Using the Universal XML Exchange

Mainframe and Multi-Platform Application Development

MP207SN

caWorld'08 | discoverability

Abstract

Universal XML Exchange for CA Repository

- Gives new opportunities for software architects and DBAs to more effectively integrate metadata from multiple sources
- Enables enhanced management and understanding of complex object oriented software assets
- Show Semantic Designs' experiences with XML exchange during the development of an OO source scanner for Java
- Provide practical insight into source scanning, metadata, and exchange technology use

Biography

> Chad Wellington
Application Engineer, Semantic Designs, Austin
software engineering tools, program analysis and transformation

> M.S. University of Texas, Austin
B.S. Texas A&M University, College Station
Computer Science
algorithms, computation theory, program semantics, theorem proving

Agenda

> Software change is hard

> Mitigation through CA Repository

> Hurdles to wider application

> XML Exchange solution

> Q&A

Perspective on Software Management

SD® | caWorld® 08

The Problem

> Software change is hard, dangerous, and necessary

> Hard from understandability

- Current state difficult to ascertain

> Dangerous from unpredictability

- Correctness is fragile
- Small alterations cause large results

> Necessary from competition

- Environment, culture, regulation
- Technology moves fast

The Solution

> Relationships create understanding

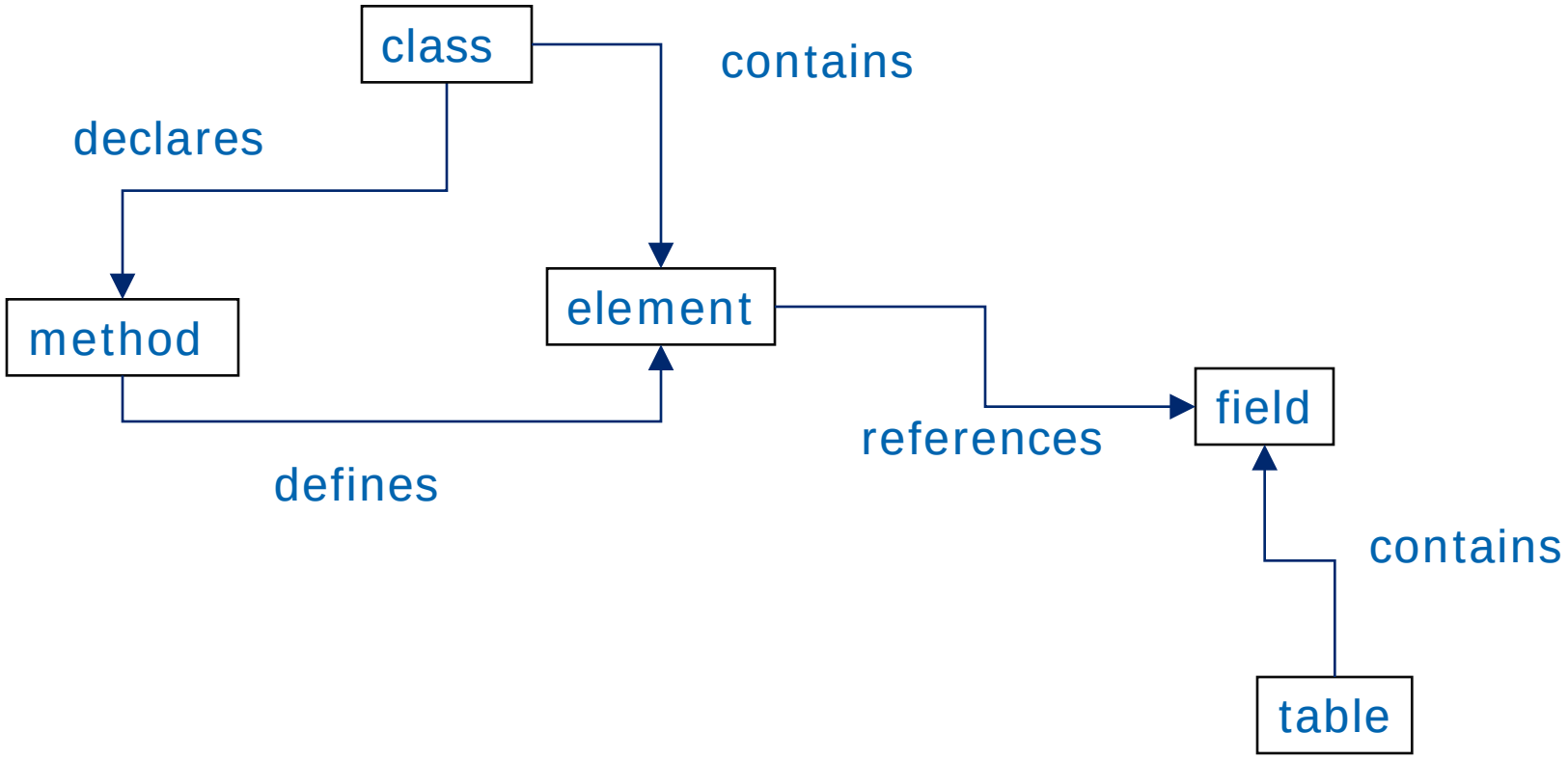
- Containment
  - Location
  - Derivation
- Information flow
  - Within domain (control flow)
  - Across domain boundaries (interfaces)

> Understanding change alleviates risk

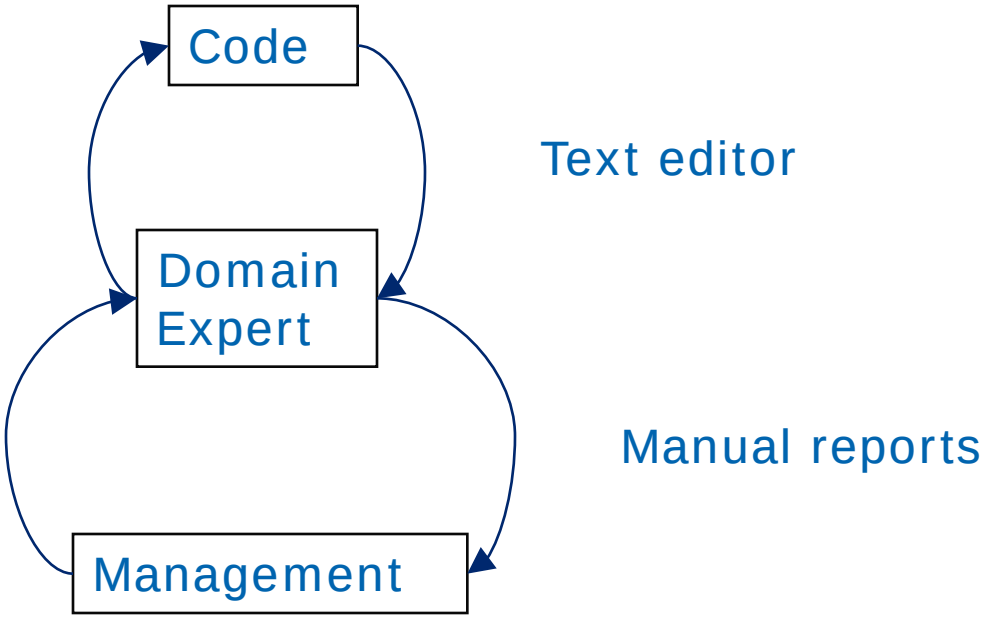
- Impact analysis
- Cost, complexity metrics

Sample Metadata

Object-Oriented and Database Relations



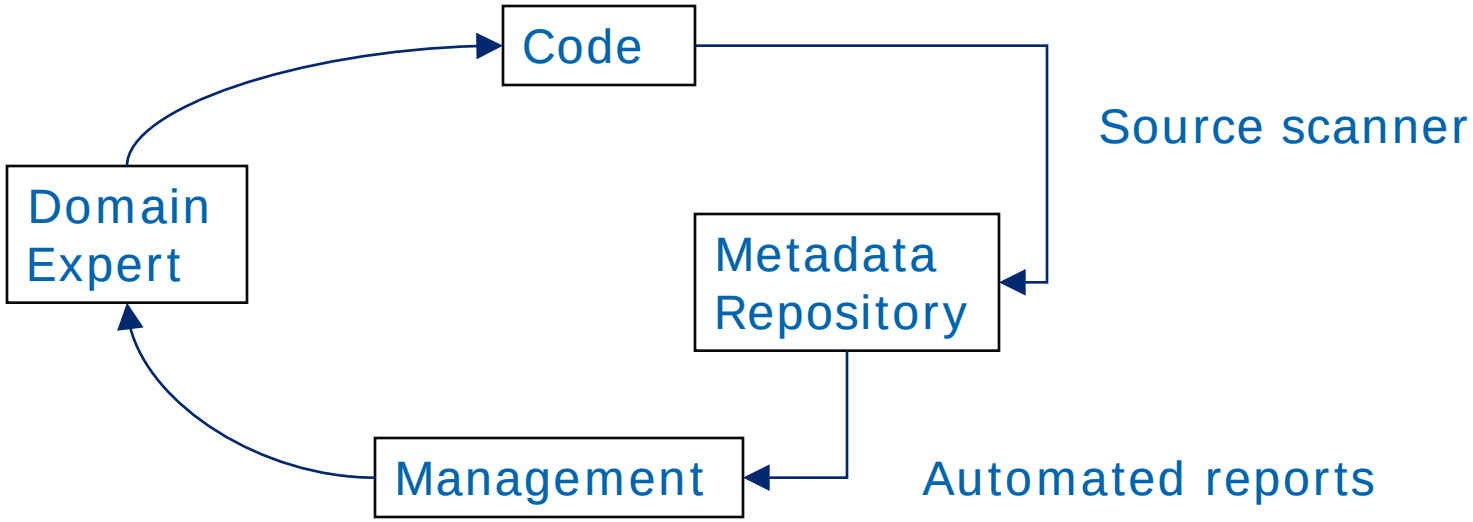
Unassisted Methodology



> Benefits: flexible, customizable, always available

> Drawbacks: slow, inefficient

Automated Methodology

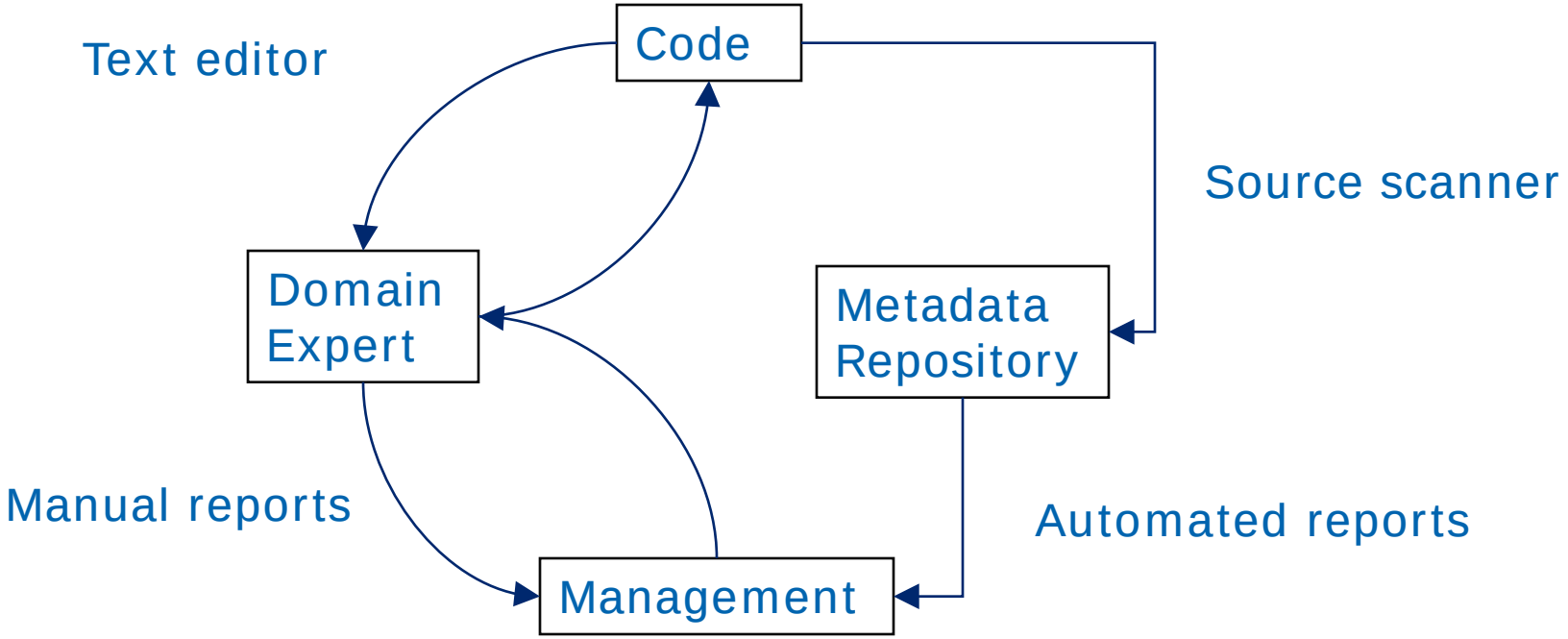


> Benefits: efficient, large-scale, cross-domain

> Drawbacks: limited domains, difficult to change

Hybrid Methodology

> Limitations in source scanning lead to hybrid approach



Two Hurdles
To Better CA Repository Integration

> Domain problem

- Can it scan a given domain?
- Can it scan all of the domains in my system?

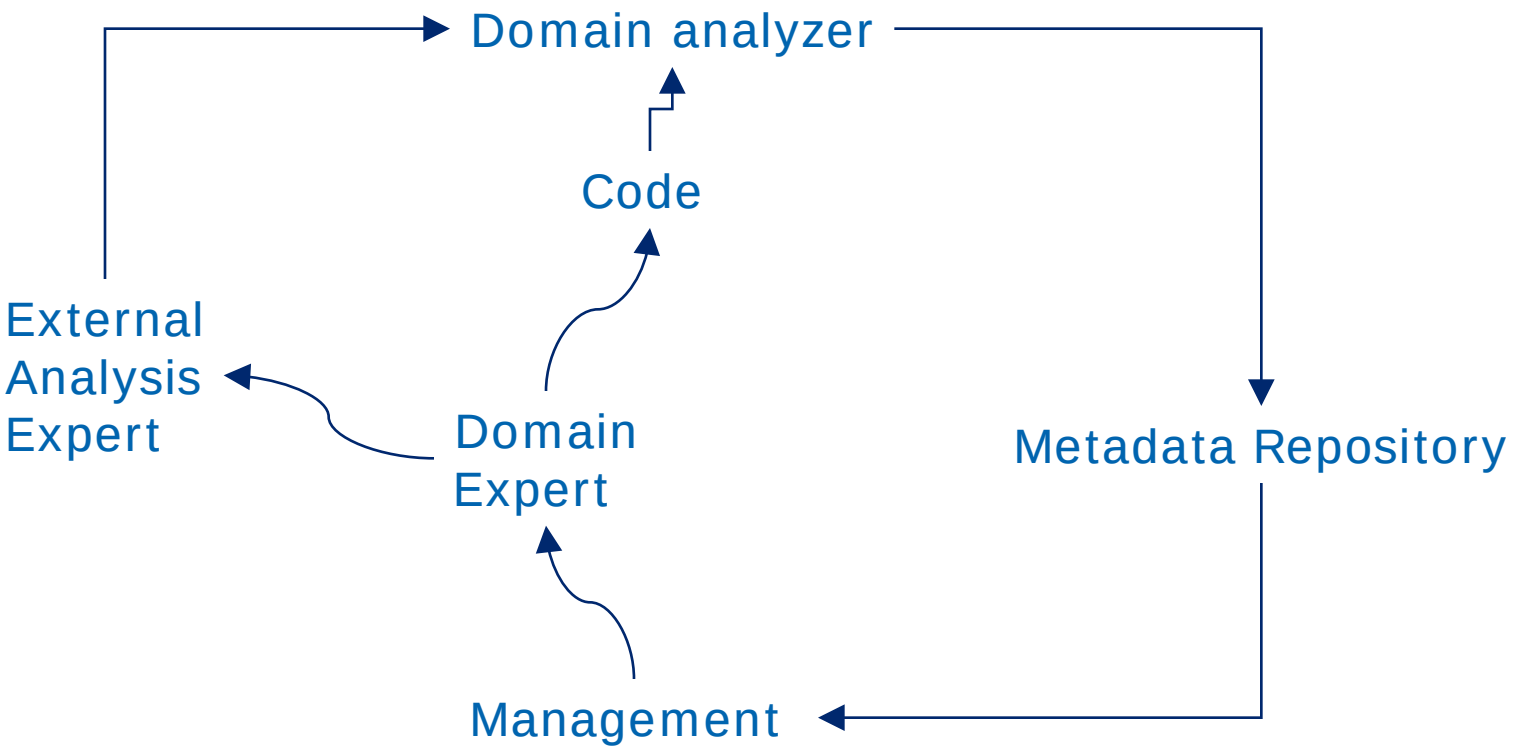
> Value problem

- Does it provide the properties I need?
- Is it worth partial domain coverage?

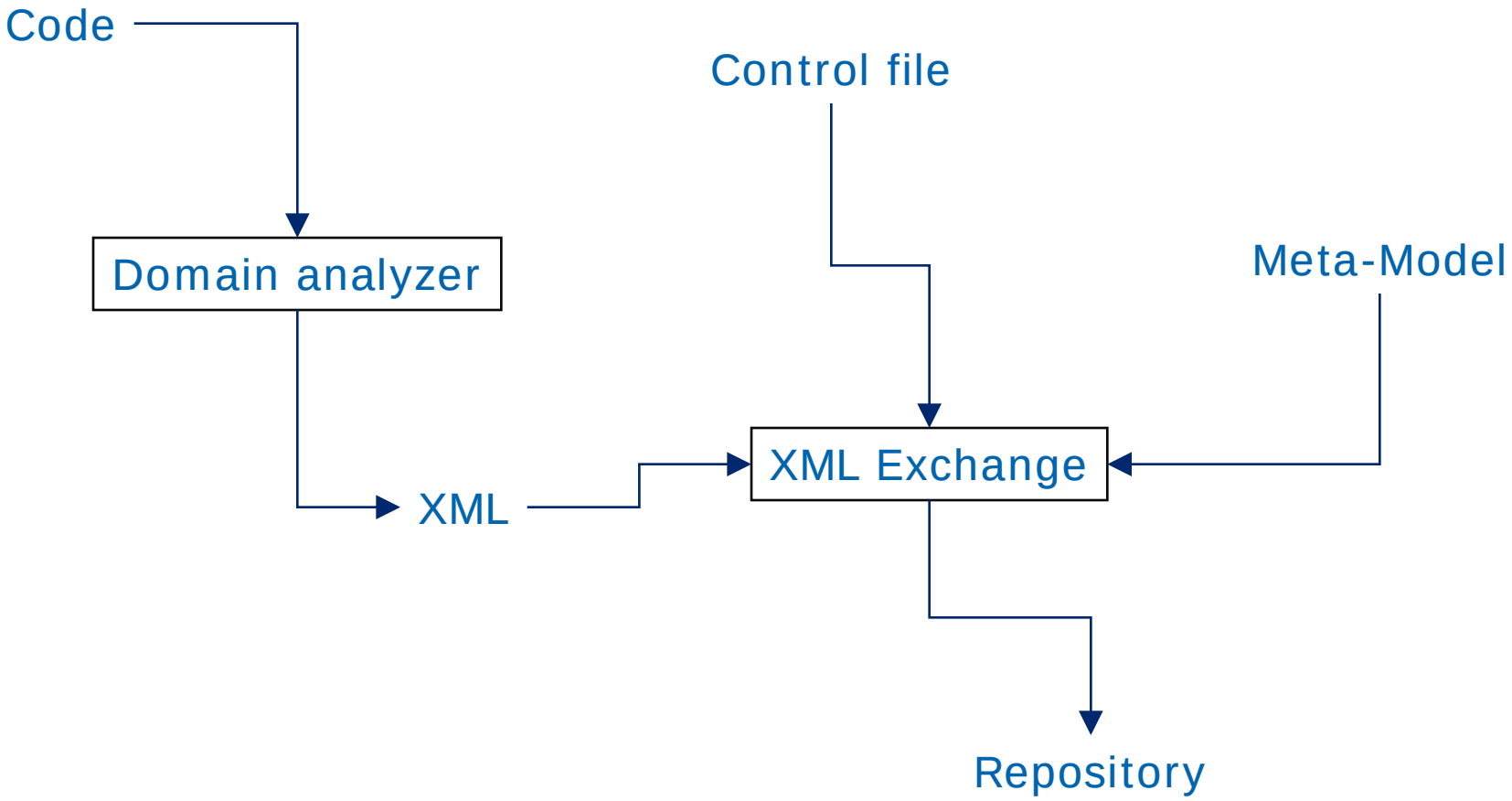
Universal XML Exchange

SD® | caWorld® 08

Unified Methodology



XML Exchange Overview



Sample Workflow: Java

```
public class Num
{
        private int currentVal;

        public void setVal(int newVal)
        {
                currentVal=newVal;
        }
        public int getVal()
        {
                return currentVal;
        }
        protected void doubleVal()
        {
                currentVal*=2;
        }
}
                                                        1,1          Top
```

```
<CARepository_OOModel id="75696144" >

<CLASS Name="Num" Visibility="public" id="75778320" >

<FUNCTION Name="doubleVal" Return_Data_Type="void" Visibility="protected" id="75779344" >
</FUNCTION>

<FUNCTION Name="getVal" Return_Data_Type="int" Visibility="public" id="75779088" >
</FUNCTION>

<FUNCTION Name="Num" Return_Data_Type="void" Visibility="public" id="75778448" >
</FUNCTION>

<FUNCTION Name="setVal" Return_Data_Type="void" Visibility="public" id="75778832" >
<PARM Name="newVal" Parm_Type="int" id="76041152" >
</PARM>
</FUNCTION>

<ELEMENT Name="currentVal" Value_Type="int" Visibility="private" id="75778576" >
</ELEMENT>
</CLASS>

</CARepository_OOModel>
Exiting with final status 0

C:\DMS\Domains\Java\Java1_4\Tools\RepositoryPrinter>
```

```
<?xml version="1.0" encoding="UTF-8" ?>
- <CARepository_OOModel id="75696144">
  - <CLASS Name="Num" Visibility="public" id="75778320">
      <FUNCTION Name="doubleVal" Return_Data_Type="void" Visibility="protected" id="75779344" />
      <FUNCTION Name="getVal" Return_Data_Type="int" Visibility="public" id="75779088" />
      <FUNCTION Name="Num" Return_Data_Type="void" Visibility="public" id="75778448" />
    - <FUNCTION Name="setVal" Return_Data_Type="void" Visibility="public" id="75778832">
        <PARM Name="newVal" Parm_Type="Input" id="76041152" Data_Type="int" />
      </FUNCTION>
      <ELEMENT Name="currentVal" Value_Type="int" Visibility="private" id="75778576" />
    </CLASS>
  </CARepository_OOModel>
```

Done | My Computer | 100%

CA Repository for Distributed Systems - Client - Exploration View - OO Model Objects - CLASS - Num

File Help

Home Back Search Help

- Class:Num
  - Object Oriented Usage:currentVal.Num
  - Object Oriented Usage:newVal.Num
  - Class Has Operation:Num.doubleVal
  - Class Has Operation:Num.getVal
  - Class Has Operation:Num.Num
  - Class Has Operation:Num.setVal
  - is contained by Object Group:testOG

Property Name	Property Value
Object Type:	Class
Name	Num
Status	DEV
Version	0
Cardinality	
Created By	CWellington
Create Date Time	2008-09-29-17.14.15.000000
Modified By	CWellington
Modified Date Time	2008-09-29-17.14.15.000000
Description	
Class Alias	

CA Repository for Distributed Systems - Client - Exploration View - OO Model Objects - ELEMENT - currentVal

File Help

Home Back Search Help

- Business Data Element:currentVal
  - Object Oriented Usage:currentVal.Num
  - is contained by Object Group:testOG

Property Name	Property Value
Object Type:	Business Data Element
Name	currentVal
Status	DEV
Version	0
Case Created Time	
Default Cobol Name	
Default C Name	
Default Data Type	

Exchange Solution

Change Scanning Bottleneck Into Domain Advantage

> Domain adaptability

- Unifies management procedures
- Immediate adoption
- Unharmed by technology shifts

> Cross-domain analysis

- Optimizes return on CA Repository investment
- Full automation of property derivation

Domain Sampler

Java 1.1-1.5
C (ANSI, GNU, MS)
C++
C#

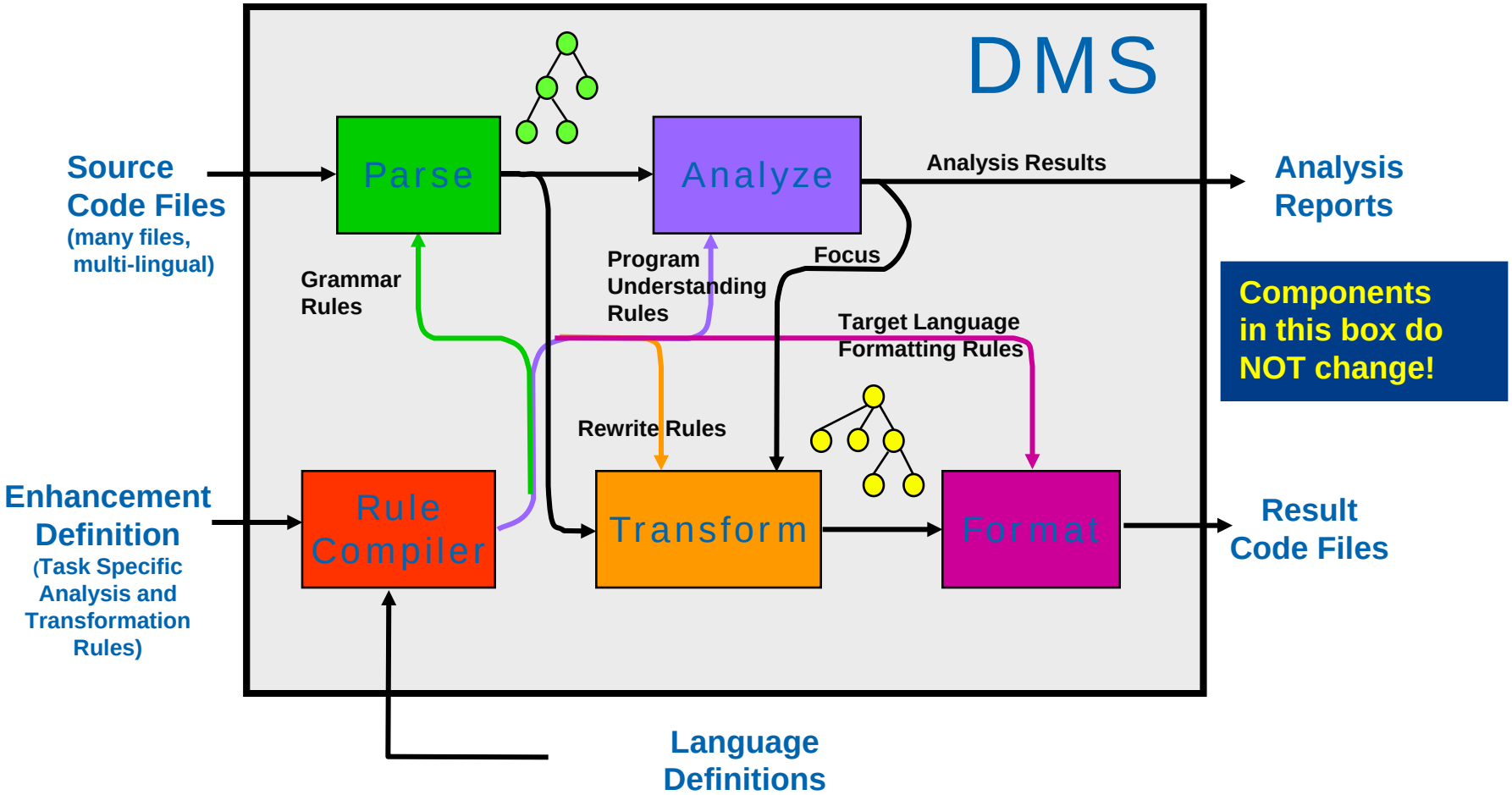
PL/SQL
SQL (ANSI SQL2, Oracle8)
Progress (4GL)

PL/1
FORTRAN (95,90,77)
COBOL (ANSI 1985, AS400)
Pascal/Delphi
IBM JCL

HTML
PHP (3,4,5)
ECMAscript/JavaScript
VBScript, ASP
Visual Basic 6

UML
Rational Rose .MDL
IDL (Corba 2.3)

DMS® S/W Reengineering Toolkit



(Grammar Rules + General Analysis Rules + Formatting Rules for 20+ languages or custom, including Assembly, C, C#, C++, COBOL, Java)

Summary

> XML Exchange enables the CA Repository to

- Jump the hurdles to wider application
- Eliminate tradeoff between technological progress and manageability

> Specialized program analysis gives

- Multi-domain support
- Custom property derivation

Q&A

“Mix and Munch”

Networking with Your Peers

Join your peers and CA to mingle and network over lunch.

In this informal setting, engage in CA solution discussions, share ideas, and establish new contacts. A buffet lunch will be provided. Seating is limited and will be on a first come, first served basis.

Where? San Polo 3405-6, 3505-6

When? Tuesday and Wednesday

Time? 12:00pm - 1:15pm

Exhibition Center
Related CA and Partner Technology

> CA

- Mainframe 428 – CA Repository solutions

> Partner

- Booth 436 — ADC Austin/Semantic Design

> Exhibition Center Tours

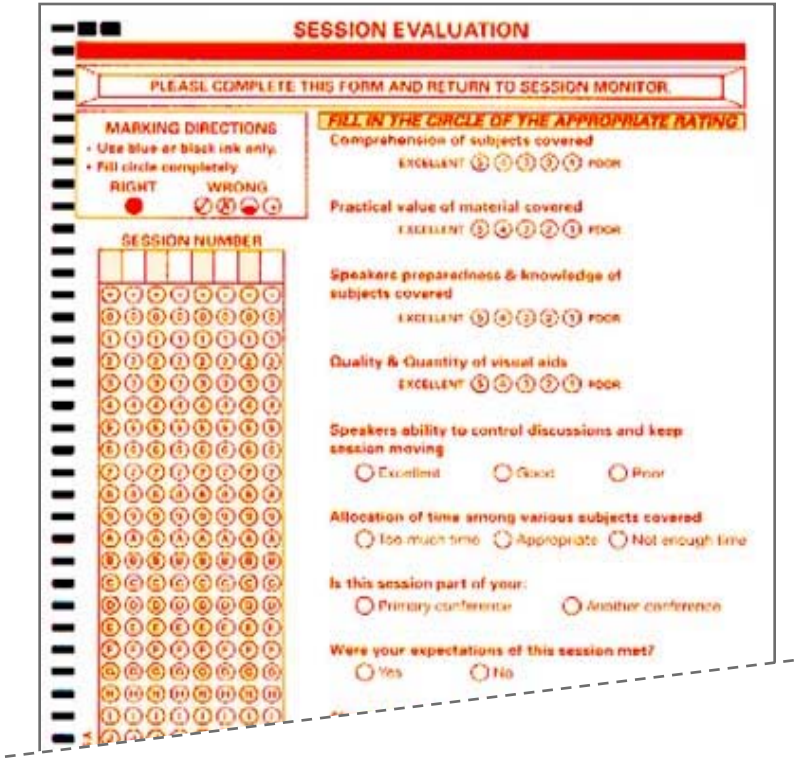
- Sign up at the Info Desk in the Exhibition Center

Please Complete a Session Evaluation Form

> The number for this session is **MP207SN**

> After completing your session evaluation form, place it in the basket at the back of the room

- Please left-justify the session number


SESSION EVALUATION

PLEASE COMPLETE THIS FORM AND RETURN TO SESSION MONITOR.

MARKING DIRECTIONS
- Use blue or black ink only.
- Fill circle completely.

RIGHT WRONG

SESSION NUMBER

FILL IN THE CIRCLE OF THE APPROPRIATE RATING

Comprehension of subjects covered
EXCELLENT 5 4 3 2 1 POOR

Practical value of material covered
EXCELLENT 5 4 3 2 1 POOR

Speakers preparedness & knowledge of subjects covered
EXCELLENT 5 4 3 2 1 POOR

Quality & Quantity of visual aids
EXCELLENT 5 4 3 2 1 POOR

Speakers ability to control discussions and keep session moving
Excellent Good Poor

Allocation of time among various subjects covered
Too much time Appropriate Not enough time

Is this session part of your:
Primary conference Another conference

Were your expectations of this session met?
Yes No

 Copyright © 2008 CA. All rights reserved.