14 – Query Compilation

1. Background

   A. The only way to improve performance of in-memory DBMS, is to reduce the number of instructions executed.

   B. Generating query-specific code can reduce number of instructions

2. Transpilation

   A. Write code that converts a query plan to efficient source code.
   (source code to source code)

   B. HIQUE
   Generate C/C++ program that implements query's execution

   C. Advantages

      i. Debugging is easy, because it is high level language

      ii. Generated query code can invoke any other function in the DBMS
      So, it can use all components of DBMS

   D. Disadvantages

      i. Compile C/C++ code may be long

      ii. Not support full pipelining

3. JIT Compilation

   A. Generate an IR of the query to organize query processing mainly on registers
   (source code to machine code)

   B. Compile time grows linearly by query size

   C. HyPer
   generate LLVM IR, execute IR using an interpreter immediately
   at background, compile the query,
   if compilation complete, then replace interpreter with the executable file

4. Real-World Implementations
    A. IBM system R
        i. Primitive form of query compilation and code generation
        compile each operator by specific template.
        ii. Abandoned when DB2 appears
        because of high cost, poor portability, etc
    B. ORACLE
        i. Store procedure into Pro*C, and compile it into C/C++ code
        ii. Oracle-specific operations directly in the SPARC chips.
    C. MS Hekaton
        i. Can compile procedures and SQL
        ii. Generate C code
    D. Actian Vector
        i. Pre-compiles primitives that perform basic operations on typed data
        actually, vectorwise version of primitives.
        ii. DBMS executes a query with primitives.
    E. MEMSQL (past)
        i. Perform code generation as HIQUE and invokes gcc
        ii. Converts all queries into parameterized form and caches query plan
    F. MEMSQL (present)
        i. Convert query plan into imperative plan
        ii. Convert imperative plan into opcodes
        iii. Compile the opcodes into LLVM and native code.
    G. PostgreSQL
        i. Support JIT compilation of predicates