SaM Code Generation

Part 2 – Actual SaM Code

What generates SaM Code?

- Functions
- Program
- Statements
- Expressions
- Expression Parts

- Just about every grammar element

The Templates?

- How you know what SaM Code each grammar element generates
- Simple
  - Read
- Dependent
  - Addition
- Compiler Dependent
  - Variable Reference

Example – Printing a Constant

```
public void printSamCode(PrintWriter out) {
    if (type.equals(DataType.INT))
        out.println("PUSHIMM " + value);
    else if (type.equals(DataType.BOOLEAN))
        out.println("PUSHIMM " +
          (((Boolean)value).booleanValue() ? 1 : 0));
    else if (type.equals(DataType.CHAR))
        out.println("PUSHIMMCH '" + value.toString().replace("'",
          "\\'") + "'");
}
```

Java code to emit SaM Code (Example)

```
public void printSamCode(PrintWriter out) {
    ifExpr.printSamCode(out);
    int number = getNumber();
    out.println("JUMPC thenif" + number);
    if (elseStatement != null)
        elseStatement.printSamCode(out);
    out.println("JUMP endif" + number);
    out.println("thenif" + number + ":");
    thenStatement.printSamCode(out);
    out.println("endif" + number + ":");
}
```

```
code for e1
JUMPC thenifLabel
code for s2
JUMP endifLabel
thenifLabel:
code for s1
endifLabel:
```

SaM Code Semantics

Checking Semantics

What are semantics?

- Semantic errors are code that is valid according to the grammar rules, but is invalid based on some reason.
- Examples
  - Type mismatch (assignment to variable of a different type)
  - Addition of Boolean variables/values
  - Calling a function with an incorrect number of parameters
  - Referencing/Assigning an undeclared variable