A Domain-Specific Visual Language for Report Writing

Using Microsoft DSL Tools

- Ruskin Dantra, John Grundy and John Hosking

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

Overview

- Introduction
  - Domain
  - Problem
- Background
  - Motivation
- Approach and design
- Evaluation
- Future work

Domain Introduction

- In conjunction with Prism
  - MIS for the printing and graphics industry
- Prism exposes a proprietary reporting language via which end-users can query their database and display the results in a visual form
- Language is called RWL for the purpose of this paper
- Procedural, “sort-of” object-oriented, interpreted programming language

Background and Motivation

- RWL is complicated
  - Implicit semantics
  - Complicated enterprise database
- No design time validation
- No dedicated IDE
  - Lack of context sensitive help
- Ease change management
  - Time to market

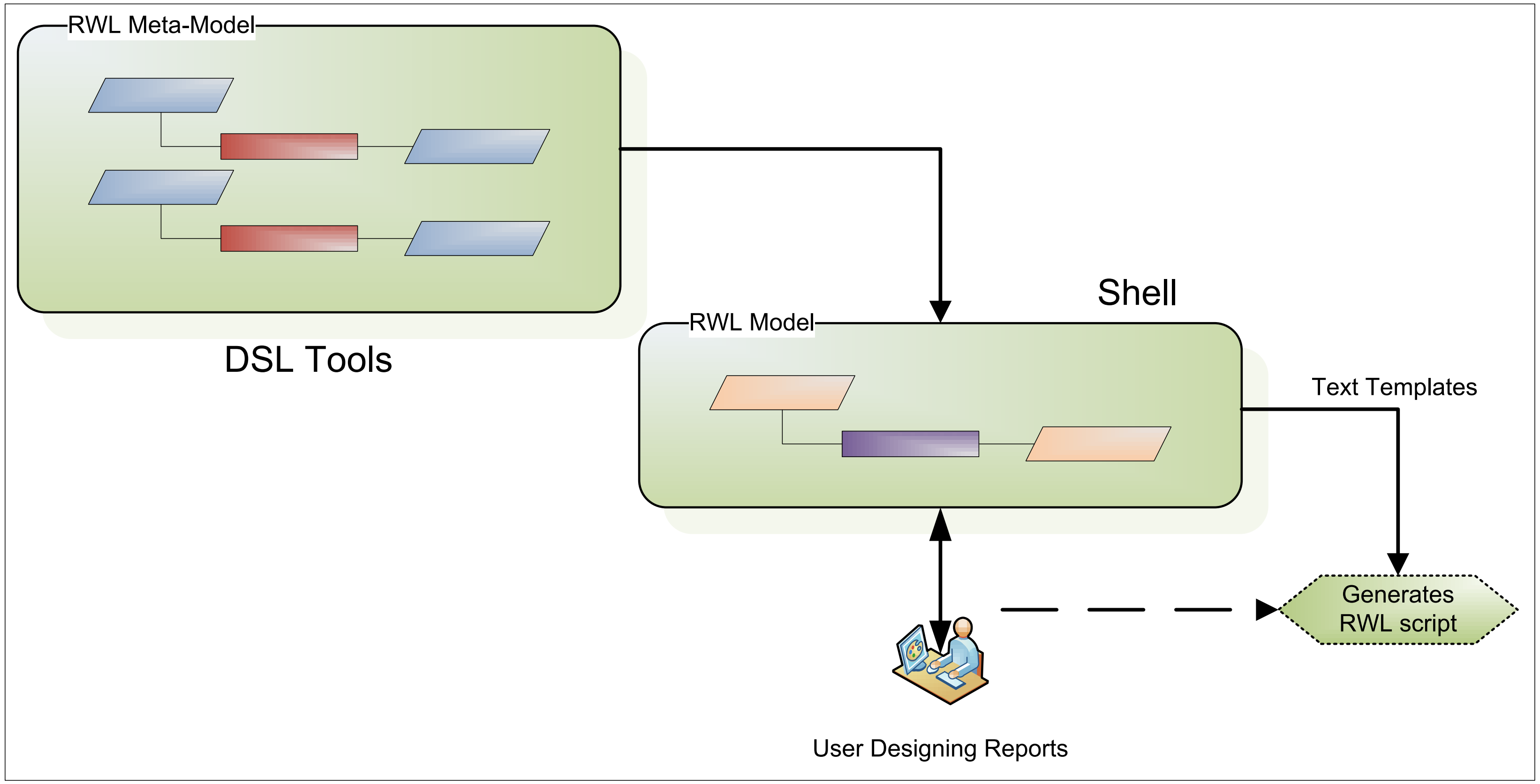
```
Code CASE_STUDY_1
Type Standard
Access JTSR

Scan RM
  Print RM_CUST + RM_NAME;
  Print "All Jobs For "  + RM_NAME;
  Scan QM
      Choose (QM_CUST_CODE, MATCH, RM_CUST)
      Choose(QM_QUOTE_JOB, MATCH, QMM_JOB)

      Print QM_JOB_NUM + QM_TITLE;
  End
End
Print StandarReportFooter;
```

Why a VL?

- Visual aid, cues and context sensitive help
  - Visual DSL
- Only expose “absolutely necessary” information
- Minimize user errors by allowing them to “design” reports rather than write them

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

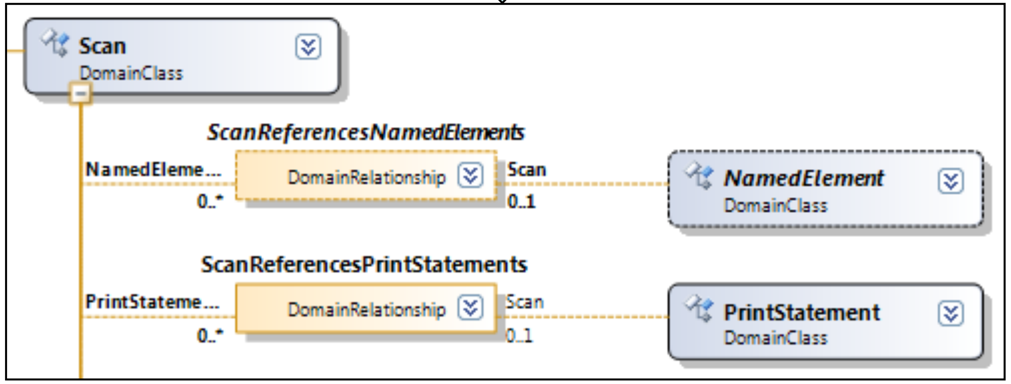
Approach overview



Our Approach

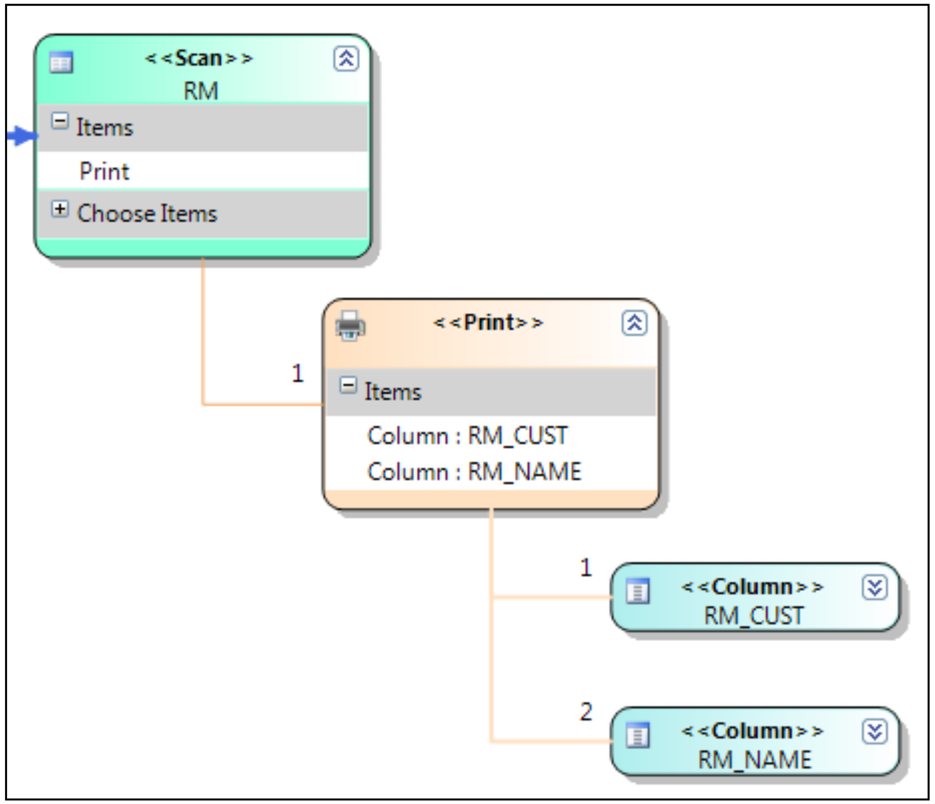
1. Reverse engineer meta-model from RWL specification

```
Scan RM
  Print RM_CUST + RM_NAME;
End
```



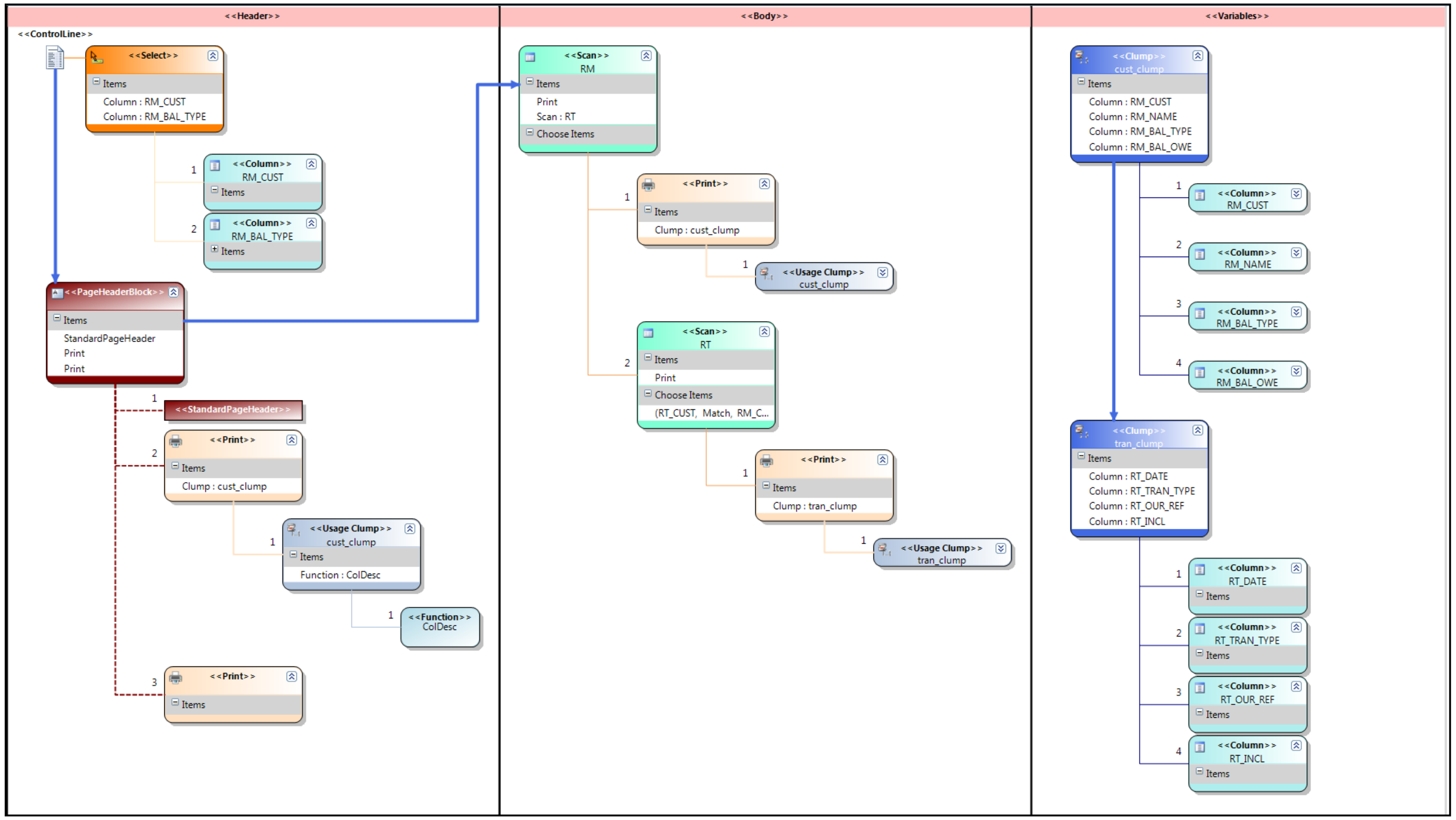
Our Approach (Cont'd…)

2. Design a VL using surface level notation



Our Approach (Cont'd…)

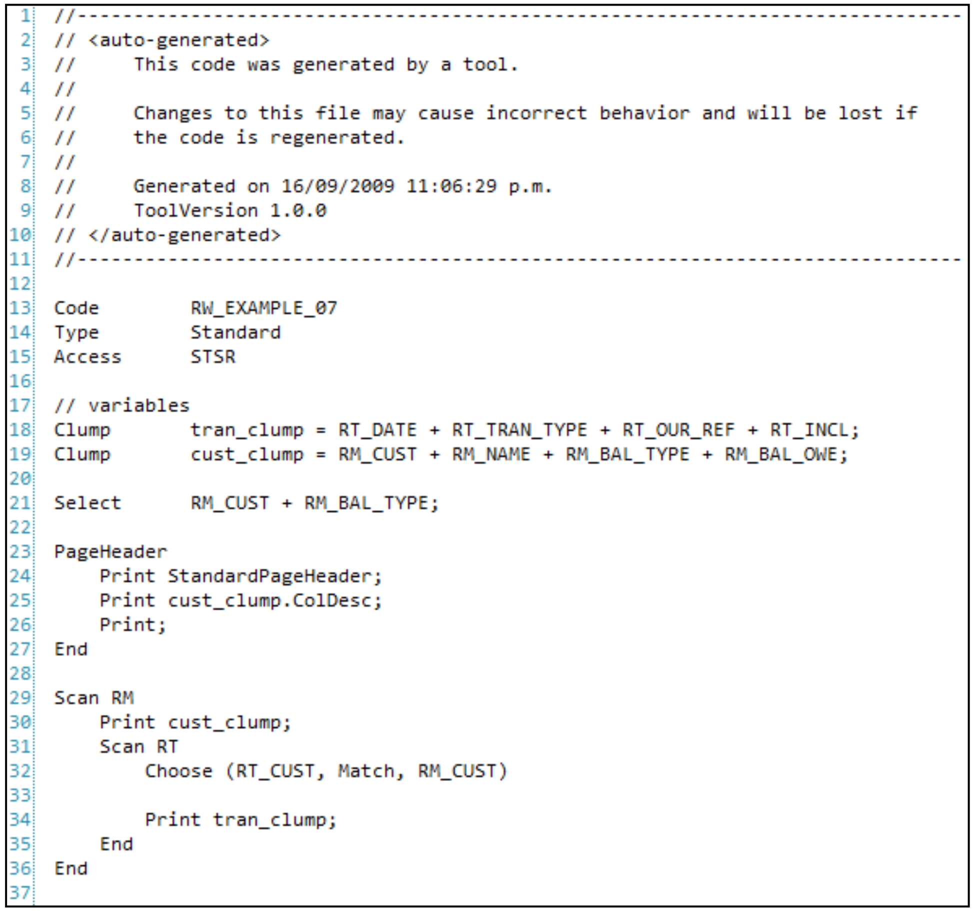
3. Allow end-users to create entire RWL models



THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

Our Approach (Cont'd…)

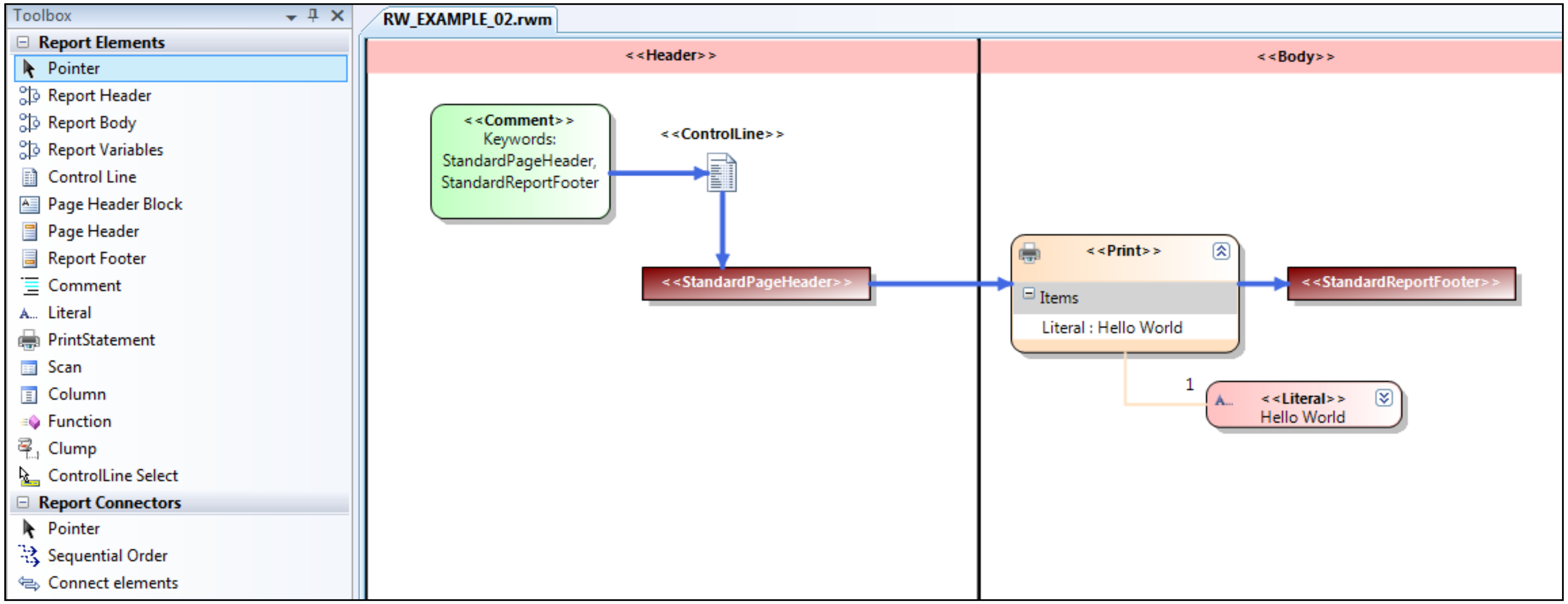
4. Automated RWL script generation from RWL model

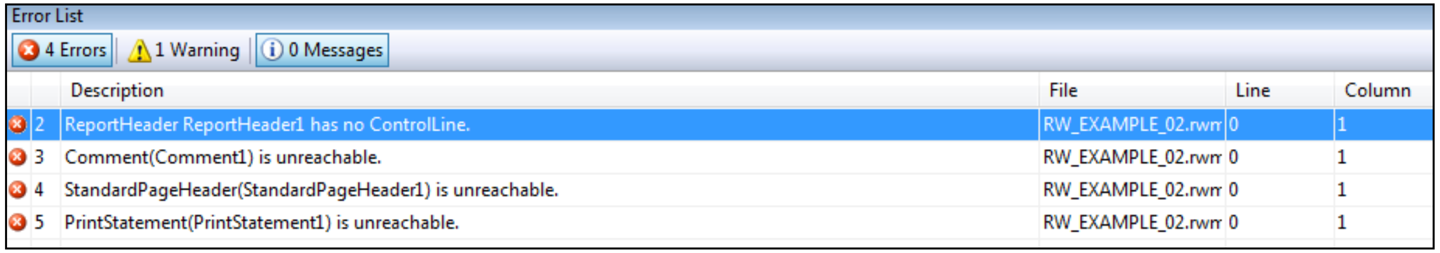


```
//-----------------------------------------------------------------------------
// <auto-generated>
//      This code was generated by a tool.
//
//      Changes to this file may cause incorrect behavior and will be lost if
//      the code is regenerated.
//
//      Generated on 16/09/2009 11:06:29 p.m.
//      ToolVersion 1.0.0
// </auto-generated>
//-----------------------------------------------------------------------------

Code        RW_EXAMPLE_07
Type        Standard
Access      STSR

// variables
Clump       tran_clump = RT_DATE + RT_TRAN_TYPE + RT_OUR_REF + RT_INCL;
Clump       cust_clump = RM_CUST + RM_NAME + RM_BAL_TYPE + RM_BAL_OWE;

Select      RM_CUST + RM_BAL_TYPE;

PageHeader
    Print StandardPageHeader;
    Print cust_clump.ColDesc;
    Print;
End

Scan RM
    Print cust_clump;
    Scan RT
        Choose (RT_CUST, Match, RM_CUST)

        Print tran_clump;
    End
End
```

THE UNIVERSITY OF AUCKLAND
FACULTY OF ENGINEERING

Simple Example



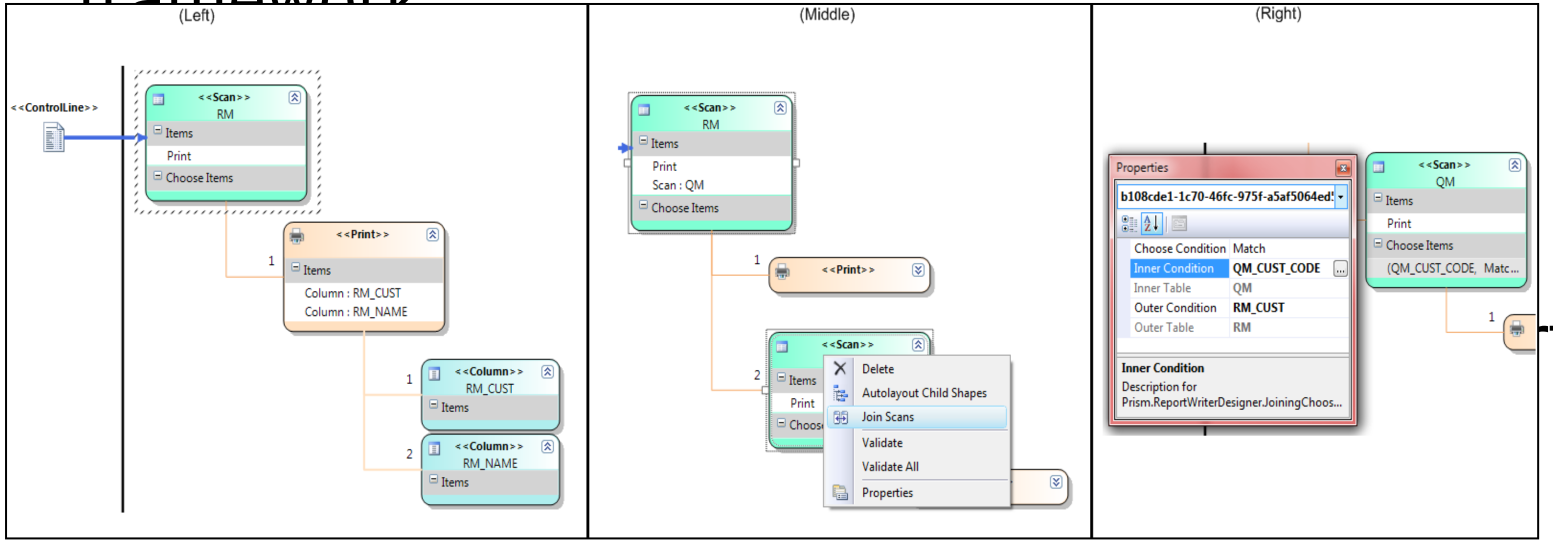

Error List

4 Errors | 1 Warning | 0 Messages

		Description	File	Line	Column
✕	2	ReportHeader ReportHeader1 has no ControlLine.	RW_EXAMPLE_02.rwm	0	1
✕	3	Comment(Comment1) is unreachable.	RW_EXAMPLE_02.rwm	0	1
✕	4	StandardPageHeader(StandardPageHeader1) is unreachable.	RW_EXAMPLE_02.rwm	0	1
✕	5	PrintStatement(PrintStatement1) is unreachable.	RW_EXAMPLE_02.rwm	0	1

Evaluation – Design

- Evaluated using the Cognitive Dimensions framework



Properties

b108cde1-1c70-46fc-975f-a5af5064ed!

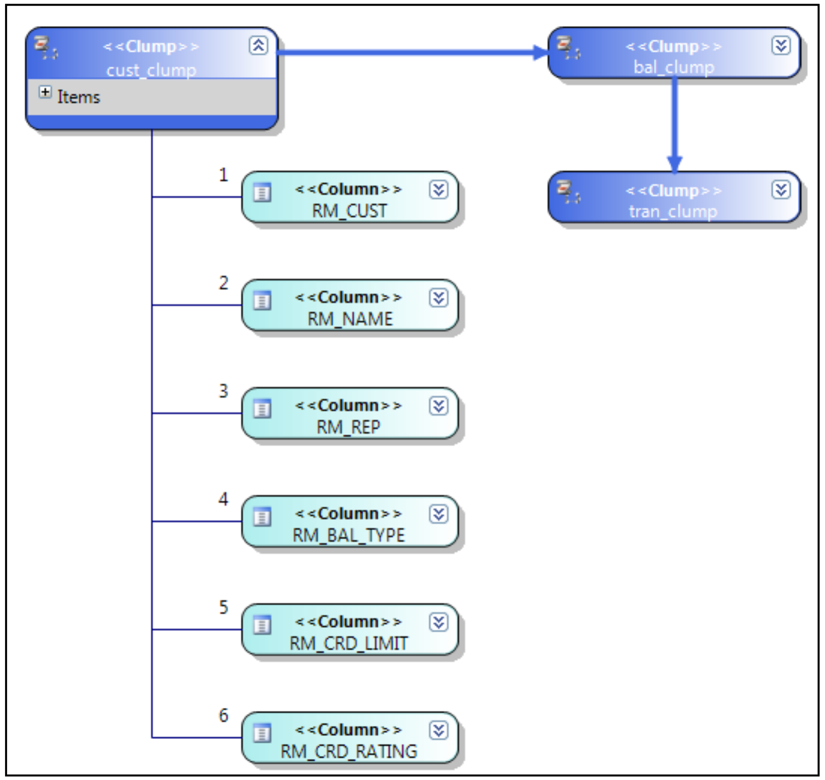
Choose Condition	Match
Inner Condition	QM_CUST_CODE
Inner Table	QM
Outer Condition	RM_CUST
Outer Table	RM

Inner Condition
Description for Prism.ReportWriterDesigner.JoiningChoos...

t

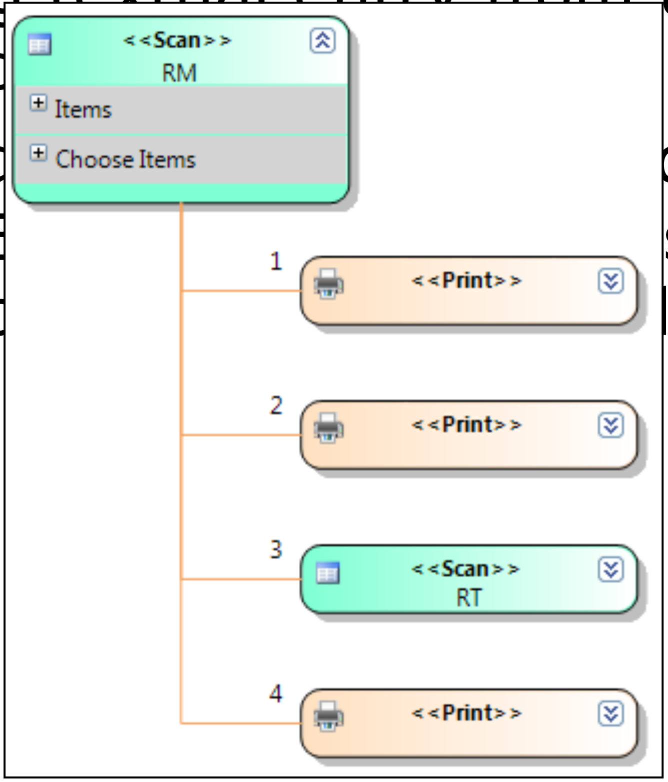
Evaluation – Design (Cont'd…)

- Auto layout and simple refactoring capabilities reduce viscosity



Evaluation – Design (Cont'd...)

- Reducing textual entry from users reduces error pro
- Hidden d d by doing simple things. E ships using connecto lements



Evaluation – Survey

- 14 participants
- Six developers and eight non-technical end-users
- Both groups given two task: an easy task and a comparatively harder task

Evaluation – Survey (Cont'd…)

- End-user survey
  - Novice and intermediate users found tool useful
  - Experienced users felt a bit constrained
  - Scalability was raised as an issue
    - At what point does a visual report become harder to understand than a textual report?
  - Making small changes require comparatively more steps
    - E.g. Adding a simple *print* statement

Evaluation – Survey (Cont'd…)

- Developer survey
  - Tasks were easy and procedural with automated code generation
  - Scalability was raised as an issue
    - Will the meta-model become so large than maintenance will be difficult?

Future Work

- Improve VL
- Improve auto-layout algorithms
- Versioning
- Wizards and code snippets

The End

- Questions?