Compilers and Computers:

Partners in Performance

Fran Allen

(IBM Fellow Emerita)

T. J. Watson Research Center

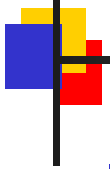
Yorktown Heights, NY 10598

allen@watson.ibm.com

Components of Performance

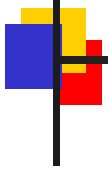
- Latency Reduction
  - Data and Instructions in the right place at the right time
- Fast Computations
- Concurrency

Talk will Cover



- Three high performance systems:
  - 1955-61: Stretch-Harvest System
  - 1962-68: Advanced Computing System
  - 1975-78: 801 System (RISC)
- And how the Compiler-Computer partnership for performance evolved

Some Context

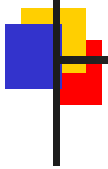


- 1954-57: Fortran I
- 1955-62: Stretch-Harvest System
- 1962-68: Advanced Computing System
- 1975-78: 801 System (RISC)

FORTRAN I

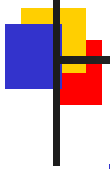
- Spectacular object code!!
- Some features:
  - Formal parsing techniques (beginnings)
  - Intermediate language form for optimization
  - Control flow graphs
  - Common sub-expression elimination
  - Generalized register allocation - for only 3 registers!

Some Context



- 1954-57: Fortran I
- 1955-62: Stretch-Harvest System
- 1962-68: Advanced Computing System
- 1975-78: 801 System (RISC)

Stretch (1955-61)



- Goal: 100 x faster than 704
- Main performance limitation (identified in 1955): Memory Access Time

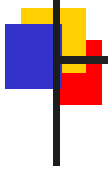
Stretch Memory

- 1Mb magnetic core memory
- Memory word lengths: 64 bits + 8 check bits
- Memory organization:
  - 8 core storage units of 16K words each
  - addresses interleaved across units
  - each unit independently connected via memory bus unit to cpu, I/O, disk
  - 2.1 us cycle time per unit
- Up to 6 storage accesses could be underway at the same time!!!!

Stretch Concurrency

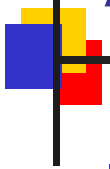
- Instruction Lookahead
  - up to 11 successive instructions executing in cpu at the same time
  - lookahead unit of virtual registers buffered instructions and data between memory and cpu
  - elaborate backout system to assure sequential consistency when interrupted
- Pipelining

Stretch Concurrency (cont'd)



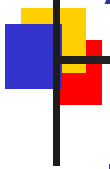
- Overlapped storage references
- I/O and disk operations
- Multiprogramming
  - to compensate for slow I/O
  - not shipped due to schedule

A Few Other Stretch Innovations



- Generalized interrupt system
- Memory protection
- Bytes [8 bits]
- Variable word length operands
- Multiple forms of floating point arithmetic
- Coupling two computers to a single memory

A Programmer's Dream

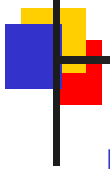


- 735 instructions (including modes)
- Bit addressable
- List walk took 2 instructions
- Multiple modes of arithmetic
- Registers and control functions part of addressable memory
- Word-level storage protection traps

A Compiler Writer's Nightmare!

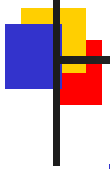
- Too many ways of doing the same thing
- FORTRAN could not use some features, e.g.multiple forms of arithmetic
- Organizing storage
- Scheduling instructions
- Etc……

Compiler as Part of the System



- A Stretch Objective: "The objective of economic efficiency was understood to imply minimizing the cost of answers, not just the cost of hardware. ... A consequent objective was to make programming easier -- not necessarily for trivial problems but for problems worthy of the computer, problems whose coding in machine language would usually be generated automatically by a compiler from statements in the user's language." Fred Brooks in "Planning a Computer System", 1962

HARVEST (1956-1962)



- Hosted by Stretch
- Designed for NSA for code breaking
- Streaming data computation model
- Seven instructions, e.g. Sort, SBBB
- Unbounded single operation times
- Only system with balanced I/O and computational speeds (per conversation with Jim Pomerene 11/00)

Harvest System

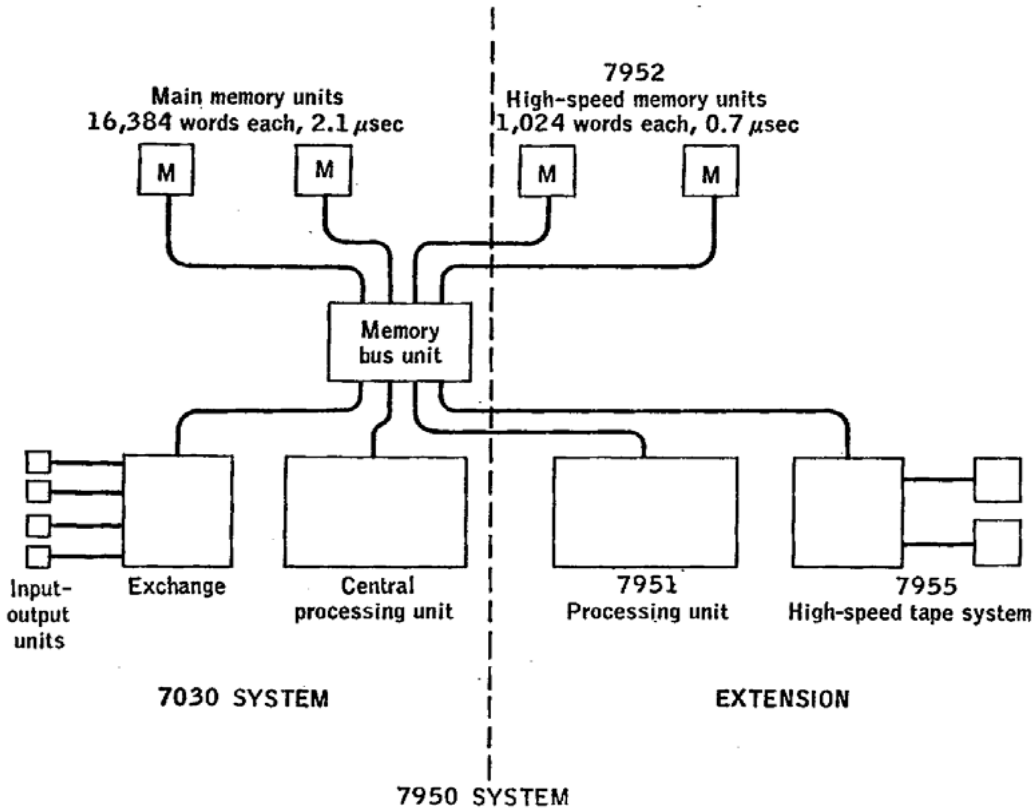


FIG. 17.1. Nonarithmetical extension of the 7030.

Harvest Streaming Unit

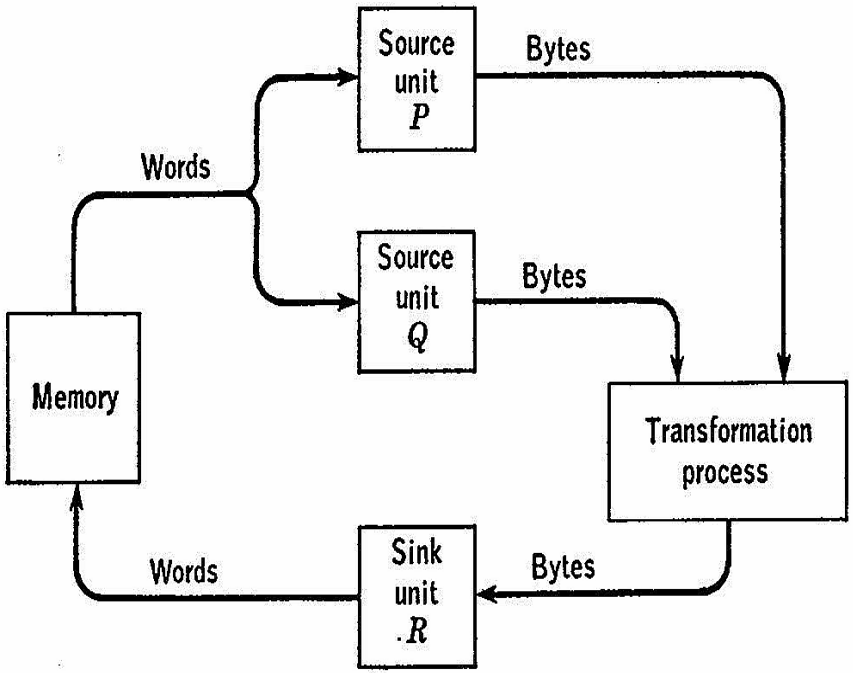
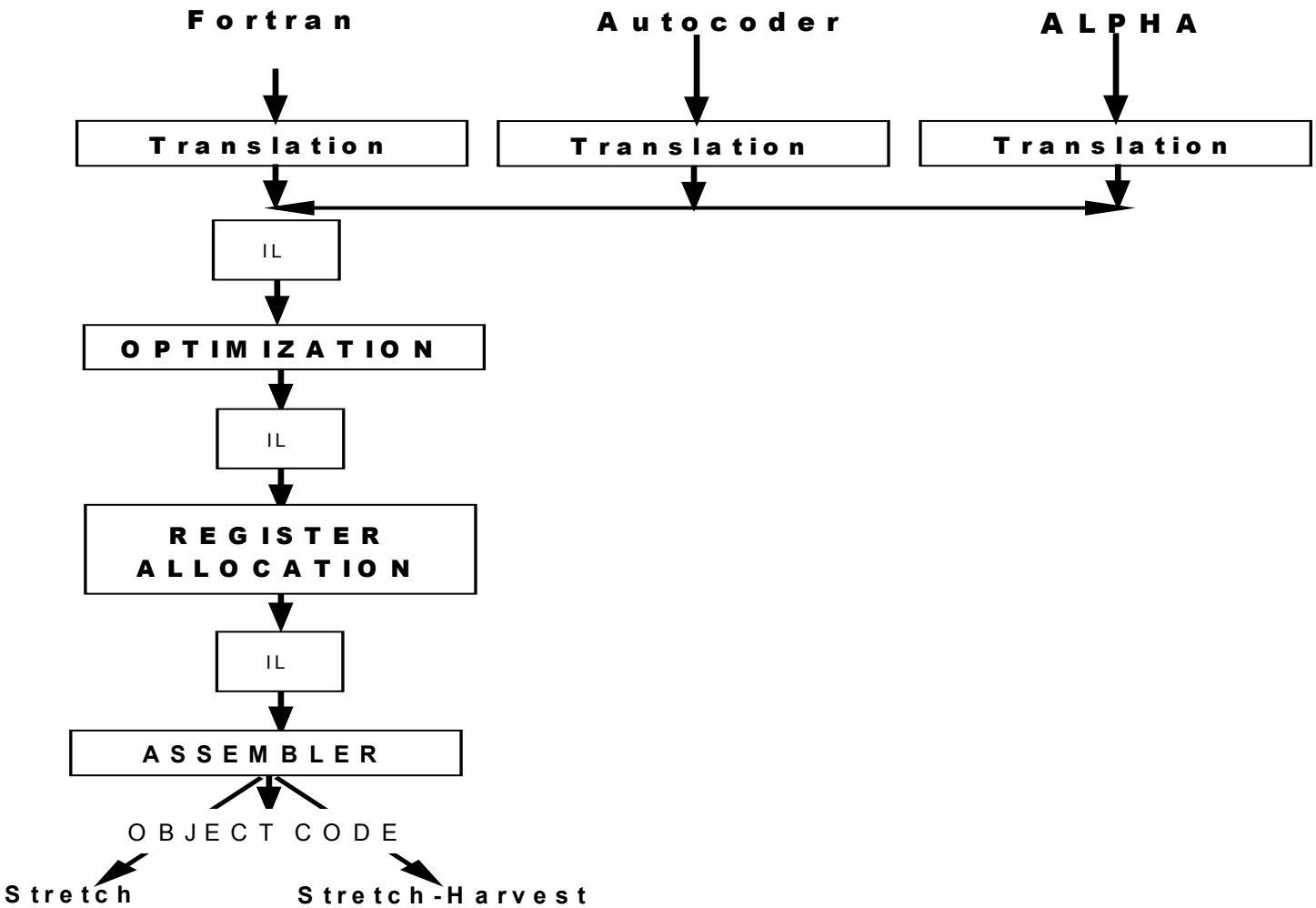


FIG. 17.2. Simplified data flow diagram.

Alpha Language for Harvest

- Language for cryptologists
  - Alphabet definition capability
  - Result descriptors by implication
- Matched the Harvest instructions but hid all details.

Stretch-Harvest Compiler



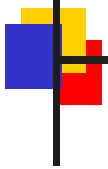
Stretch Retrospective

- Stretch machine missed 100 x goal
- Stretch compiler for Fortran replaced with simpler, faster compiler
- But "Stretch defined the limits of the possible for later generations of computer designers and users." (Dag Spicer - Curator Computer History Museum)

Harvest Retrospective

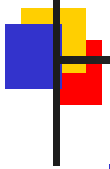
- Harvest heavily used for 14 years
  - Hardware was very successful
  - ALPHA and Compiler use is unknown
- Nov. 1968 NSA report: “Recently HARVEST scanned 7,075,315 messages of approximately 500 characters each -- examining every possible offset -- to see if they contained any of 7,000 different words or phrases. This ... required three hours and 50 minutes to complete -- an average of over 30,000 messages a minute.”

Next Step



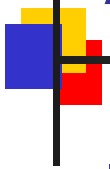
- 1954-57: Fortran I
- 1955-62: Stretch-Harvest System
- 1962-68: Advanced Computing System
- 1975-78: 801 System (RISC)

What We Had Learned



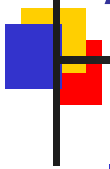
- Design compiler & computer together
- No instructions the compiler can’t use
- Keep the instruction set simple
- Keep the compiler simple
- A lot about building compilers and computers

ACS System (1962-1968)



- Goal: To build the fastest scientific computer feasible
- Compiler built early to drive hardware design

ACS Computer (1964-1968)



- Single instruction counter
- Superscalar: up to 7 ops per cycle;
- Pipelined
- Branch prediction
- > 50 insts in execution concurrently
- Programmable condition codes

ACS Compiler

- Early design used to establish:
  - Branch prediction strategies
  - Performance bottlenecks
  - Instruction scheduling techniques
- Code was sometimes faster than the best handcode

Some Compiler Results

- A theoretical basis for program analysis and optimization
- A Catalogue of Optimizations including:
  - Procedure integration
  - Loop transformations: unrolling, jamming, unswitching
  - Redundant subexpression elimination, code motion, constant folding, dead code elimination, strength reduction, linear function test replacement, carry optimization, anchor pointing
  - Instruction scheduling
  - Register allocation

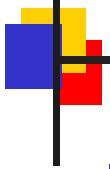
ACS

- ACS was cancelled in May 1968
- Too costly, too big, …..

Next Step

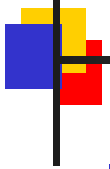
- 1954-57: Fortran I
- 1955-62: Stretch-Harvest System
- 1962-68: Advanced Computing System
- 1975-78: 801 System (RISC)

The 801 System (RISC)



- Goal: High Performance and Low Cost
- Simple, 1-cycle instructions
- Hardware, compiler, and new programming language (PL.8) developed simultaneously

Some 801 Project Results



- Graph coloring register allocator
- Influenced Berkeley RISC Project
- IBM's Power PC family of computers
- Optimizer became the core of IBM's XL family of retargetable compilers for multiple source languages.

More Steps

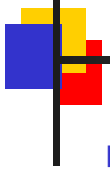
- 1954-57: Fortran I
- 1955-62: Stretch-Harvest System
- 1962-68: Advanced Computing System
- 1975-78: 801 System (RISC)
- Parallel Systems
- C
- Java
- Etc…..

Challenges

- The memory wall is getting worse
  - Caches
  - Locality
- Parallelism
- Programming languages
- Compilers for the New Millenium!

BACKUP CHARTS

Stretch Machine



- speed: ~ 500 KIPS (code dependent)
- base machine cycle: 300 ns (3.3MHz)
- transistors: 169,200
- disk: 2MW and 8Mbps
- tape drives: 12 *IBM 729 IV units
- card reader: 1000 cpm
- printer: 600 lpm
- card punch: 250 cpm
- cpu size: 900 sq. ft. (30 x 6 x 5)
- cpu power req: 21 KW
- total size: 2,500 sq. ft.
- weight: 40,000 lbs.

Tractor Tape for Harvest

- Tape library system with automatic retrival & storage
- 3 Cartridge handling units
- 160 cartridges/handler
- 90 million characters/tape
- 1,128,000 characters/sec transfer rate
- Automatic checking and error correction
- The 3 cartridge handlers could execute in parallel

Tractor Tape

The max read/write transfer rate was > 3.3MB/sec,

matching the rate of the streaming unit!