

Cobol to Java Translation

Steve O'Hara

CS 6463 Program Analysis

May 6, 2008

Cobol to Java

Input Cobol

```
PERFORM 2000-C1 THRU  
2000-EXIT UNTIL  
ATTACH-SUB > 36
```

```
PERFORM 3500-ISRT  
5 TIMES.
```

Output Java

```
do  
{  
    FN_2000_C1();  
    FN_2000_C2();  
    FN_2000_C3();  
    FN_2000_EXIT();  
} while (ATTACH_SUB <= 36);  
  
for (int jj=0; jj<5; jj++)  
{  
    FN_3500_ISRT();  
}
```

Approach #1 : Manual (C#)

```
COBOL cobol = new COBOL();  
COBOL.Program cp = cobol.parseProgram(inFile);  
cobol.printProgram(Console.Out, cp);
```

```
COBOL_to_JAVA c2j = new COBOL_to_JAVA();
```

```
JAVA java = new JAVA();  
JAVA.Program jp = c2j.trans(cp);  
java.printProgram(Console.Out, jp);
```

Approach #2 : Grammar (POET)

```
<code CobolPerformThru pars=(proc : CobolExp, thru : CobolExp)>  
    PERFORM @proc@ THRU @thru@  
</code>
```

```
<xform TransPerformNTimes pars=(input)>  
    repl="";  
    foreach (input : (perform = CobolPerformNTimes#(proc, n)) {  
        proc = "FN_" proc;  
        repl = BD_list( (perform, JavaCallNTimes#(proc,n)), repl);  
    }  
    REPLACE(repl,input) ;  
</xform>
```

Comparison

- Input scanning (advantage to Manual)
- Input grammar changes (huge advantage to Grammar)
- Adding a new input or output language (advantage to Grammar)
- Complexity (advantage to Manual)
- Debugging (advantage to Manual)
- Verification (huge advantage to Grammar)