Writing JCL to Compile, Link, and Run a Program



Dave NicoletteSoftware Developer

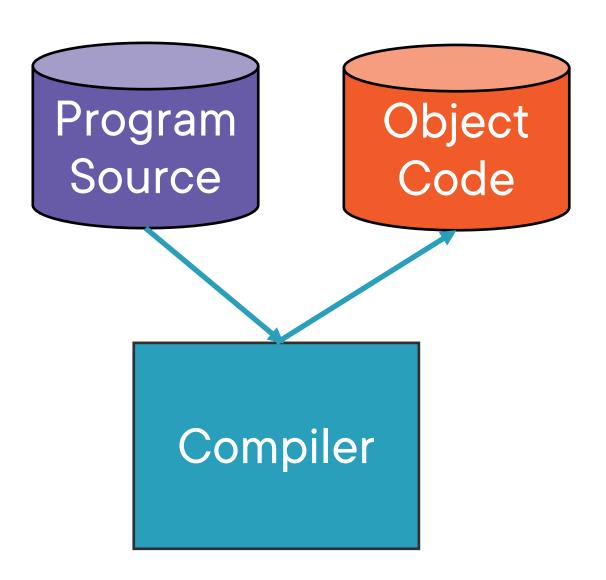
@davenicolette neopragma.com

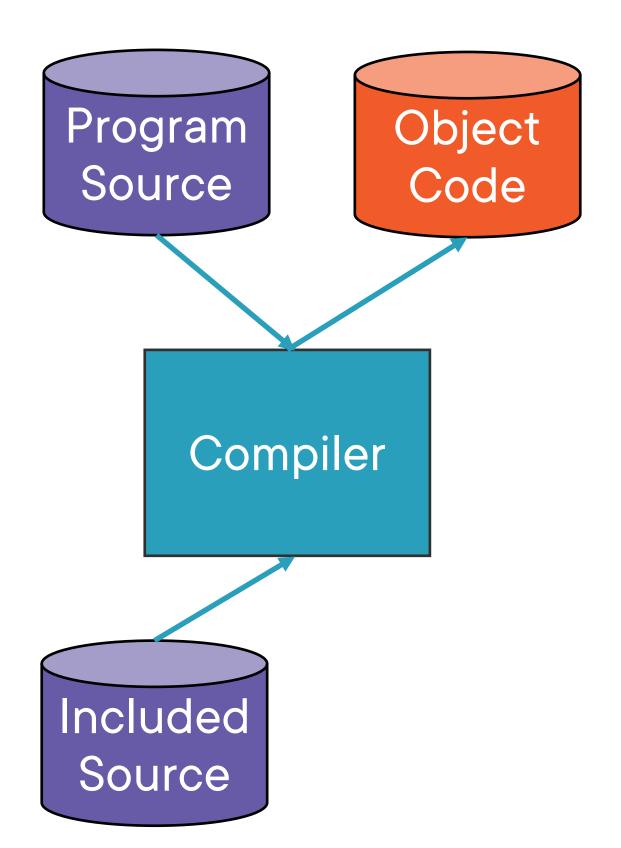
Overview

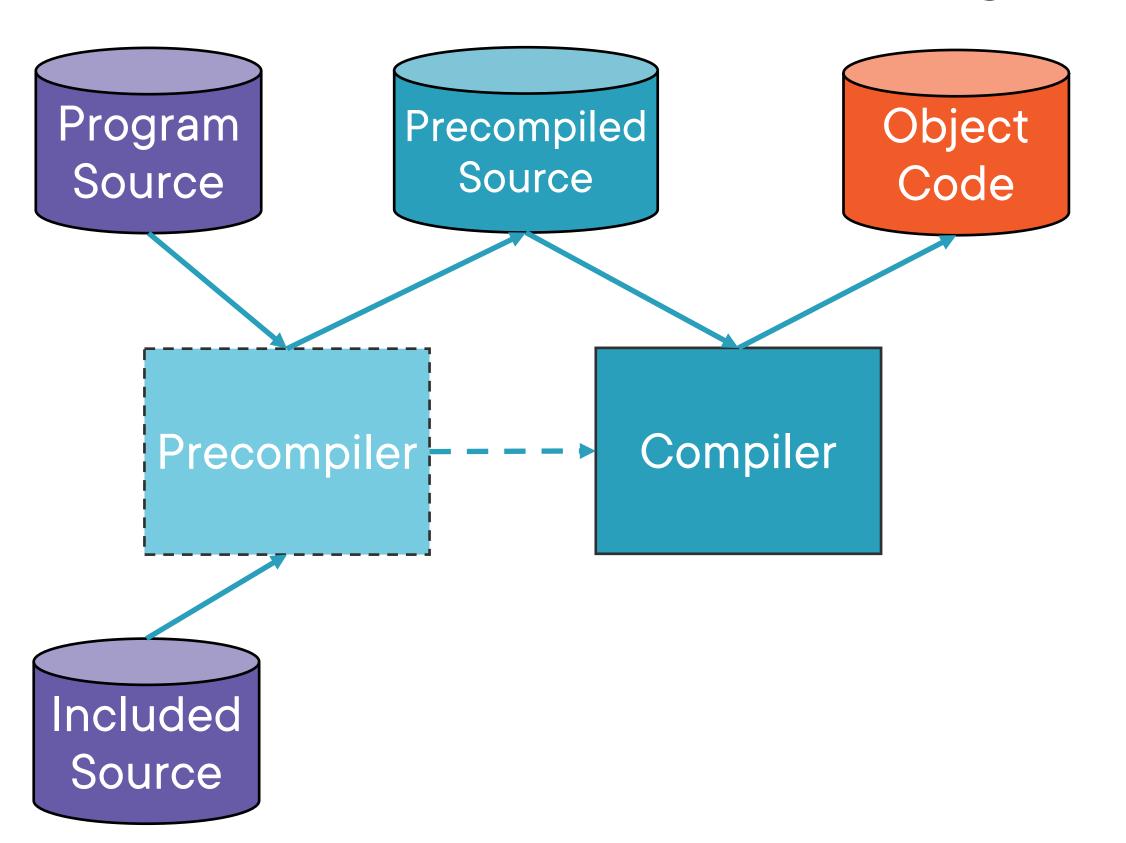


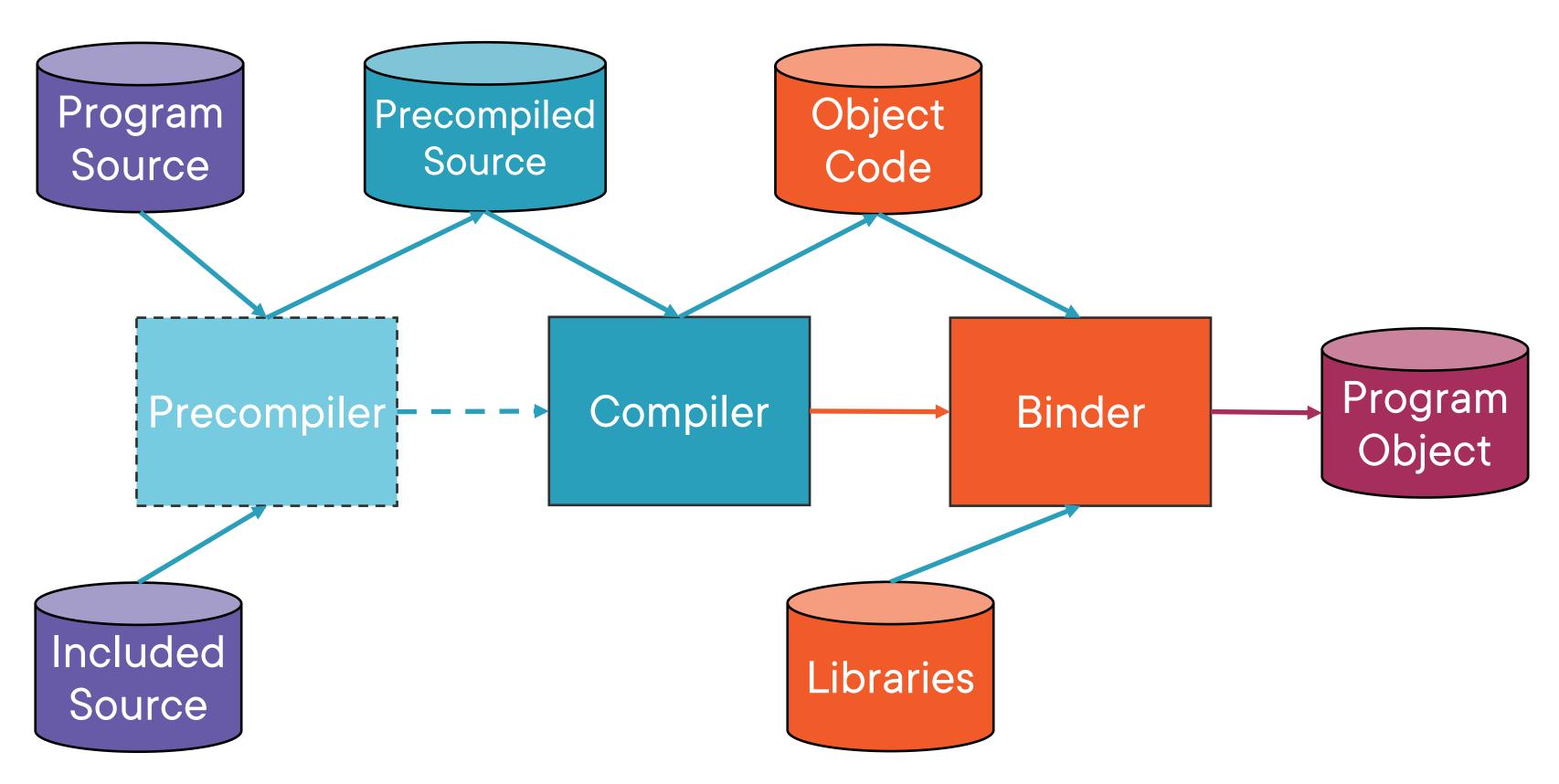
- How Programs Are Prepared for Execution
- Writing an In-Stream JCL Procedure
- Writing a Catalogued JCL Procedure
- Using Compile Procedures

How Programs Are Prepared for Execution









What Are Catalogued Procedures?

```
DCL NUM_VALUE DECIMAL;
DCL SQ_RESULT DECIMAL;
SQUARE_OF:
  PROC(VAL, SQ);
  SQ = VAL*VAL;
END SQUARE_OF;
invocation
NUM_VALUE = 9;
SQ_RESULT = SQUARE_OF(NUM_VALUE, SQ_RESULT);
  • • •
NUM_VALUE = 62;
SQ_RESULT = SQUARE_OF(NUM_VALUE, SQ_RESULT);
  • • •
```

► PL/I internal procedure
 Definition and use in the same source file

```
DATA DIVISION.
WORKING-STORAGE SECTION.
    05 WS-VALUE PIC S9(9) COMP-3.
    05 WS-RESULT PIC S9(11) COMP-3.
PROCEDURE DIVISION.
500-SQUARE-OF.
    MULTIPLY WS-VALUE BY WS-VALUE
        GIVING WS-RESULT
    END-MULTIPLY
invocation
    MOVE 9 TO WS-VALUE
    PERFORM 500-SQUARE-OF
    MOVE 62 TO WS-VALUE
    PERFORM 500-SQUARE-OF
```

◆ COBOL performed paragraph
Definition and use in the same source file

```
SQUARE: PROC(VAL, SQ) OPTIONS(MAIN);
  SQ = VAL*VAL;
  RETURN;
END SQUARE_OF;
invocation (different source file)
DCL NUM_VALUE DECIMAL;
DCL SQ_RESULT DECIMAL;
DECLARE SQUARE EXTERNAL
ENTRY(NUM_VALUE, SQ_RESULT);
   RETURNS (DECIMAL);
NUM_VALUE = 9;
SQ_RESULT = SQUARE(NUM_VALUE, SQ_RESULT);
NUM_VALUE = 62;
SQ_RESULT = SQUARE(NUM_VALUE, SQ_RESULT);
  . . .
```

► PL/I external procedure
 Definition and use in different source files

```
DATA DIVISION.
LINKAGE SECTION.
01 LS-ARGLIST.
   05 WS-VALUE PIC S9(9) COMP-3.
    05 WS-RESULT PIC S9(11) COMP-3.
PROCEDURE DIVISION USING LS-ARGLIST.
    MULTIPLY WS-VALUE BY WS-VALUE
       GIVING WS-RESULT
    END-MULTIPLY
   GOBACK
invocation (different source file)
DATA DIVISION.
WORKING-STORAGE SECTION.
01 WS-ARGLIST.
    05 WS-VALUE PIC S9(9) COMP-3.
    05 WS-RESULT PIC S9(11) COMP-3.
PROCEDURE DIVISION.
    CALL 'SQUAREOF'
       USING BY REFERENCE WS-ARGLIST
```

◆ COBOL called subprogram

Definition and use in different source files

Output

Description

Descript

JCL Procedures

In-Stream Procedure

The procedure is coded in the JCL stream that invokes the procedure

Catalogued Procedure

The procedure is coded in a library member separate from the JCL streams that invoke it

Compile-Link-Go Procedures

Variations in Compilation Procedures

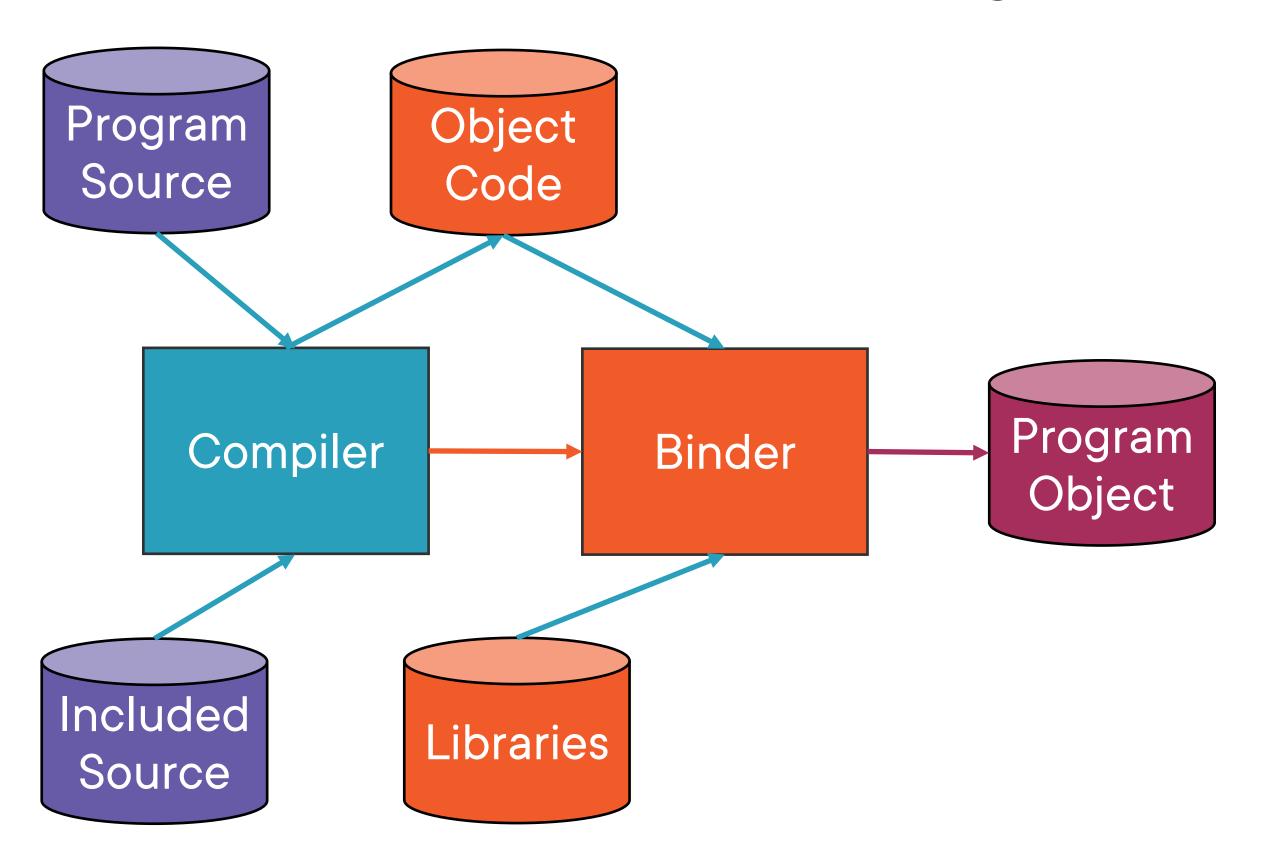
Variation	Examples
Language	COBOL, PL/I, C/C++, Assembler
Language version	(varies)
Execution environment	Batch, CICS, IMS/TM
Database access	DB2, IMS/DB
Steps	Precompile, compile, pre-link, link, bind, execute, add to lib

Batch Program Compile Procedures

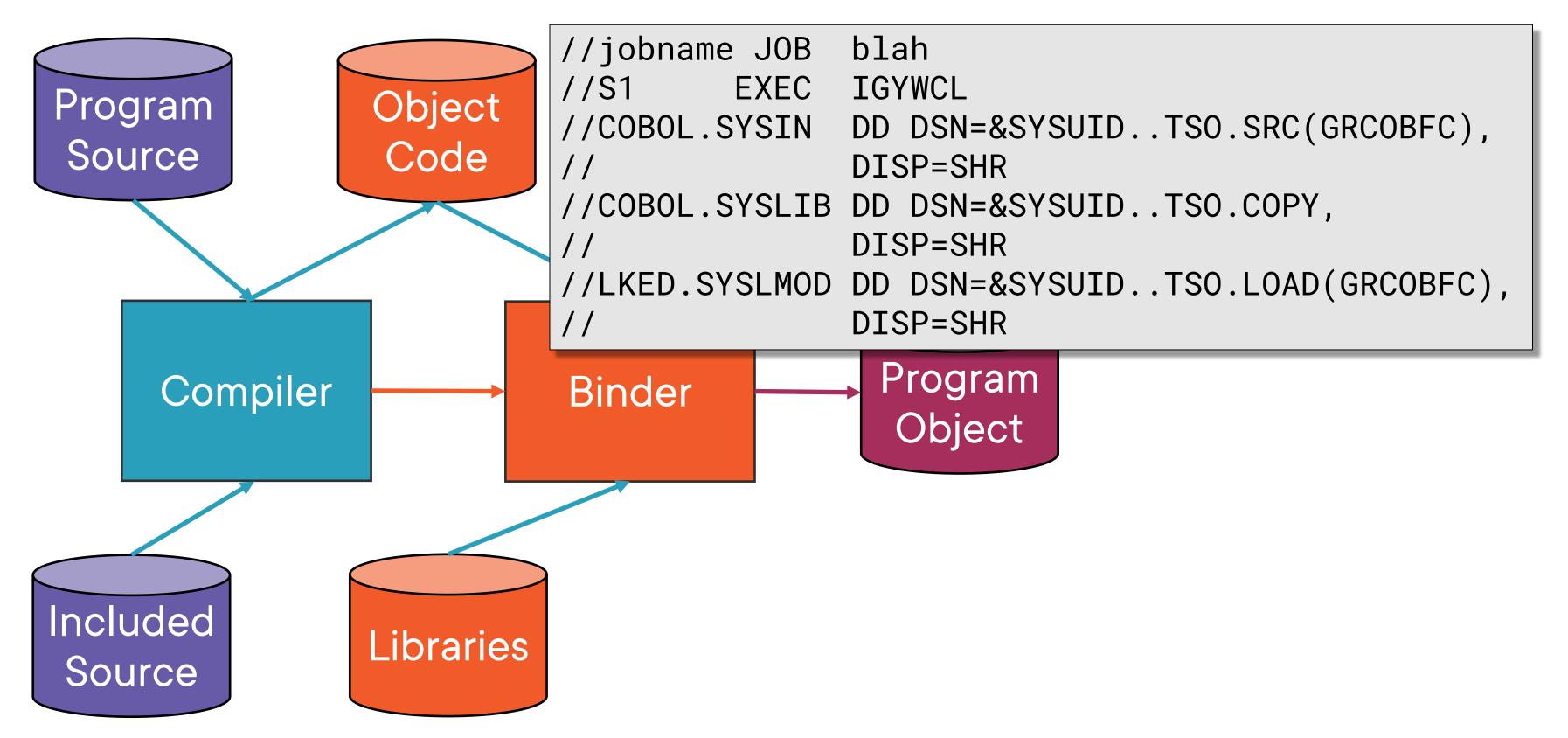
Base Names Language PLIX, PLICK, IEL1 PL/I IGYW, COB2 COBOL ASMA, HLASM Assembler CEDC C/C++ Suffix Steps compile link/bind P pre-link execute ("go") Examples COBOL compile and link/bind IGYWCL High-level assembler compile HLASMC

IEL1CG

PL/I compile and execute



Compile & Link a COBOL Program



Module Summary

Summary



- How Programs Are Prepared for Execution
- Writing an In-Stream JCL Procedure
- Writing a Catalogued JCL Procedure
- Using Compile Procedures

Up Next: Outlining Conditional Step Execution