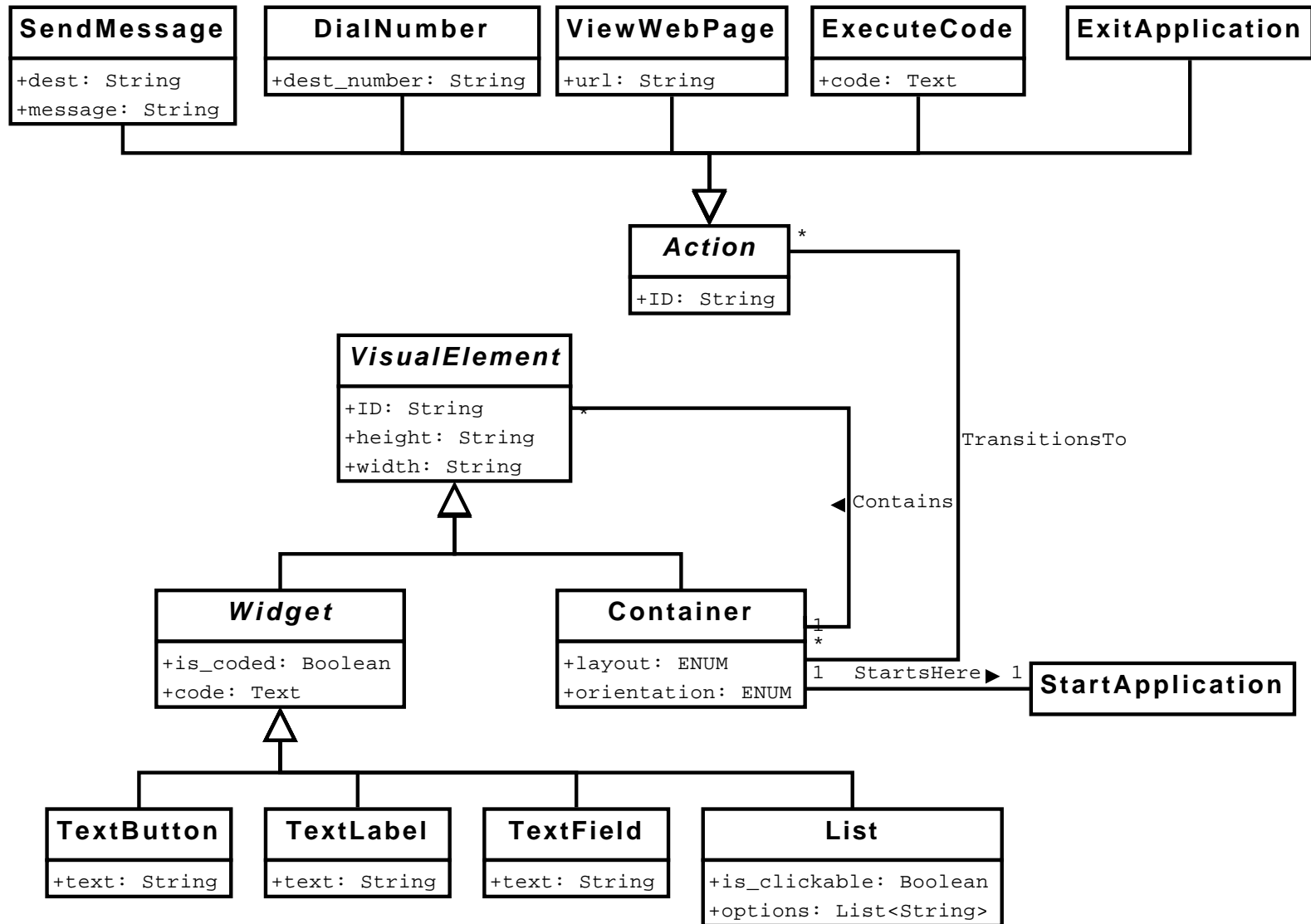
\Rightarrow **model everything**

(in the most appropriate formalism,
at the appropriate level of abstraction)

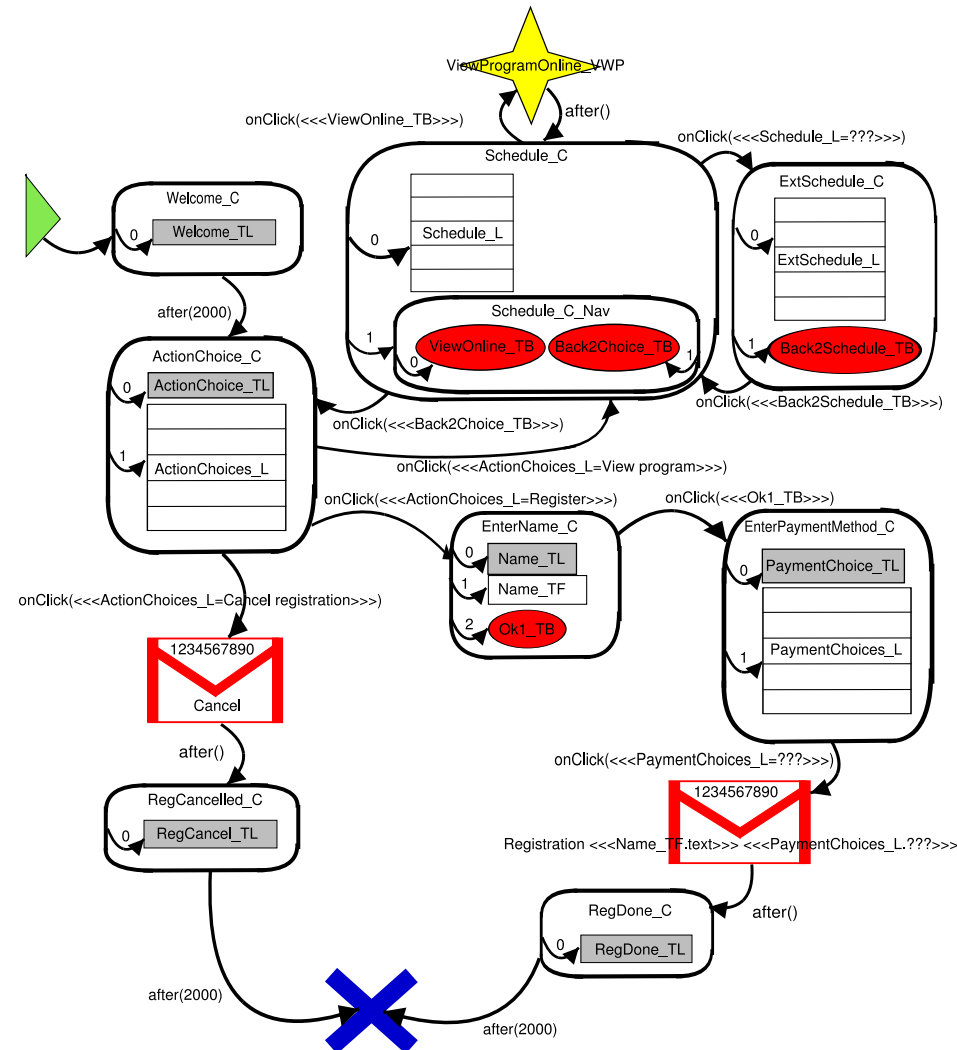
Formalism Transformation Graph



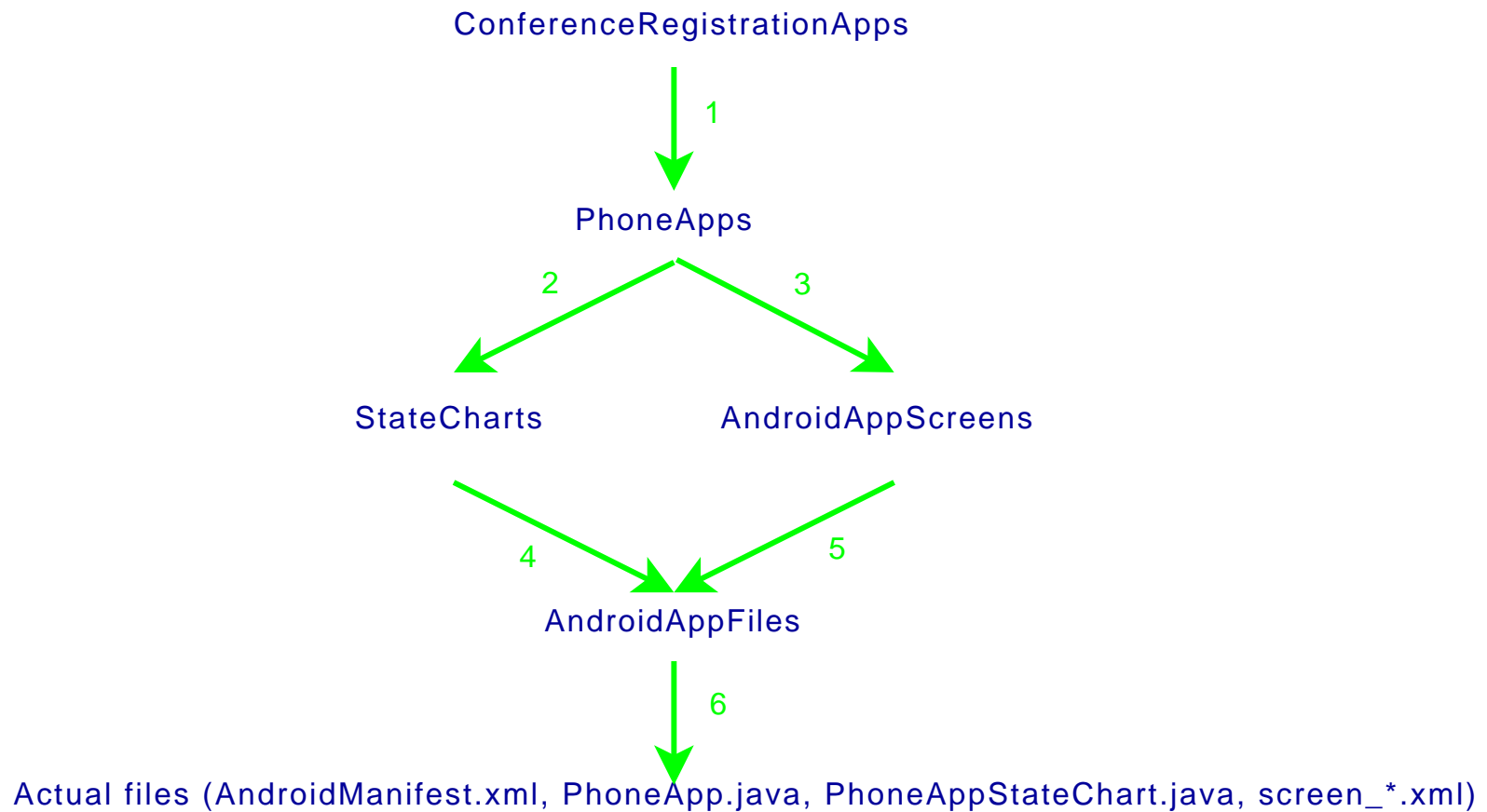
PhoneApps Meta-model



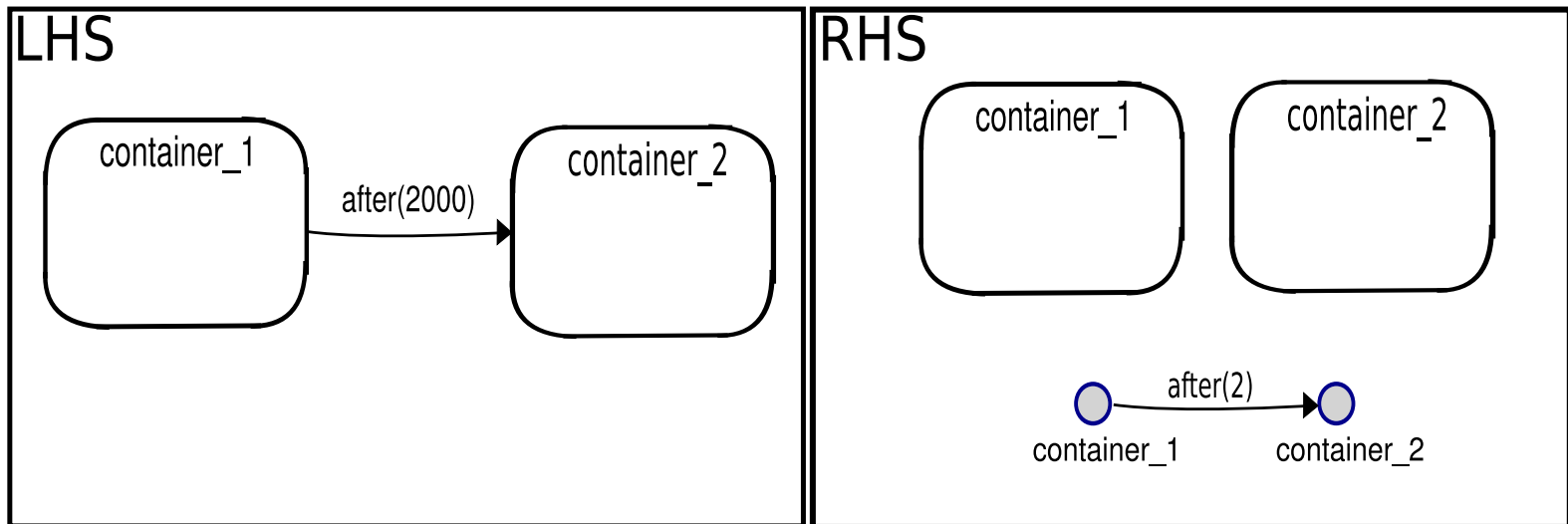
DS(V)M example application, the PhoneApps Domain-Specific model



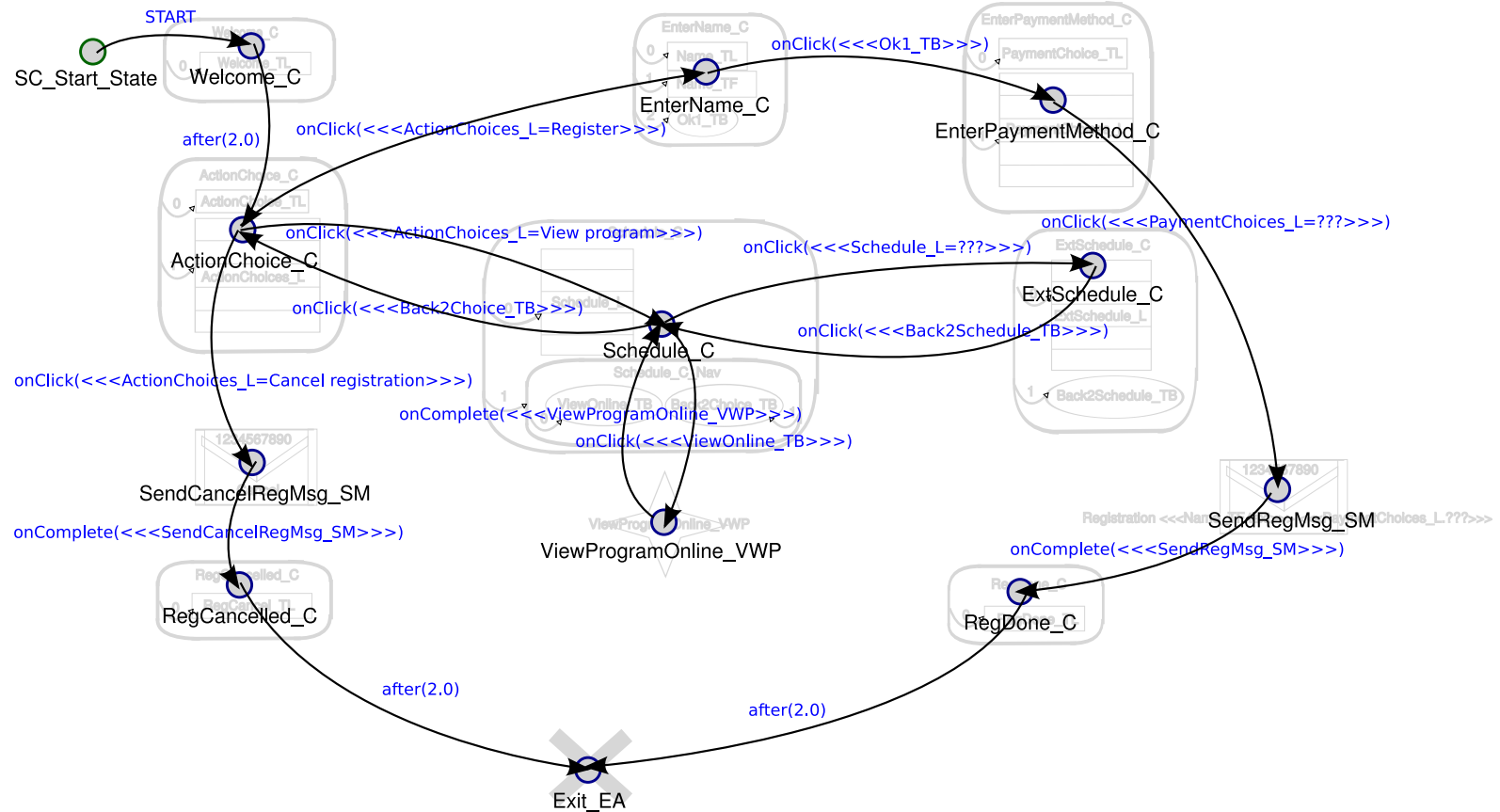
Formalism Transformation Graph



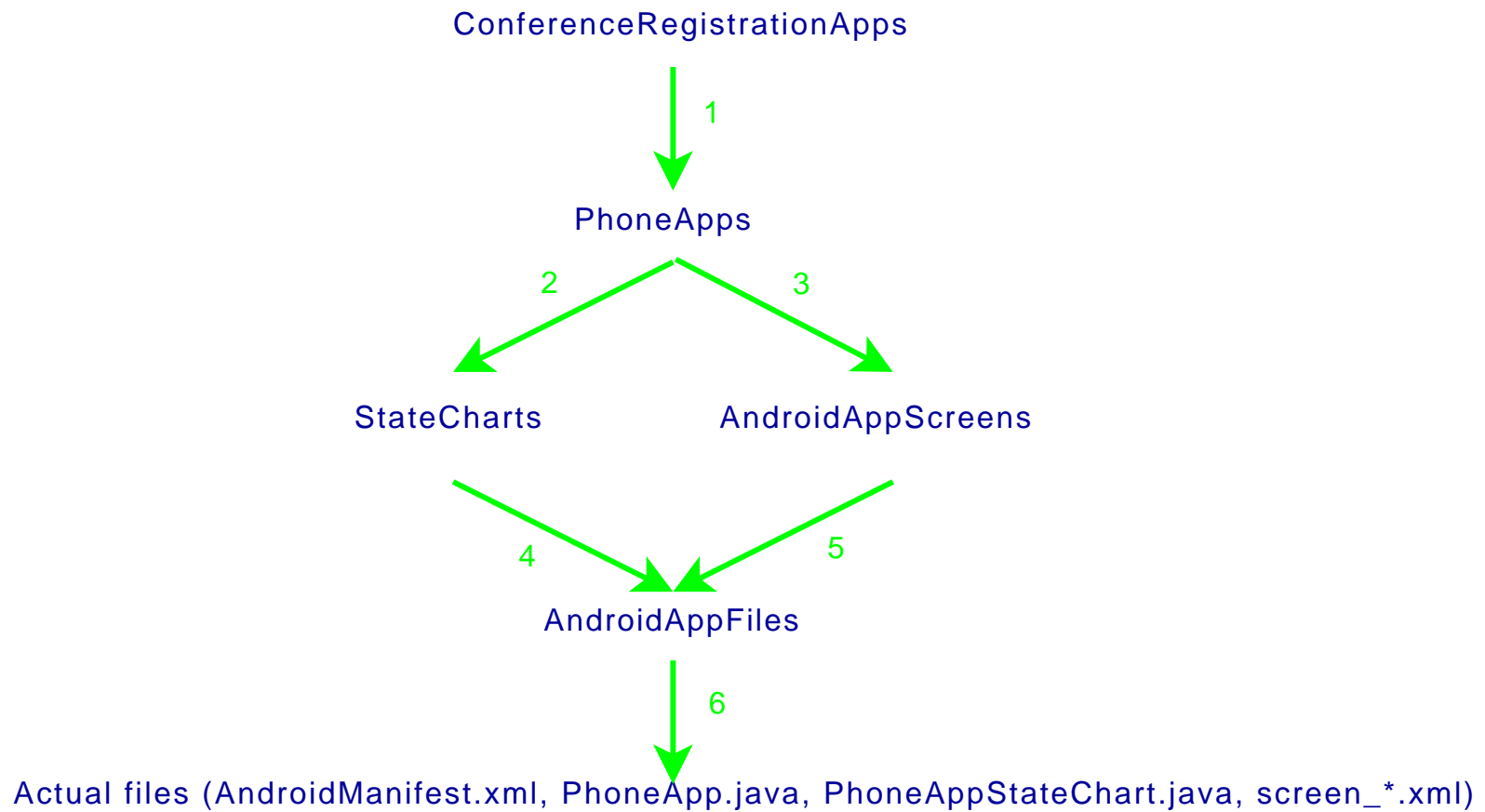
Rule: PhoneApps containers to Statechart



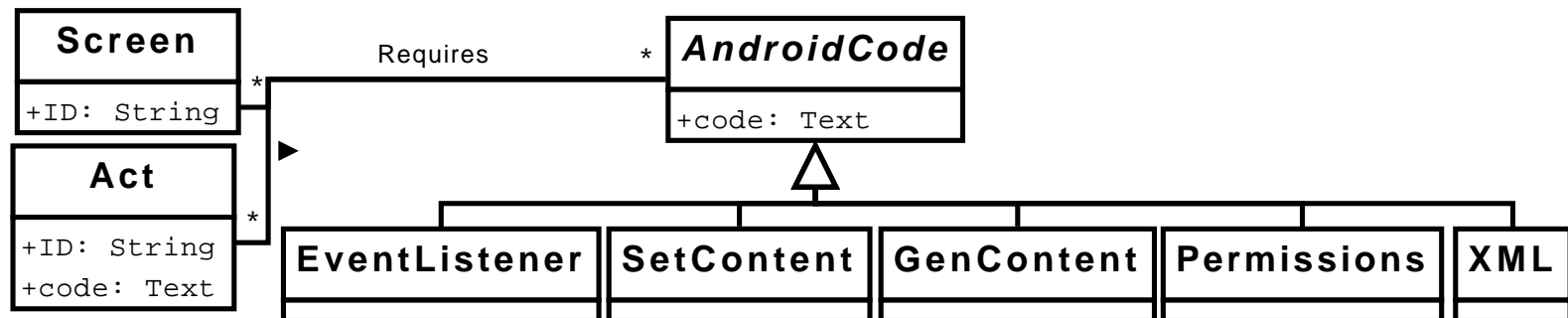
Transformation result: Statechart



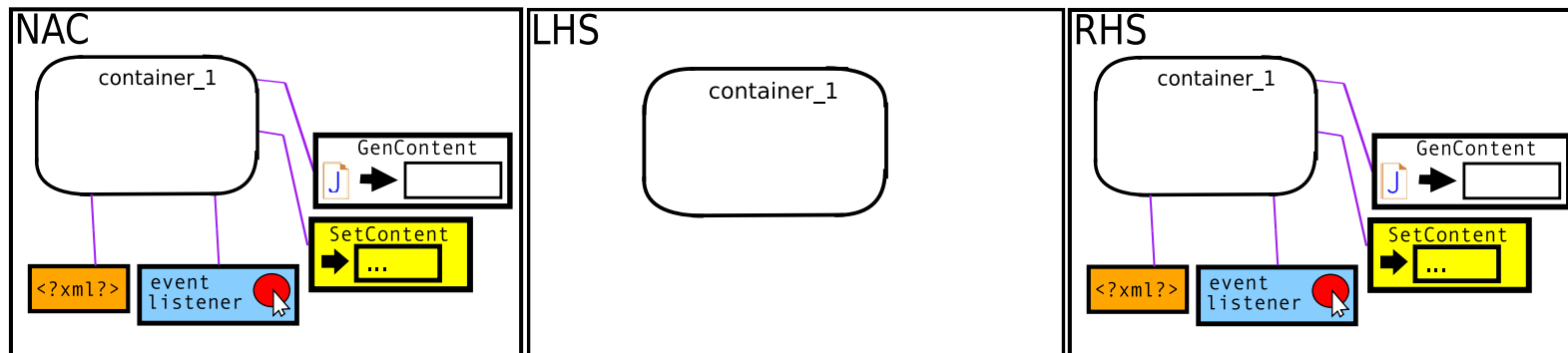
Formalism Transformation Graph



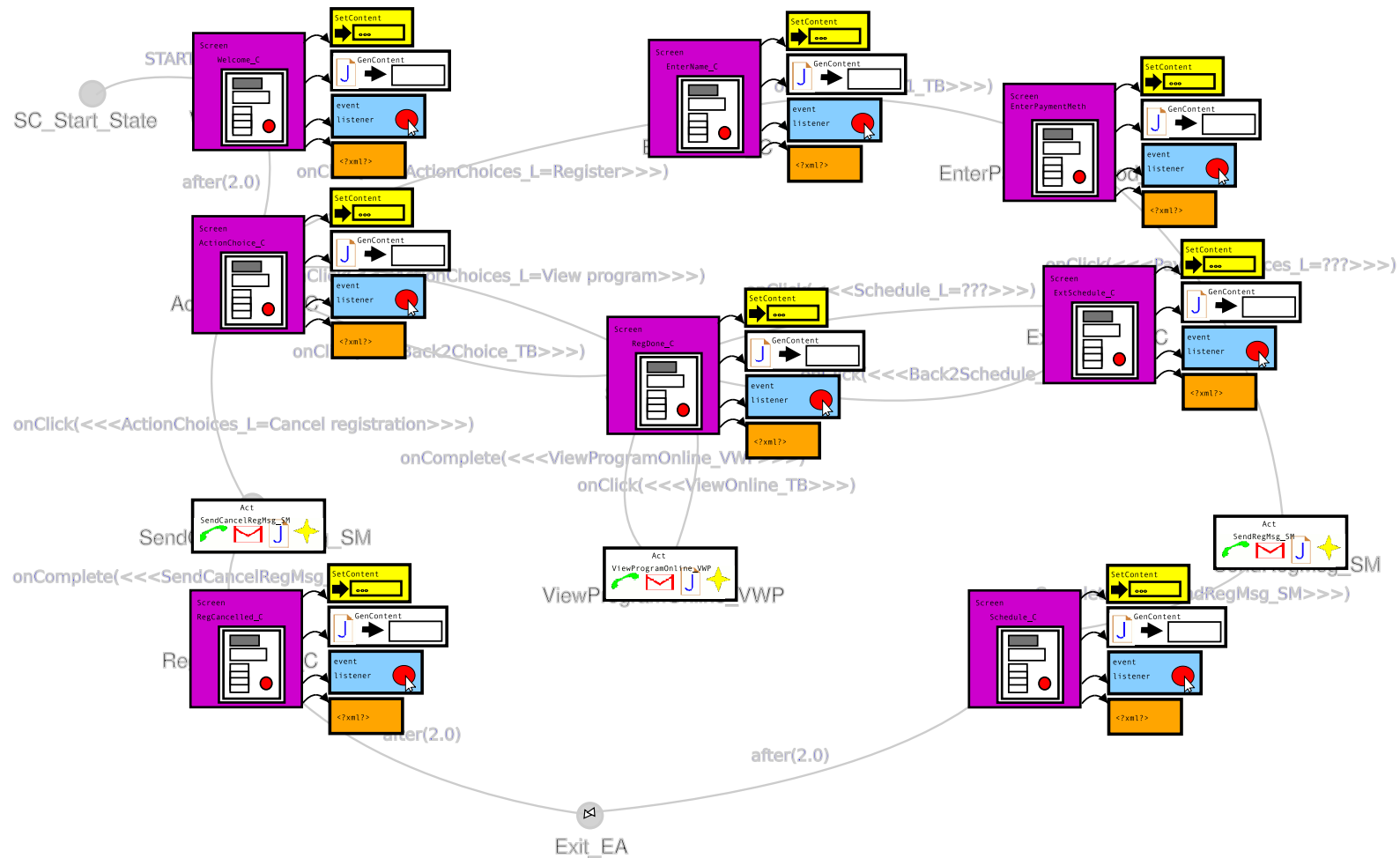
PhoneAppsScreens Meta-model



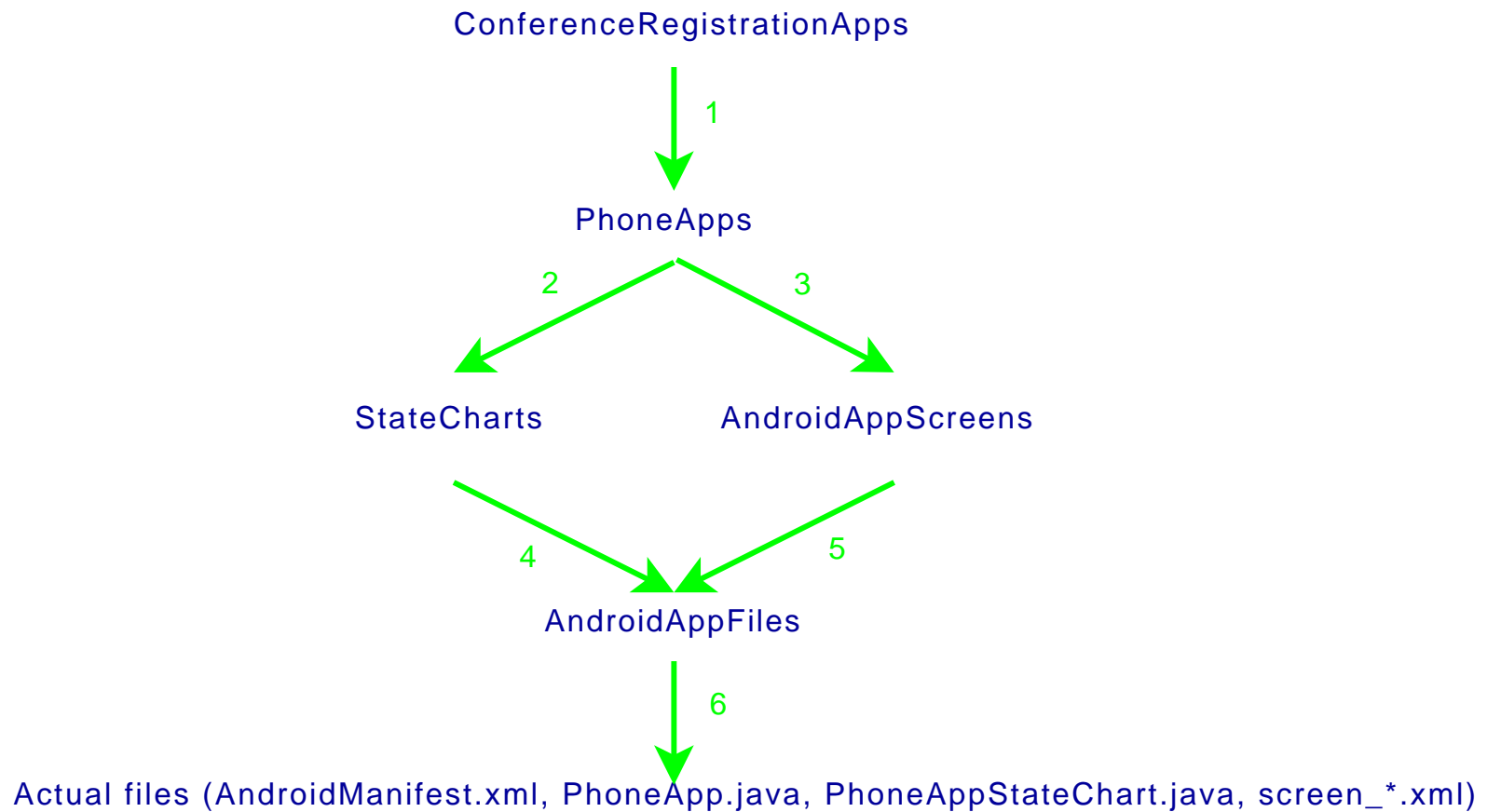
Rule: PhoneApps to PhoneAppsScreens



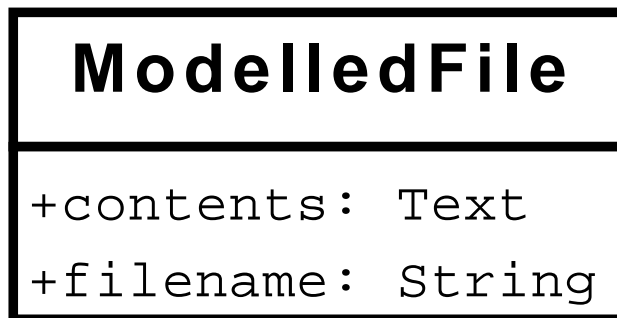
Transformation result: PhoneAppsScreens



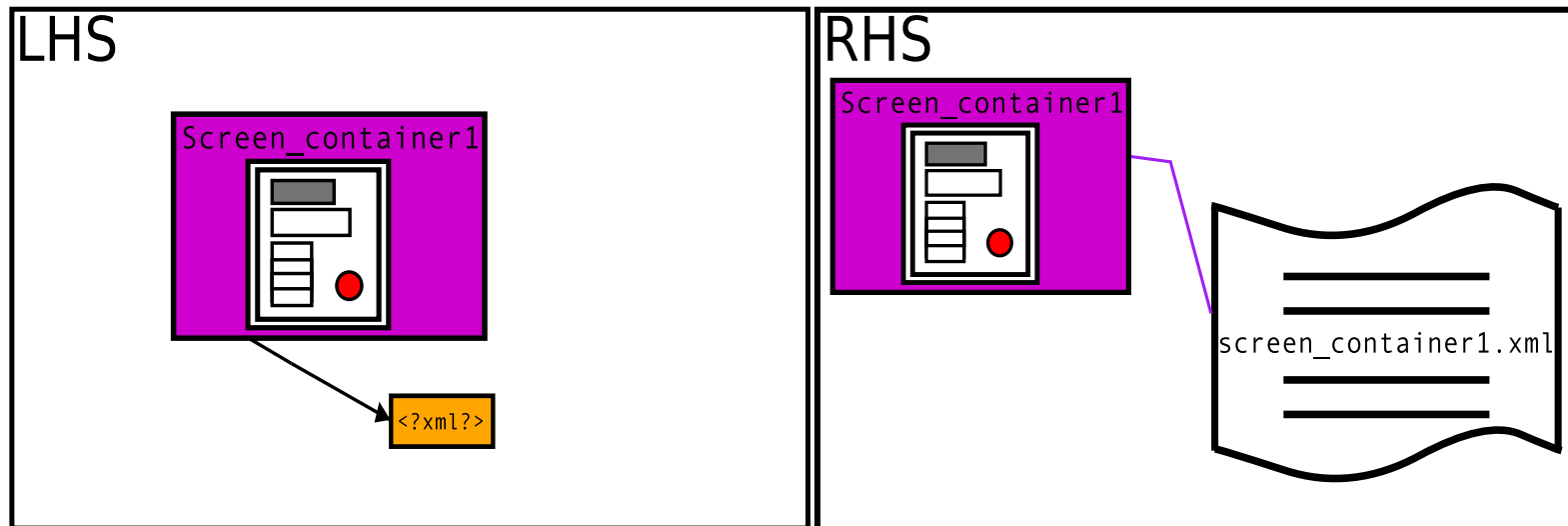
Formalism Transformation Graph



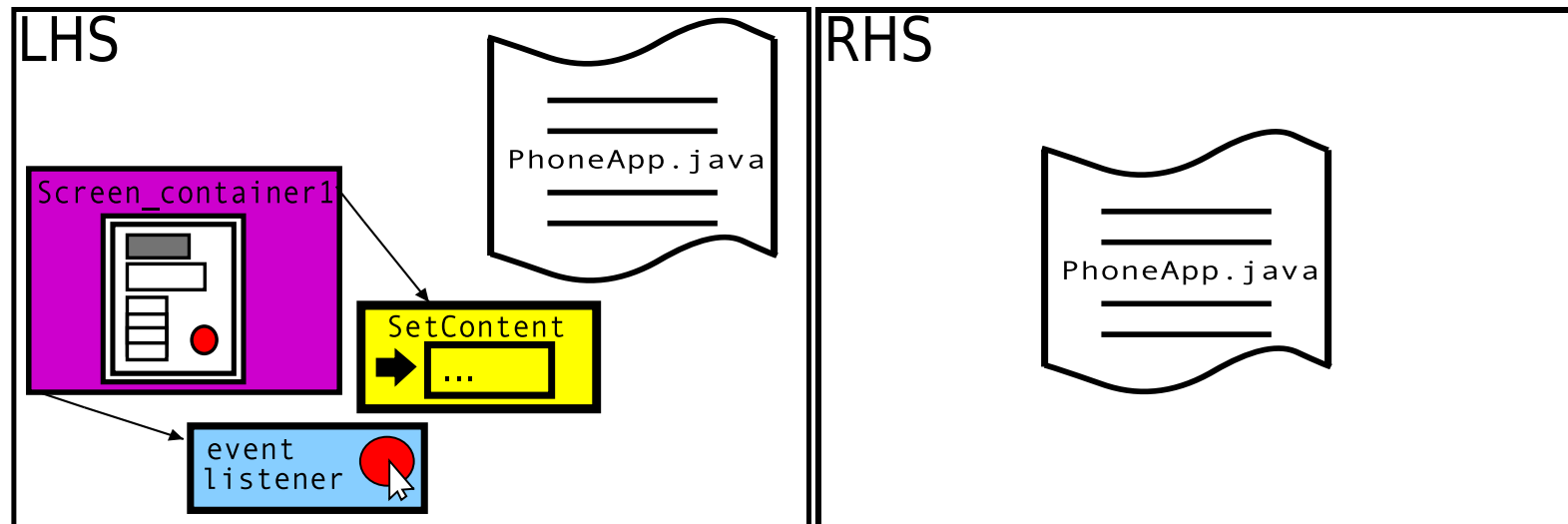
Android target-specific files Meta-model



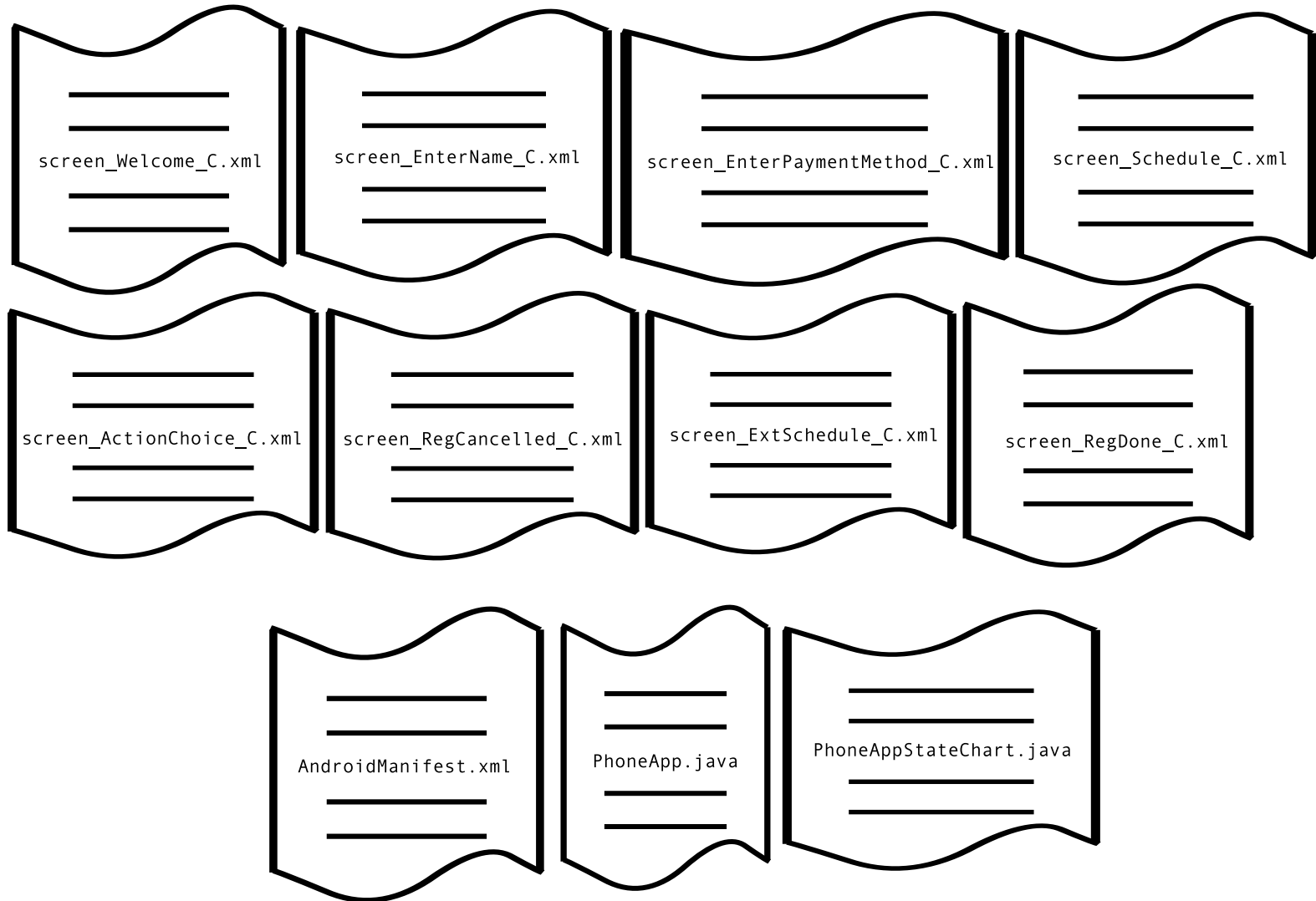
PhoneAppsScreens to Android XML files



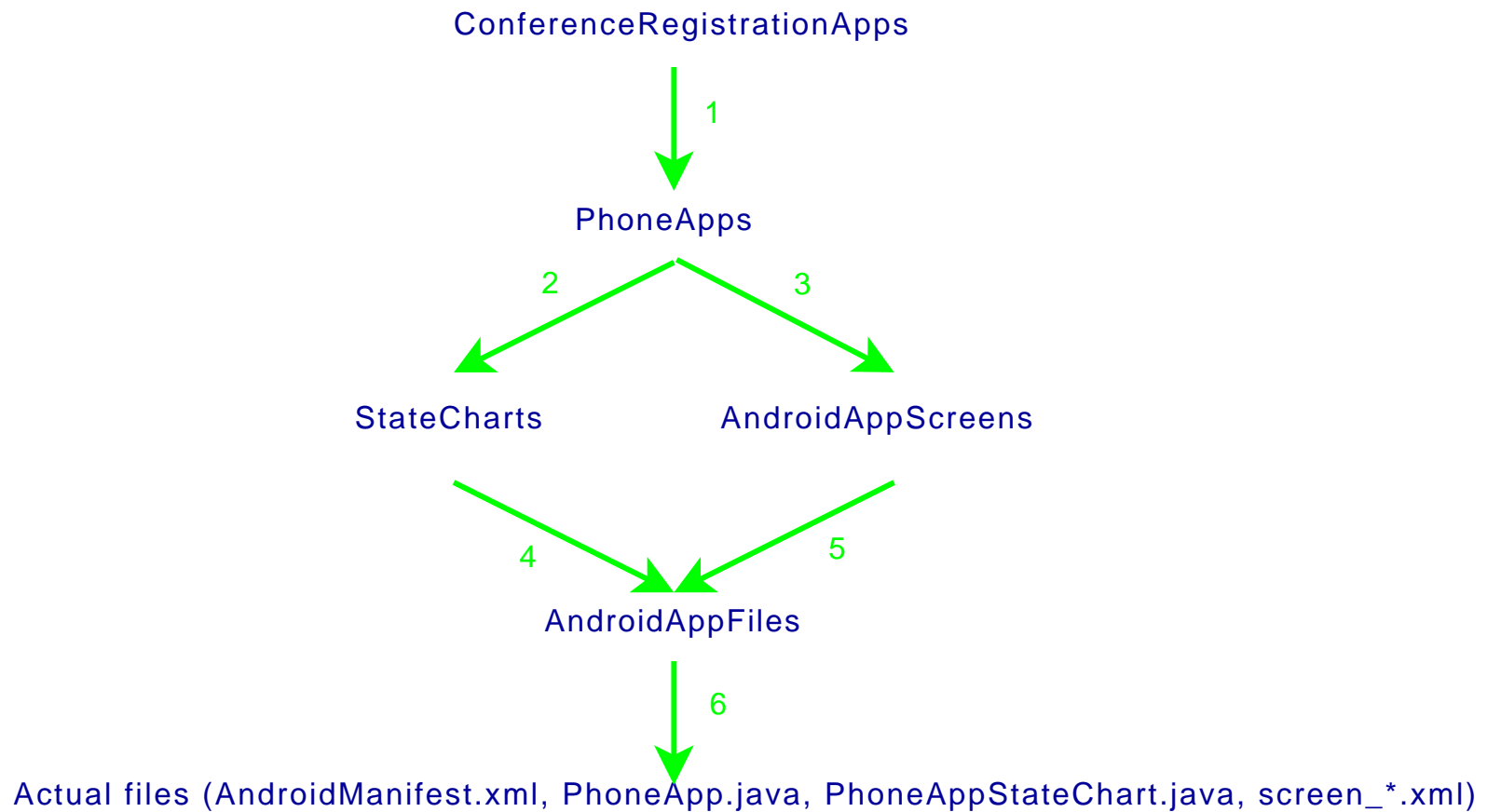
PhoneAppsScreens to Android Java code



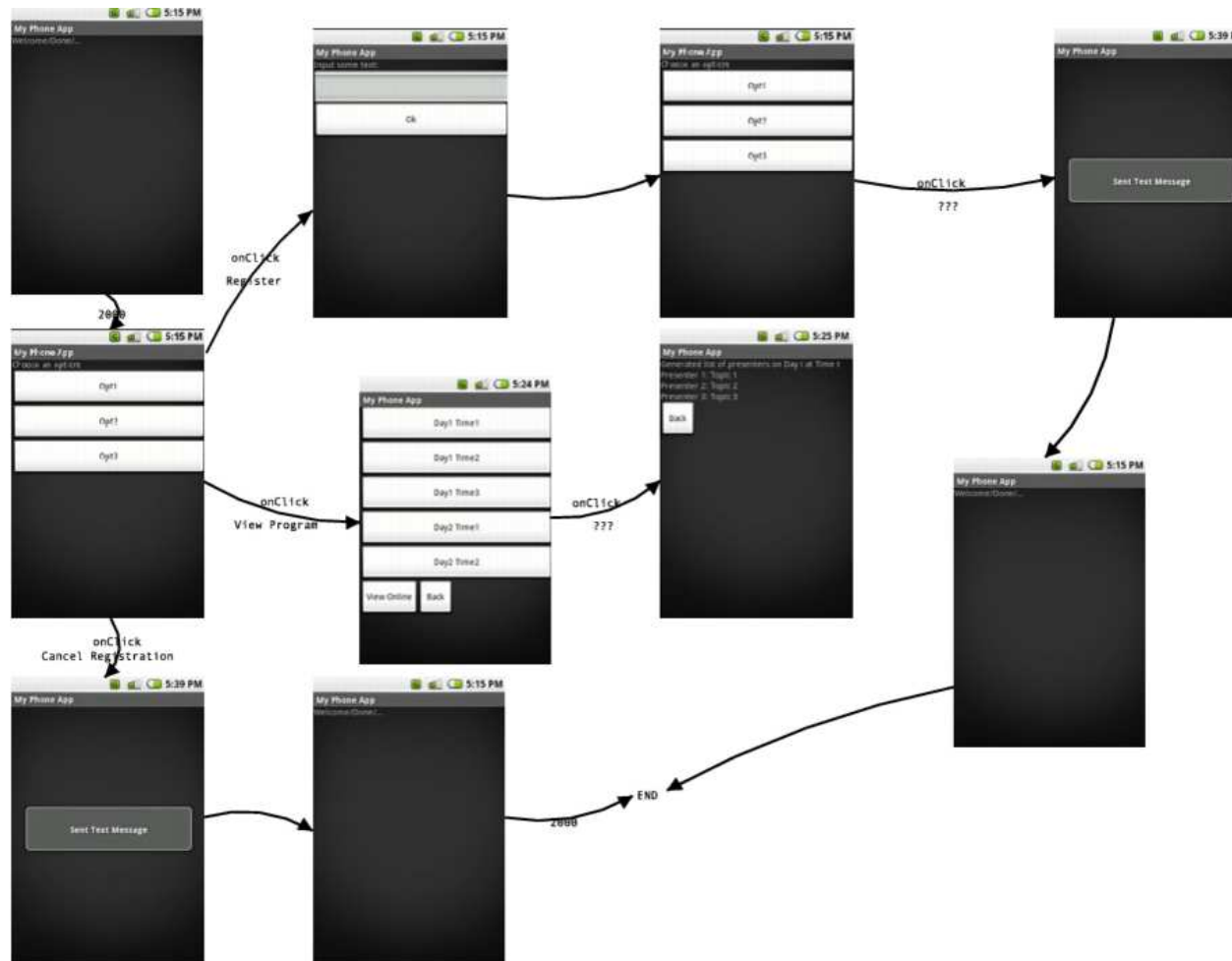
Transformation result: Android target files



... more domain-specific ...



... more domain-specific ... ConferenceRegistration Domain-Specific model



Conclusions and Future Work

- Domain-Specific Modelling: Concrete/Abstract Syntax + **Lattice of Transformations**
- TODO: trace-ability/backward links
- TODO: (meta-)model, transformation evolution
- TODO: embedded systems