

Zoo Story

How the SHARP APL Development Group Got Its Name

Robert Bernecky

Snake Island Research Inc, Canada
bernecky@snakeisland.com

October 5, 2016

Abstract

Once upon a time, there was a high-tech Canadian company, called I.P. Sharp Associates. This was in the days before the phrase "high-tech" had been coined, and few people knew what an amazing and wonderful thing it was to be associated with so many intelligent, generous, and thoughtful people. Few of us thought of it as "work".

This talk presents some of my early encounters at IPSA.

The Picture: 1970

- ▶ I moved to Toronto from Buffalo

The Picture: 1970

- ▶ I moved to Toronto from Buffalo
- ▶ I went to work for Douglas Aircraft of Canada

The Picture: 1970

- ▶ I moved to Toronto from Buffalo
- ▶ I went to work for Douglas Aircraft of Canada
- ▶ Would I do high-tech aircraft software? No. That was in California

The Picture: 1970

- ▶ I moved to Toronto from Buffalo
- ▶ I went to work for Douglas Aircraft of Canada
- ▶ Would I do high-tech aircraft software? No. That was in California
- ▶ Instead, I ended up writing COBOL programs. . .

The Picture: 1970

- ▶ I moved to Toronto from Buffalo
- ▶ I went to work for Douglas Aircraft of Canada
- ▶ Would I do high-tech aircraft software? No. That was in California
- ▶ Instead, I ended up writing COBOL programs. . .
- ▶ Pre-Cambrian hardware

The Picture: 1970 Hardware

- ▶ Pre-Cambrian hardware: dinky IBM S/360 Model 30
 - ▶ 8KB memory
 - ▶ 1401 emulator (hardware)
 - ▶ card reader/punch (Read 1000/minute, punch 250/minute)
 - ▶ paper tape reader (1000 characters/second!!)
 - ▶ line printer (600 lines/minute = ten pages/minute)
 - ▶ no console
 - ▶ one tape drive & tape controller

The Picture: 1970 Hardware

- ▶ Pre-Cambrian hardware: dinky IBM S/360 Model 30
 - ▶ 8KB memory
 - ▶ 1401 emulator (hardware)
 - ▶ card reader/punch (Read 1000/minute, punch 250/minute)
 - ▶ paper tape reader (1000 characters/second!!)
 - ▶ line printer (600 lines/minute = ten pages/minute)
 - ▶ no console
 - ▶ one tape drive & tape controller
- ▶ Pre-Cambrian hardware: bigger IBM S/360 Model 30
 - ▶ 32KB memory
 - ▶ very slow card reader
 - ▶ no printer
 - ▶ no console
 - ▶ tape drives & another tape controller
 - ▶ disk drives (8x29MB)

The Picture: 1970

- ▶ No console: all OS interactions via memory patching

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough
- ▶ No printer on bigger machine: all output written to tape

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough
- ▶ No printer on bigger machine: all output written to tape
- ▶ One tape reel per job, printed on dinky machine

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough
- ▶ No printer on bigger machine: all output written to tape
- ▶ One tape reel per job, printed on dinky machine
- ▶ Pile after pile of tapes awaiting to be printed. . .

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough
- ▶ No printer on bigger machine: all output written to tape
- ▶ One tape reel per job, printed on dinky machine
- ▶ Pile after pile of tapes awaiting to be printed. . .
- ▶ Turnaround time: 2-3 days for **anything**

The Picture: 1970

- ▶ No console: all OS interactions via memory patching
- ▶ Two tape controllers, when one would be enough
- ▶ No printer on bigger machine: all output written to tape
- ▶ One tape reel per job, printed on dinky machine
- ▶ Pile after pile of tapes awaiting to be printed. . .
- ▶ Turnaround time: 2-3 days for **anything**
- ▶ **Mental health: absent**

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler
 - ▶ - no more need for dinky machine

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler
 - ▶ - no more need for dinky machine
- ▶ My boss thought it was a great idea

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler
 - ▶ - no more need for dinky machine
- ▶ My boss thought it was a great idea
- ▶ **Boss's boss didn't like it**

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler
 - ▶ - no more need for dinky machine
- ▶ My boss thought it was a great idea
- ▶ Boss's boss didn't like it
- ▶ **Boss's boss *REALLY* didn't like it**

Proposed Hardware/Software Improvements

- ▶ I recommended doubling bigger machine memory, to 64KB
- ▶ This would let us run DOS/POWER, a print-spooling system:
 - ▶ - no more tape printer output handling
 - ▶ - concurrent reading/printing while other jobs running
- ▶ I recommended using my 1401 simulator, not emulator:
 - ▶ - allowed 1401 jobs to run under print spooler
 - ▶ - no more need for dinky machine
- ▶ My boss thought it was a great idea
- ▶ Boss's boss didn't like it
- ▶ Boss's boss *REALLY* didn't like it
- ▶ Eventually, boss's boss's boss heard about it, & approved idea

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!
- ▶ So, did I get a big raise?

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!
- ▶ So, did I get a big raise?
- ▶ No. I got fired.

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!
- ▶ So, did I get a big raise?
- ▶ No. I got fired.
- ▶ Lesson: Look at The Big Picture

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!
- ▶ So, did I get a big raise?
- ▶ No. I got fired.
- ▶ Lesson: Look at The Big Picture
- ▶ **Boss's boss was getting kickbacks from IBM salesman**

The Picture: Results of Project

- ▶ Turnaround time dropped from days to minutes
- ▶ No more need for dinky machine
- ▶ No more need for several tape drives & controller
- ▶ No more need for one card reader
- ▶ Mainframe-related hardware costs halved – my annual salary each month!
- ▶ So, did I get a big raise?
- ▶ No. I got fired.
- ▶ Lesson: Look at The Big Picture
- ▶ Boss's boss was getting kickbacks from IBM salesman
- ▶ Off to the headhunter

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:
 - ▶ - compilers: yes

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:
 - ▶ - compilers: yes
 - ▶ - hours: no problem

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:
 - ▶ - compilers: yes
 - ▶ - hours: no problem
 - ▶ - suit: no problem

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:
 - ▶ - compilers: yes
 - ▶ - hours: no problem
 - ▶ - suit: no problem
 - ▶ - pay: 20% less than last job

Bernecky(rbe) Interview with Ian Sharp(ips): 1971

- ▶ Me to headhunter: "I want to work on compilers."
- ▶ Me to headhunter: "I want to work my own hours."
- ▶ Me to headhunter: "I don't want to have to wear a suit."
- ▶ Me to headhunter: "I don't care what it pays."
- ▶ Headhunter to me: "There is only one place in town like that."
- ▶ Interview with Ian Sharp:
 - ▶ - compilers: yes
 - ▶ - hours: no problem
 - ▶ - suit: no problem
 - ▶ - pay: 20% less than last job
 - ▶ - "However, you'll have to talk with... *drumroll*... ROGER."

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumpled, smoking a cigarette & muttering at a listing

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumped, smoking a cigarette & muttering at a listing
- ▶ I told him that Ian Sharp sent me, re compiler job

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumped, smoking a cigarette & muttering at a listing
- ▶ I told him that Ian Sharp sent me, re compiler job
- ▶ rdm kept muttering about some bug, staring at listing. . .

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumpled, smoking a cigarette & muttering at a listing
- ▶ I told him that Ian Sharp sent me, re compiler job
- ▶ rdm kept muttering about some bug, staring at listing. . .
- ▶ Eventually, I asked him about the bug he was chasing; we discussed it

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumped, smoking a cigarette & muttering at a listing
- ▶ I told him that Ian Sharp sent me, re compiler job
- ▶ rdm kept muttering about some bug, staring at listing. . .
- ▶ Eventually, I asked him about the bug he was chasing; we discussed it
- ▶ More eventually, I told him I had another interview scheduled

Bernecky (rbe) Interview With Roger Moore (rdm): 1971

- ▶ Fade in: Noisy room, with 2741 APL "opr" terminal logging signons
- ▶ S360/50 with 384KB memory, running SHARP APL. User load 25.
- ▶ The late Jacquei Sellgren, a delightful soul & first commercial APL system operator, pointed me at rdm
- ▶ rdm: rumped, smoking a cigarette & muttering at a listing
- ▶ I told him that Ian Sharp sent me, re compiler job
- ▶ rdm kept muttering about some bug, staring at listing. . .
- ▶ Eventually, I asked him about the bug he was chasing; we discussed it
- ▶ More eventually, I told him I had another interview scheduled
- ▶ rdm said, offhand, that he was willing to hire me.

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ **IPSCOBOL: I.P. Sharp COBOL compiler**

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ IPSCOBOL: I.P. Sharp COBOL compiler
- ▶ **Very fast compiles: 10 seconds vs. 30 minutes for IBM compiler**

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ IPSCOBOL: I.P. Sharp COBOL compiler
- ▶ Very fast compiles: 10 seconds vs. 30 minutes for IBM compiler
- ▶ Excellent diagnostics: "Attempt to write to closed file" vs. core dump

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ IPSCOBOL: I.P. Sharp COBOL compiler
- ▶ Very fast compiles: 10 seconds vs. 30 minutes for IBM compiler
- ▶ Excellent diagnostics: "Attempt to write to closed file" vs. core dump
- ▶ rbe to brown-thumb the data-type conversion algorithms

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ IPSCOBOL: I.P. Sharp COBOL compiler
- ▶ Very fast compiles: 10 seconds vs. 30 minutes for IBM compiler
- ▶ Excellent diagnostics: "Attempt to write to closed file" vs. core dump
- ▶ rbe to brown-thumb the data-type conversion algorithms
- ▶ rbe hand-codes several (hundreds are needed) unit tests

The IPSCOBOL Compiler Project: 1971

- ▶ I start work, but nobody gives me anything to work on...
- ▶ After a week or so, I ask rdm if I can do anything useful
- ▶ rdm suggests I try to break the IPSCOBOL compiler
- ▶ IPSCOBOL: I.P. Sharp COBOL compiler
- ▶ Very fast compiles: 10 seconds vs. 30 minutes for IBM compiler
- ▶ Excellent diagnostics: "Attempt to write to closed file" vs. core dump
- ▶ rbe to brown-thumb the data-type conversion algorithms
- ▶ rbe hand-codes several (hundreds are needed) unit tests
- ▶ rdm suggests I use APL to generate unit tests and check results

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
- ▶ `15`

rbe learns APL: 1971

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
- ▶ 15
- ▶ 1 2 3 4 5

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
- ▶ 15
- ▶ 1 2 3 4 5
- ▶ dlf sits back, looks at me, and smiles, smugly

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
- ▶ `15`
- ▶ `1 2 3 4 5`
- ▶ dlf sits back, looks at me, and smiles, smugly
- ▶ rbe, bemused: "I could write a program to do that..."

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
- ▶ 15
- ▶ 1 2 3 4 5
- ▶ dlf sits back, looks at me, and smiles, smugly
- ▶ rbe, bemused: "I could write a program to do that. . ."
- ▶ dlf: "I just did."

- ▶ rdm says I should get Doug Forkes (dlf) to teach me some APL
- ▶ dlf sits down at a terminal with me, enters:
 - ▶ `15`
 - ▶ `1 2 3 4 5`
- ▶ dlf sits back, looks at me, and smiles, smugly
- ▶ rbe, bemused: "I could write a program to do that. . ."
- ▶ dlf: "I just did."
- ▶ Later, I got the idea that APL was not just a one-trick pony, & learned APL

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y
- ▶ rdm opined that the indexof algorithm was not optimized.

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y
- ▶ rdm opined that the indexof algorithm was not optimized.
- ▶ rbe and rdm eyeballed APL interpreter source code...

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y
- ▶ rdm opined that the indexof algorithm was not optimized.
- ▶ rbe and rdm eyeballed APL interpreter source code. . .
- ▶ Indeed, X^1Y used quadratic-complexity algorithm

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y
- ▶ rdm opined that the indexof algorithm was not optimized.
- ▶ rbe and rdm eyeballed APL interpreter source code...
- ▶ Indeed, X^1Y used quadratic-complexity algorithm
- ▶ rdm: "Fix it."

rbe Becomes an Implementor: 1971

- ▶ rbe's APL function to check IPSCOBOL unit test output ran very slowly
- ▶ Hitting ATTN (break) always had caret pointing at X^1Y
- ▶ rdm opined that the indexof algorithm was not optimized.
- ▶ rbe and rdm eyeballed APL interpreter source code. . .
- ▶ Indeed, X^1Y used quadratic-complexity algorithm
- ▶ rdm: "Fix it."
- ▶ rbe: So I did, designing (mostly) linear time algorithms:

... Becomes an Implementor: 1971

- ▶ My first published paper eventually appeared in 1973



Figure: Cover of APL Congress 1973, Copenhagen

My First Published Paper!

- ▶ Computer-typeset, no less!

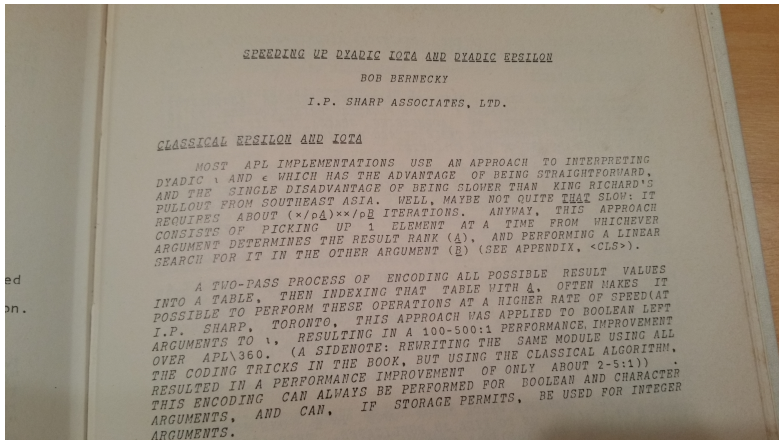


Figure: First page of article

The Amsterdam Connection

- ▶ The Intersystems office at 244 Herengracht



Figure: The Intersystems Office at 244 Herengracht, Amsterdam

The Amsterdam Connection

- ▶ Michael Harbinson: IPSANET packet switching, pre-internet

The Amsterdam Connection

- ▶ Michael Harbinson: IPSANET packet switching, pre-internet
- ▶ Intersystems BV: Schiphol Airport Air Traffic Control System

The Amsterdam Connection

- ▶ Michael Harbinson: IPSANET packet switching, pre-internet
- ▶ Intersystems BV: Schiphol Airport Air Traffic Control System
- ▶ rbe and others: Siemens APL implementation for BS2000 OS

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers
- ▶ rbe (and others) discovered CPU billing oddities

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers
- ▶ rbe (and others) discovered CPU billing oddities
- ▶ Various theories about causes, perhaps new Amdahl V6-II CPU

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers
- ▶ rbe (and others) discovered CPU billing oddities
- ▶ Various theories about causes, perhaps new Amdahl V6-II CPU
- ▶ rbe designed high-precision timers for SHARP APL:
tricentiseconds replaced by nanoseconds

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers
- ▶ rbe (and others) discovered CPU billing oddities
- ▶ Various theories about causes, perhaps new Amdahl V6-II CPU
- ▶ rbe designed high-precision timers for SHARP APL:
tricentiseconds replaced by nanoseconds
- ▶ rbe redesigned *reverse* & *rotate* for performance & nested array support

Finding Hardware Problems, circa 1978

- ▶ Hardware was getting very fast; APL needed precise timers
- ▶ rbe (and others) discovered CPU billing oddities
- ▶ Various theories about causes, perhaps new Amdahl V6-II CPU
- ▶ rbe designed high-precision timers for SHARP APL:
tricentiseconds replaced by nanoseconds
- ▶ rbe redesigned *reverse* & *rotate* for performance & nested array support
- ▶ rbe, seeking ego boost, compared old vs. new versions with new timer code

Finding Hardware Problems, circa 1978, user load 500

- ▶ 50 milli-CPU units = 2.85msec

Finding Hardware Problems, circa 1978, user load 500

- ▶ 50 milli-CPU units = 2.85msec
- ▶ Hey, what is that jitter and those spikes?

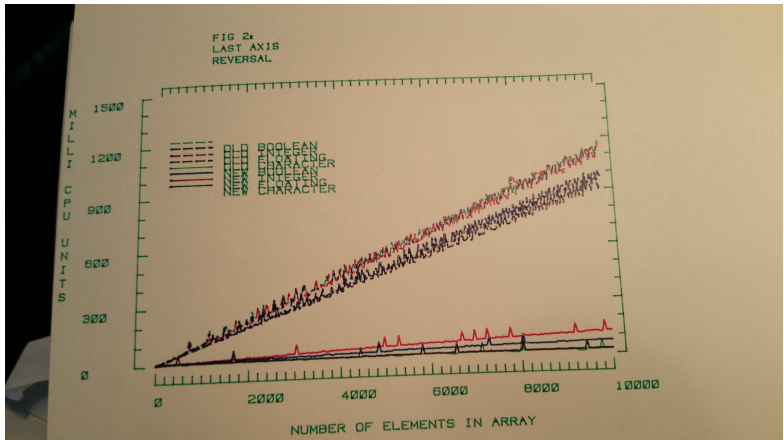


Figure: Ooh... faster, better, pretty pretty!

Finding Hardware Problems, circa 1978

► Let's zoom in

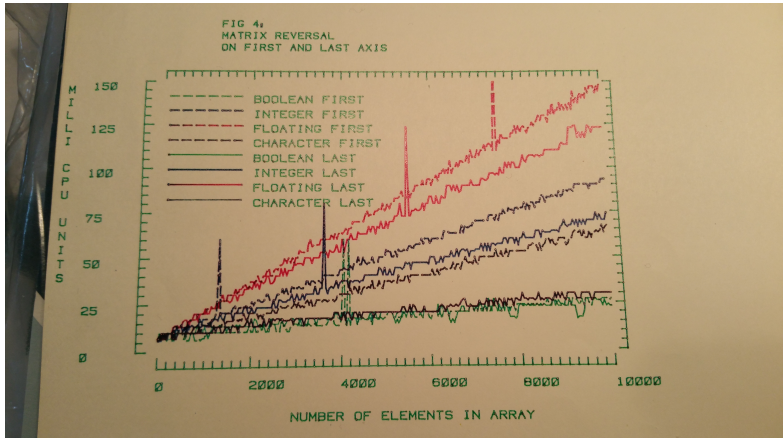


Figure: Wow! One spike = 10000-element reverse!

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ **Disabling SSK made spikes disappear!**

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ Disabling SSK made spikes disappear!
- ▶ New Amdahl V6-II had larger cache, with *much* slower SSK

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ Disabling SSK made spikes disappear!
- ▶ New Amdahl V6-II had larger cache, with *much* slower SSK
- ▶ rbe met with Gene Amdahl to discuss the problem & possible solutions

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ Disabling SSK made spikes disappear!
- ▶ New Amdahl V6-II had larger cache, with *much* slower SSK
- ▶ rbe met with Gene Amdahl to discuss the problem & possible solutions
- ▶ It became evident that the V6-II architecture was not going to get fixed.

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ Disabling SSK made spikes disappear!
- ▶ New Amdahl V6-II had larger cache, with *much* slower SSK
- ▶ rbe met with Gene Amdahl to discuss the problem & possible solutions
- ▶ It became evident that the V6-II architecture was not going to get fixed.
- ▶ So, we gave each workspace slot its own key, at APL startup

Finding Hardware Problems, circa 1978

- ▶ SHARP APL: many user workspaces in memory at once
- ▶ SSK isolated interpreter bugs to a single workspace
- ▶ I traced spikes to the Set Storage Key (SSK) instruction
- ▶ Disabling SSK made spikes disappear!
- ▶ New Amdahl V6-II had larger cache, with *much* slower SSK
- ▶ rbe met with Gene Amdahl to discuss the problem & possible solutions
- ▶ It became evident that the V6-II architecture was not going to get fixed.
- ▶ So, we gave each workspace slot its own key, at APL startup
- ▶ Slightly less system integrity, but problem solved

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed
- ▶ IPSA writes STAR APL for Control Data STAR-100 supercomputer

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed
- ▶ IPSA writes STAR APL for Control Data STAR-100 supercomputer
- ▶ My STAR inner product algorithm derived from this work

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed
- ▶ IPSA writes STAR APL for Control Data STAR-100 supercomputer
- ▶ My STAR inner product algorithm derived from this work
- ▶ rbe and others: Siemens APL implementation, Amsterdam

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed
- ▶ IPSA writes STAR APL for Control Data STAR-100 supercomputer
- ▶ My STAR inner product algorithm derived from this work
- ▶ rbe and others: Siemens APL implementation, Amsterdam
- ▶ JIT: Larry M. Breed designs & implements APL scalar function compiler

IPSA Happenings, early 1970s

- ▶ 1971: 666 box: email before internet, written by Larry Breed
- ▶ IPSA writes STAR APL for Control Data STAR-100 supercomputer
- ▶ My STAR inner product algorithm derived from this work
- ▶ rbe and others: Siemens APL implementation, Amsterdam
- ▶ JIT: Larry M. Breed designs & implements APL scalar function compiler
- ▶ JIT: rbe, Greg Bezoff (gbe), Doug Forkes (dlf) implement various verbs & conjunctions as JIT code

- ▶ IPISA wrote APL systems for Hitachi & Fujitsu

Time Marches on

- ▶ IPSA wrote APL systems for Hitachi & Fujitsu
- ▶ IPSA outgrows 74 Vic data centre, moves to 145 King St. W.; user load 1000

Time Marches on

- ▶ IPSA wrote APL systems for Hitachi & Fujitsu
- ▶ IPSA outgrows 74 Vic data centre, moves to 145 King St. W.; user load 1000
- ▶ IPSA development, marketing, & sales now shared same floor. No security

Time Marches on

- ▶ IPSA wrote APL systems for Hitachi & Fujitsu
- ▶ IPSA outgrows 74 Vic data centre, moves to 145 King St. W.; user load 1000
- ▶ IPSA development, marketing, & sales now shared same floor. No security
- ▶ One day: lost client wandered into data centre, was shocked and appalled by scruffy developers

Time Marches on

- ▶ IPSA wrote APL systems for Hitachi & Fujitsu
- ▶ IPSA outgrows 74 Vic data centre, moves to 145 King St. W.; user load 1000
- ▶ IPSA development, marketing, & sales now shared same floor. No security
- ▶ One day: lost client wandered into data centre, was shocked and appalled by scruffy developers
- ▶ Next day: New signs appeared at the elevator doors:

Time Marches on

- ▶ IPSA wrote APL systems for Hitachi & Fujitsu
- ▶ IPSA outgrows 74 Vic data centre, moves to 145 King St. W.; user load 1000
- ▶ IPSA development, marketing, & sales now shared same floor. No security
- ▶ One day: lost client wandered into data centre, was shocked and appalled by scruffy developers
- ▶ Next day: New signs appeared at the elevator doors:

▶ < - - - - *SUITS* *JEANS* - - - - >

Time Marches on. Again

- ▶ Ian Sharp wanted to give people “an interesting place to work”.

Time Marches on. Again

- ▶ Ian Sharp wanted to give people “an interesting place to work”.
- ▶ He did that! Thank you, Ian, for your great vision and spirit



Time Marches on. Further

- ▶ Lost client saw feral developers; exclaimed “This place is a zoo!”

Time Marches on. Further

- ▶ Lost client saw feral developers; exclaimed “This place is a zoo!”
- ▶ The name stuck, even unto the email address

Time Marches on. Further

- ▶ Lost client saw feral developers; exclaimed “This place is a zoo!”
- ▶ The name stuck, even unto the email address
- ▶ So, with apologies to Albee, I can truly say: “I’ve been to the zoo!”