

# Graphical formalisms for mapping languages

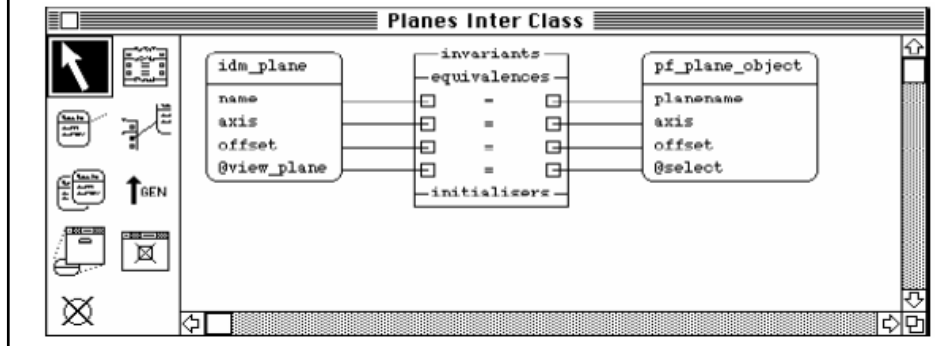
- VML-G
- Forms-based mapper

# Why do we need a graphical formalism?

- Easier to comprehend the mapping?
  - Diagrams can be read by non-experts
- Easier to check that everything is mapped?
  - All attributes for classes from both schemas
  - All combinations of invariants for the same classes
  - All classes in a schema?
- Faster to specify the mappings?
  - Support for mapping equation syntax
- Graphical mapping specification tool can provide validation support and schema management support

# VML-G

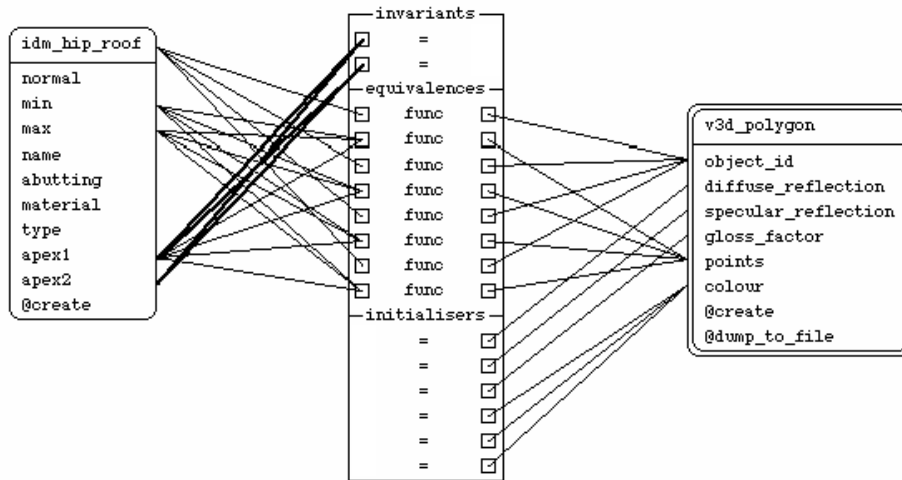
- Provides visualisation of main components within a VML mapping specification
  - Classes
  - inter\_class specification



# VML-G approach

- Icons for classes
  - View all attributes and methods (or subsets)
- Icon for an inter\_class
  - Break up invariants, equivalences, initialisers
  - Equation type is denoted by a symbol
  - Full specification of each equation is viewed in a textbox
  - Full text of the inter\_class specification can be viewed in a text window
- Wiring approach to join attributes and classes to equations in the inter\_class

## VML-G example



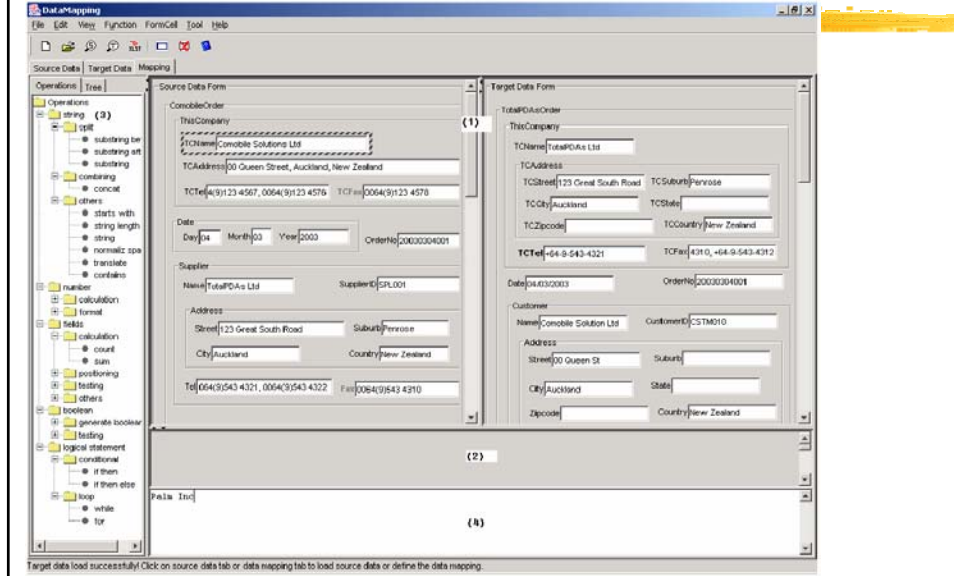
COMPSCI 732 FC §6. Graphical formalisms for mapping languages

## Forms-based mapper

- VML and VML-G are aimed at analyst programmers
- In the real world a business analyst often knows the correspondences between information in different business systems
  - How to get this information out of them?
    - Business people deal with forms containing information
    - Provide a forms-based view of the two representations
    - Allow the business analyst to connect form components together to specify the mapping
    - This specification can be refined by programmers later

COMPSCI 732 FC §6. Graphical formalisms for mapping languages

# Forms-based mapper (XML)



# Drag and drop mapping

One-to-one mapping


One-to-many, function-based mapping

The block contains a sequence of screenshots and numbered steps illustrating the drag-and-drop mapping process:

- (1) Select TCName field in source form:** A screenshot of the source form with the 'TCName' field selected.
- (2) Drag TCName to Name field in target form:** A screenshot showing the 'TCName' field being dragged from the source to the 'Name' field in the target form.
- (3) Formula shows on target form after mouse released:** A screenshot showing the target form with a formula editor open, displaying the mapping logic.
- (4) Sample mapping result shows on target field after mouse is clicked again:** A screenshot showing the target form with the 'Name' field populated with the mapped value.

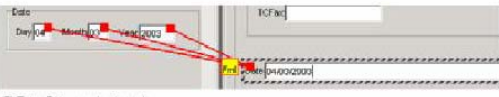
Additional screenshots show dialog boxes for selecting fields and formulas, and a final screenshot showing the result of the mapping.

# Drag and drop mapping



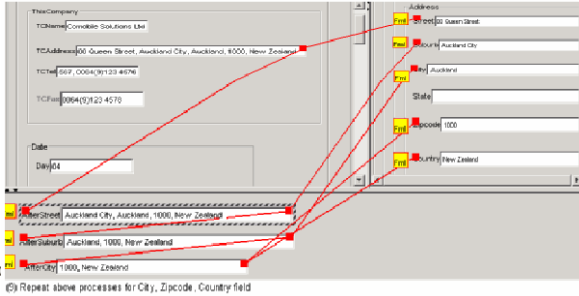
(4) Formula shows on Date field on target field

Many-to-one attribute mapping



(5) Enter Return and get result

Visualising temporary elements



COMPSCI 732 FC §6. Graphical f

© Repeat above process for City, Zipcode, Country field

# Graphical formalisms

- Adjunct to the textual mapping notation
  - Provide a more comprehensible view of a mapping
  - Enable greater checking of mapping specifications
  - Tailorable to specifier specific notations