

The first 35+ years of Hursley
software 1958-1993
with some hardware asides,
a personal account

Andy Krasun
1968-2003

Caveats

- This is a personal view which has been looked at, but not verified as a paper, by friends and colleagues from Hursley
- Any errors are mine, and I would welcome any corrections and additions
- In an hour, I can only skim the subject
- IBM is full of acronyms mostly TLAs or FLAs and even nested TLAs and FLAs!

But the trouble is that the expanded version of these acronyms are probably as incomprehensible too! And sometimes tautologous

Hursley House



BCS October 2006

Hursley' s history - 2006



BCS October 2006

Background

- IBM started to invest in UK development in mid 1950s
 - Excellent pool of talent which was less expensive than the USA
 - Predominantly engineers
 - IBM development arrived at Hursley on December 1, 1958
 - Worked mainly on special European projects
 - Scientific and Modulator Processor (SCAMP)
 - Overlapped I/O
- Multiple machine architectures in IBM by 1962
 - Commercial programmed in machine language and COBOL
 - 1401, 1403 supported by the COMMON user group
 - 704, 7040 supported by the GUIDE user group
 - Scientific programmed in machine language and Fortran
 - 709, 7090, 7094 supported by the SHARE user group
 - 1620
 - Process control
 - 1710

Acronym Soup

- Background
- PL/I-F compiler
- PL/I Checker and Optimiser (Optimizer)
- STAM 1,2,3,4, VTAM
- HMP
- CICS/VS and CICS DOS/VS
- REXX
- NDB
- IBM3730 and DISOSS
- CICS/MVS
- GDDM
- CICS/ESA
- OSI and CICS
- OS/2 PM
- CICS OS/2 and the rest of the CICS family

Soup to Nuts

- Background
- PL/I-F compiler
- PL/I Checker and Optimiser (Optimizer)
- STAM 1,2,3,4, Virtual Telecommunications Access Method
- Hursley Macro Project
- Customer Information Control System/VS and CICS DOS/VS
- Restructured Extended EXecutor
- Norman's Data Base
- IBM3730 and DIStributed Office Support System
- Customer Information Control System/Multiple Virtual System
- Graphical Data Display Manager
- Customer Information Control System/Extended Systems Architecture
- Open System Interconnect and Customer Information Control System
- OS/2 Presentation Manager
- CICS OS/2 and the rest of the CICS family

Competition

- BUNCH based in the USA and selling worldwide
 - Burroughs, UNIVAC, NCR, CDC, Honeywell
 - Same languages as IBM with the addition of ALGOL
- UK based
 - Ferranti, ICT, English Electric, LEO, STC
 - Same as above plus various Auto-coders

IBM hardware

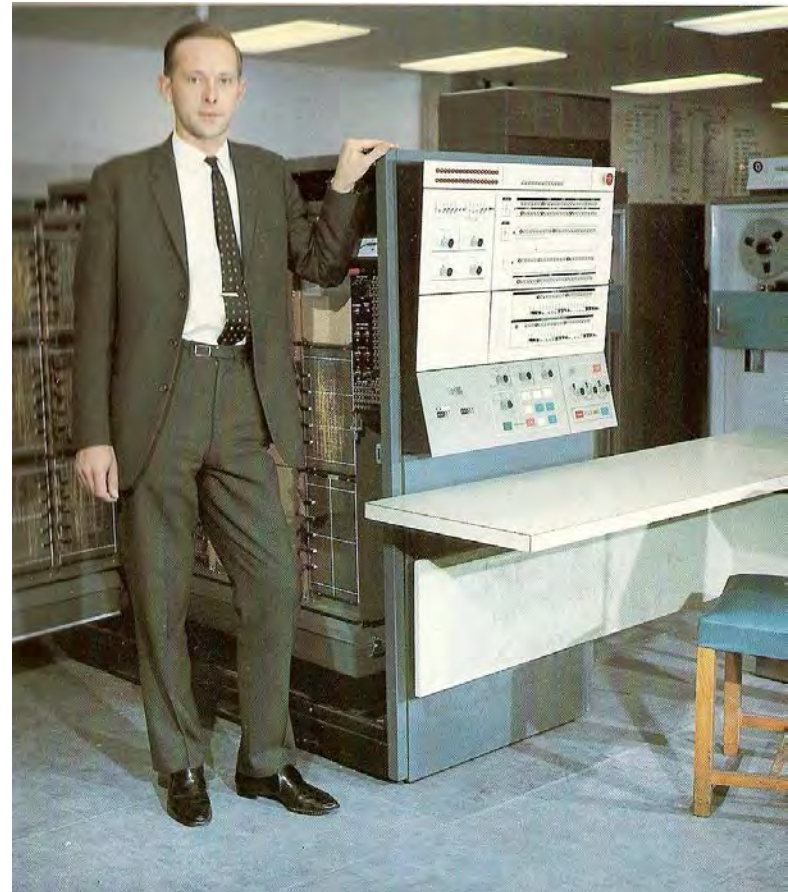
- Proliferation of different architectures in IBM hardware and the subsequent software started ringing alarm bells
- The “Solution”
 - System 360 announced in 1964
 - A range of computers with the same instruction set (well almost) though not the same machine technologies
 - Models 20, 30, 40, 50, 65, 67 and 91
 - 4K to 1M of memory
 - Various buffered peripherals
 - Communication controllers, removable disks and VDUs!
 - A set of operating systems to fit the box size
 - TOS, BOS
 - PCP, MFT and MVT to come later

360 architecture

- 32 bit aligned word
- 16 general purpose registers
- 4 float registers
- 24 bit addressing (31 bit on the Model 67)
- Hardware protection between regions ((address spaces), similar to processes)
- Buffered I/O
 - Channels, really separate I/O processors
 - Access through a DCB (Data Control Block)
incredibly overloaded for all devices (files, lines, terminals, etc.)
- Software bundled with the hardware

Hursley's early involvement

- 360 simulator, SUPPAK
 - Ran on a 7094
- Hardware
 - System 360 Model 40
 - First microcode machine using Transformer Read only Storage (TROS)
 - System 360 Model 44 (Scientific instruction set)
 - First machine I ever saw a floppy disk on
- Software
 - APEX, Random generator for the Production of EXecutable PL/I programs
 - PL/I F compiler



John Fairclough

PL/I, the ‘Betamax’ of programming languages

- Language designed by the 3x3 committee
 - 3 from GUIDE, 3 from SHARE and 3 from IBM
 - Intended to take over from COBOL and FORTRAN and also be a systems programming language
 - No reserved keywords
 - Mix of
 - IF-THEN-ELSE, Character handling, “subroutines”, data structures from COBOL
 - $(+ - *)$ syntax, DO loops and mathematical built-in functions from FORTRAN
 - ; syntax, block structures, subroutines and scope for variables from ALGOL
 - and later
 - Integrated macro pre-processor
 - Implicit declarations
 - Pointers and dynamic storage allocation from machine language
 - Record I/O
 - Telecommunications access

PL/I F compiler

- Designed to run in a 64K machine (1965-6)
- Software delivered in versions
- Upward source compatibility was a key selling point
- Written in pure assembler language
- Shipped as part of the operating system
- PL/I F Version 4 shipped in 1968

PL/I follow-on

- By PL/I F Version 3, it was decided that new compilers were needed because of spaghetti (non-structured) code and the now discernable differing objectives of:
 - Optimised run time
 - Increased programmer productivity (if you were lucky you might get three runs a day)
- So the concept of a pair of compilers the PL/I Optimiser and PL/I Checkout with unified run-time control blocks and libraries was born
 - In fact there were to be two pairs:

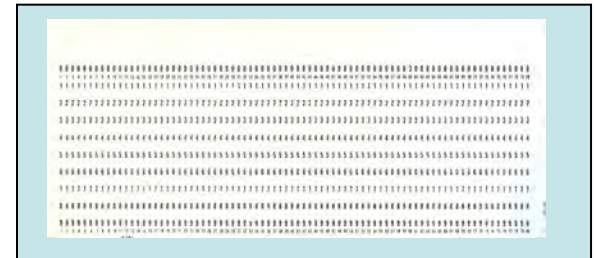
	42K - DOS	100K - OS
Checkout	Böblingen	Hursley
Optimiser	Hursley	Boulder, CO

PL/I Optimiser

- Written in optimised assembler macros to run in 44K using typewriter terminals
- Prototype was the F compiler
- Team built from F compiler personnel
- Objective was to be 50% faster than the F compiler
- Run-time did this through by-passing operating system services
 - Getmain/Freemain(Getmem/Freemem) especially for the stack
 - Buffered I/O only when buffer was full
- Also much better code generation phases
 - Register allocation
 - More inline code
- Shipped as a Program Product

PL/I Checkout Version 1

- Written in BSL(a proprietary PL/I subset optimised for systems programming, later renamed PLS, PLAS and then PLX)
- First objective was to “compile” 4 times faster than the F compiler so find syntactic and semantic errors more cheaply
- Second objective was to find errors and report them in source language terms with the correct statement number
- Use the common PL/I run-time library
- Provide the ability to link with PL/I Optimiser compiled code
- Prototype built: interpretive
- Team built from some F compiler personnel, rest were new graduates
 - Around a hundred maths graduates were recruited in 1968 (about 10% of the UK output) of which 20 worked on the Checker
- Shipped as a program product



PL/I Checkout Version 1

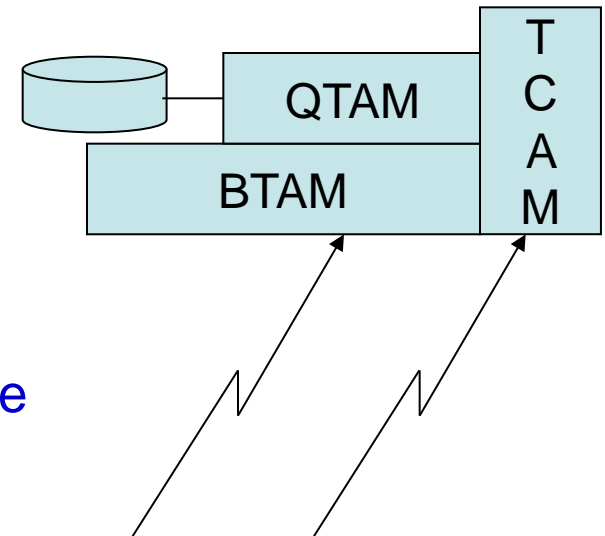
- First objective
 - Did this by having only three passes
 - Source errors shown with source “ Unknown keyword ‘RAED ? FILE(..’
 - Large amount of fix-up
 - Respelling ‘RAED’ respelled ‘READ’
 - Ability to format source
 - Private loading mechanism to avoid operating system
 - Output was a half-word text, “H-text “
 - | Operator | sub-code | optional length | dictionary reference(s) |
 - c.f. ITEXT, PCODE
- Second objective
 - Un-initialised variables
 - Bad pointers, 4-byte standard versus 16 byte which carried descriptor data

PL/I Checkout Version 2

- Used Time sharing Option (TSO) to provide terminal I/O
- Major new technology
 - Allowed additional statements, including declarations, to be added and executed during run-time with incremental translation
 - e.g.
120 PAYMENT=0; /* Initialise */
 - Also allowed source to be saved including the updates and formatted as well

STAM 1,2,3,4, VTAM

- Single terminal access method
- Planned to supersede Basic Telecommunications Access Method and Queued TAM
 - Session based
- Built using CLEAR/CASTER
 - First IBM development system to only use a video screen
 - Written in PL/S
- Later to become DOS Virtual TAM
- VTAM was the primary product to implement Systems Network Architecture (SNA)



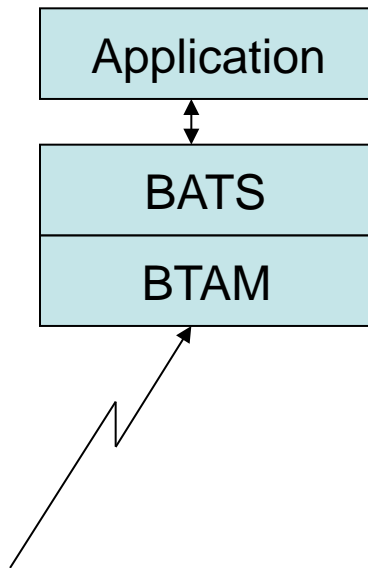
HMP

- Hursley macro processor
- Attempt to build an extensible language
 - Some similarities with Simula/C++
- Moved to New York Programming Centre in 1972
- Project wound up in 1975
- Leads later into EXEC CICS

FS

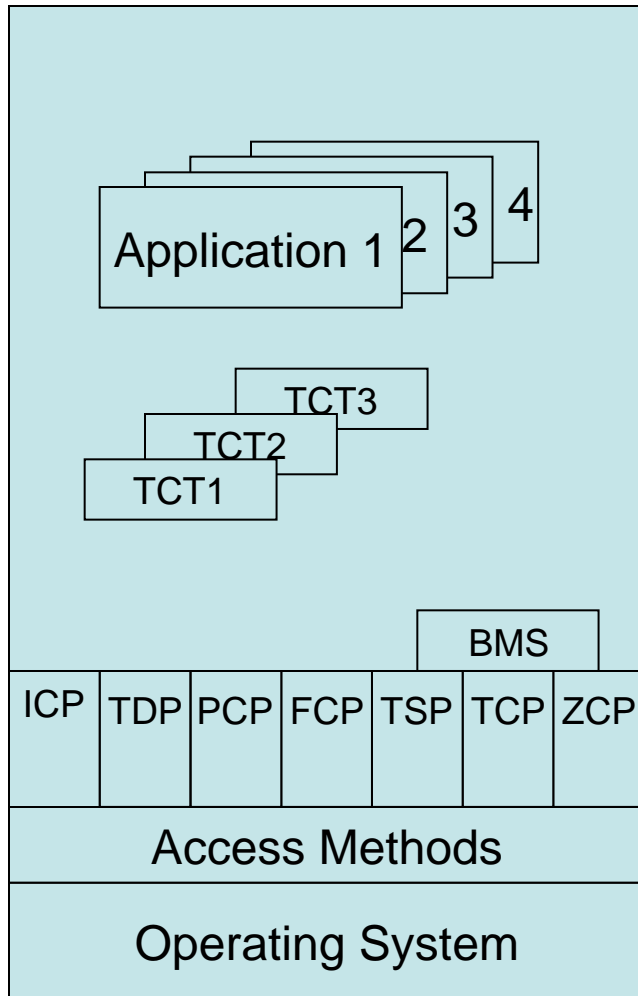
- Even after the 360 consolidation there were other machines, mainly 1130s and System 3s from GSD
- Concept for the series was a single level store machine with a built in database to run on an interpretive intermediate text very similar to the Checkout's H-text
 - This then implied that all languages should compile to this text and have a common runtime environment and subroutine library
 - So plans were made to consolidate all compilers and run times in Palo Alto, CA
 - We were not happy!

So CICS came to Hursley



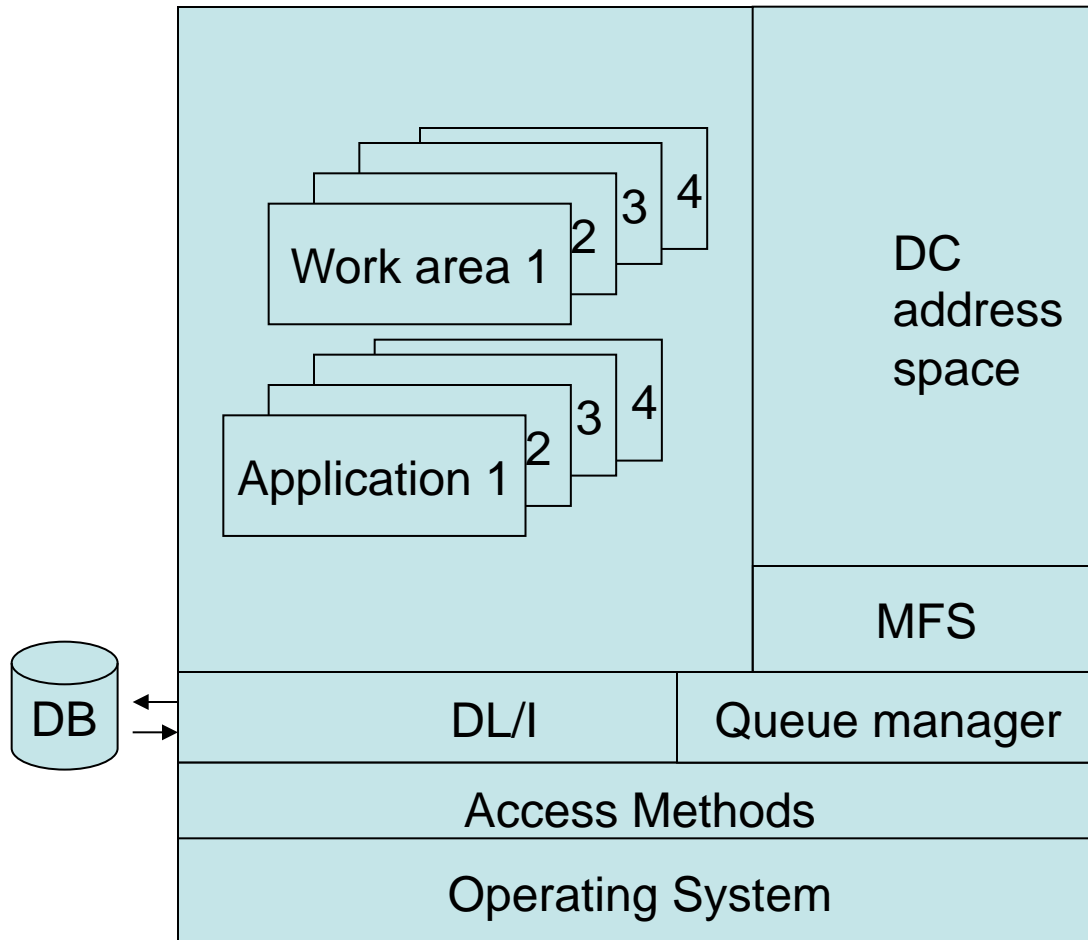
- Hursley had been offered a choice three products to replace PL/I
 - Data Dictionary
 - CICS
 - Low end time sharing system (VSPC)
- Malcolm Beaver, an IBM UK manager, was moved into Hursley to take over the Programming Centre
 - He had previously built British Administrative Terminal System in London which was used by most of the UK banks
 - high level interface to BTAM
 - SS and BSC support
 - basic CICS competitor
 - He chose CICS
 - Program product
 - Major competitor was SHADOW

CICS basic architecture



- ‘Invented’ by Ben Riggins in Chicago in late 1960’s
- Single region/address space
- Internal scheduler to avoid operating system overhead
- Resources
 - Defined by table entry, assembled for execution
 - rather nasty assembler macro programming interface for all languages (c.f. method)
 - Device independence through BMS
- Co-operative application program design

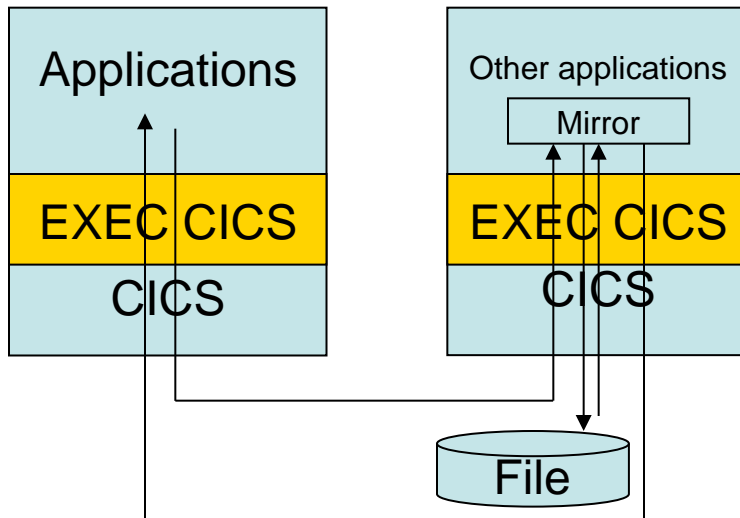
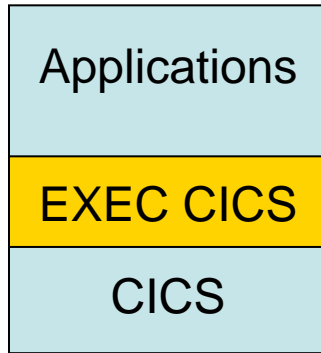
IMS basic architecture



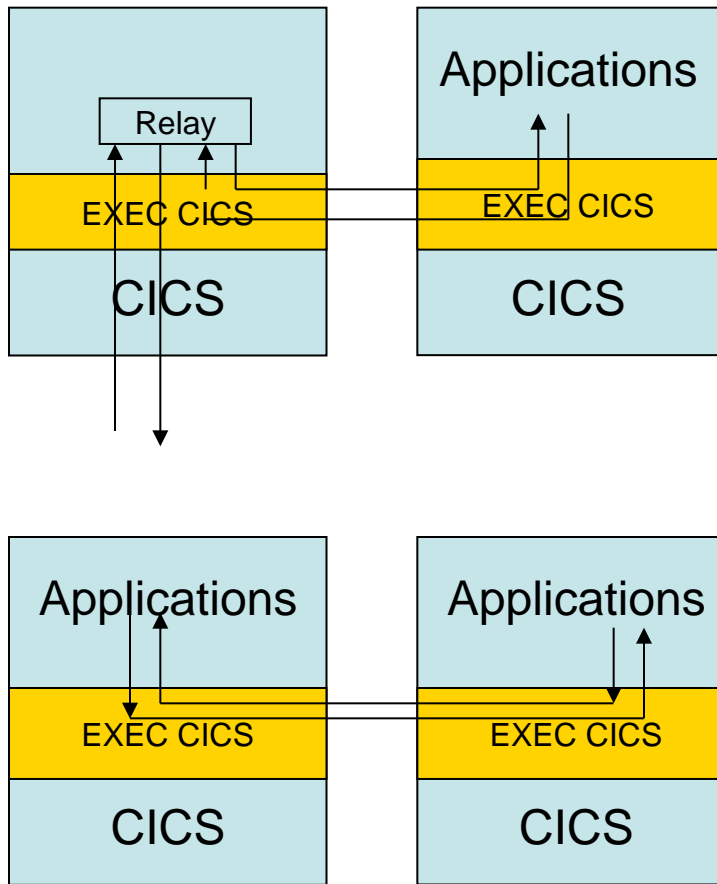
- Hierarchical data base ‘invented’ at Rockwell
 - Transactional
 - One application region
 - Simple CALL interface
 - No storage protection between tasks
 - Program product
- One address space for terminals
 - Feature
 - Device independence through MFS

Early CICS'

- 1.2
 - SNA device support(LU0)
- 1.3
 - EXEC CICS
 - Dynamic transaction backout
 - File control rewrite
 - Program isolation
- 1.4
 - Transparent access to remote data (Function shipping) through communications access method
 - Files
 - Queues
 - Base debugger (EDF)

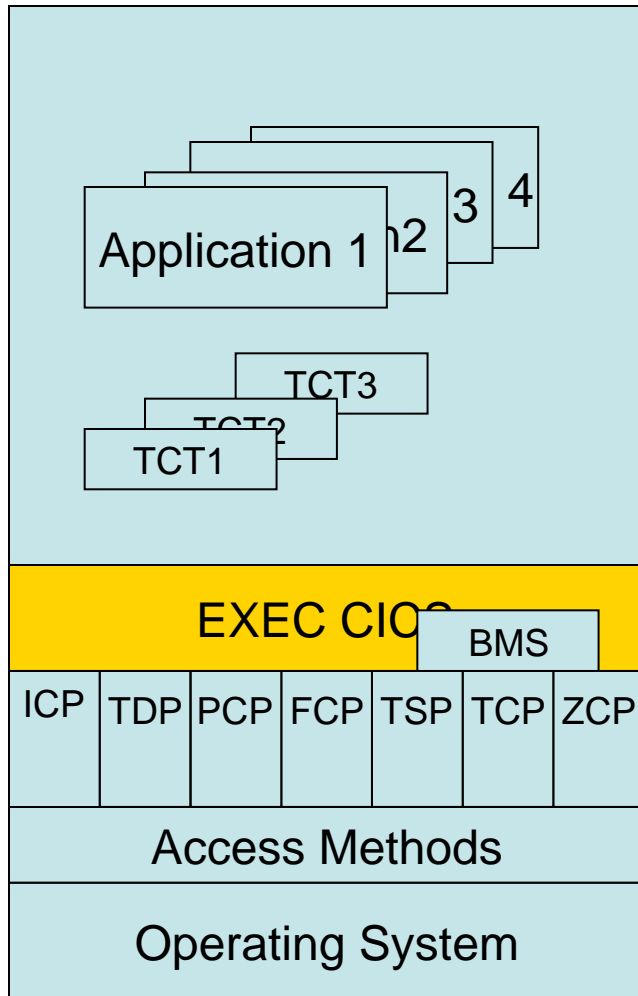


Early CICS' cont.



- 1.5
 - Transaction routing
 - Major increase in resilience
- 1.6
 - APPC
 - Application to application(LU6.2)
 - One phase commit
- 1.6.1
 - Start of 31 bit addressing

Early CICS' cont.



- 1.7

- APPC

- 2 phase commit

- File control rewrite

- Terminal auto-install

- Major storage relief

- Large cost of table generation removed

And then the challenges came!

REXX

- Restructured eXecutor
 - Designed firstly by Mike Cowlshaw
 - System independent
 - After much interaction, all by e-mail, the follow-on Restructured eXtended eXecutor was created
 - Documentation before implementation
 - PL/I based syntax
 - Can be used in a variety of ways
 - Personal programming, tailoring commands, macro language and for protoyping
 - Firt integrated into VM/CMS

Norman's DB

- Chris Date worked on DB extensions for PL/I
- Peter Titman and Norman Winterbotham had defined an entity-set/binary relational DB architecture
 - Major conflicts with Ted Codd
- Norman then built the Personal Binary Relational DB with an integrated User Interface which would interactively construct databases with reports
 - Still in use today

3730

- A Hursley programmed 3790 to perform shared office function for up to 24 users
 - Size of a large kitchen table
 - Not enough memory or engine speed (disk was a Hursley 62M Gulliver)
 - Terminals were reworked 3277s with new internals to do word-spill etc.
 - Pattern letters
 - Connected to a mainframe using SNA LU0 and LU1 protocols
- Other hardware office products were coming to market
 - Displaywriter- stand alone workstation - LU6.2 connection
 - Snowball- TP connected fax machine - LU6.2 connection
 - IBM6670- TP connected copier/printer - LU4 connection
- Centralised Office system required
 - DISOSS which ran on CICS and IMS
 - Introduced CICS licences into a 1,000 IMS shops because IMS had decided they would not implement LU6.2!



3730 text formatted to IBM6670

Mr E Jones
C.C.
Mr G H Cottrell
Mr F Hinch
Mr G K Hunt
Mr B Hunter

9 December 1979

PROJECT: 'Ammonia Expansion Bellows'

This letter confirms my telephone request of 5th December to Mr B Hunter for your assurance on the following points:

- 1) That there will be sleeves inside all bellows and that, where sleeves are of the butt type shown as detail 1 on drawing 7767-FB4-99, the gap between the sleeve and the next piece of ductwork will be such that a small but definite clearance remains under service conditions.
- 2) That clamping strips have been ordered for all bellows for delivery with the ductwork.

Inspection of drawing issues currently held by us and marked 'Approved for Construction' reveals confusion on both the above matters. Certainly in some cases the 'typical' clearance between sleeve and ductwork shown on the drawings is less than the expansion tabulated on drawing 7767-FB4-99.

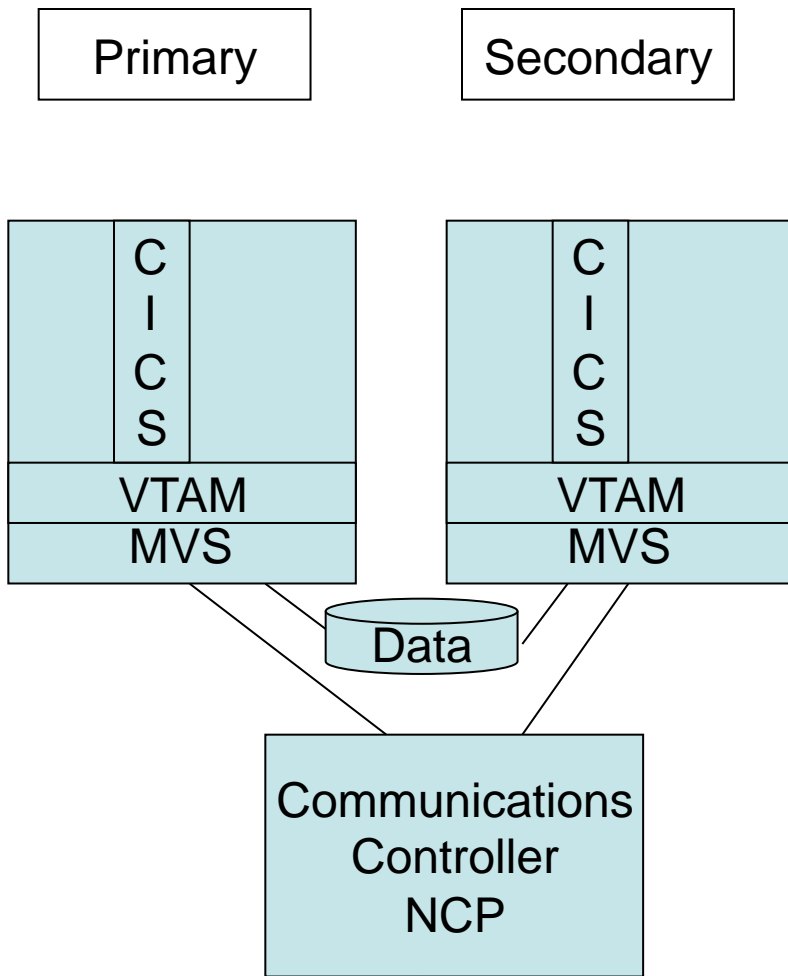
I would be grateful for an early reply by letter or telex.

G. W. Robinson

G W Robinson,
Product Design Manager,
Office Communication Systems
gwr.seo.9.12



CICS/MVS



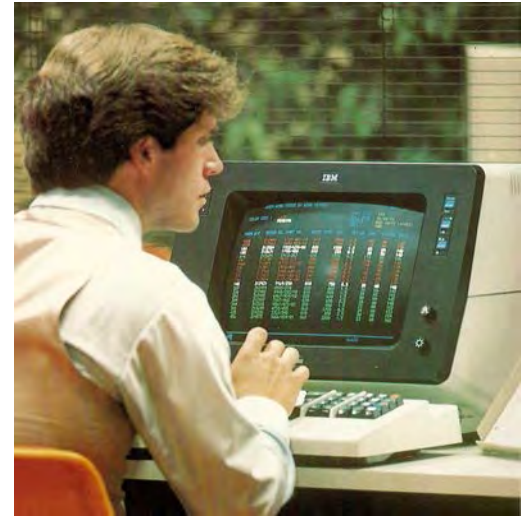
- Major competition was Tandem Non-Stop
- XRF was created
 - Primary and back-up
 - Switch in Network Control Program when primary fails to respond
 - Increased availability because no network restart
 - Shared DASDI for data
- Built offsite in Millbrook

Meanwhile back in Hursley

- CICS restructure
 - Kernel
 - Domains
 - Storage control
 - Dispatcher
 - Use of Z
 - Saved approximately 9% of development cost
 - Probably 10 times that in service costs
 - Queen's Award to Industry
- Shipped as Version 3

GDDM

- A concerted attempt was made very late 1970s to amalgamate the various mapping technologies
 - CICS - BMS
 - IMS – MFS
 - ISPF – MVS
 - Died a death
- A simpler product was created with its own API for CICS and IMS to support 3270 green screen and the upcoming colour terminal the 3279 with proper graphical support
 - Technology was programmed symbol sets
 - Extended for printers
 - The key addition was the Interactive Chart Utility (ICU) which took numerical input
 - Drew graphs and various types of charts
 - Produced printed reports

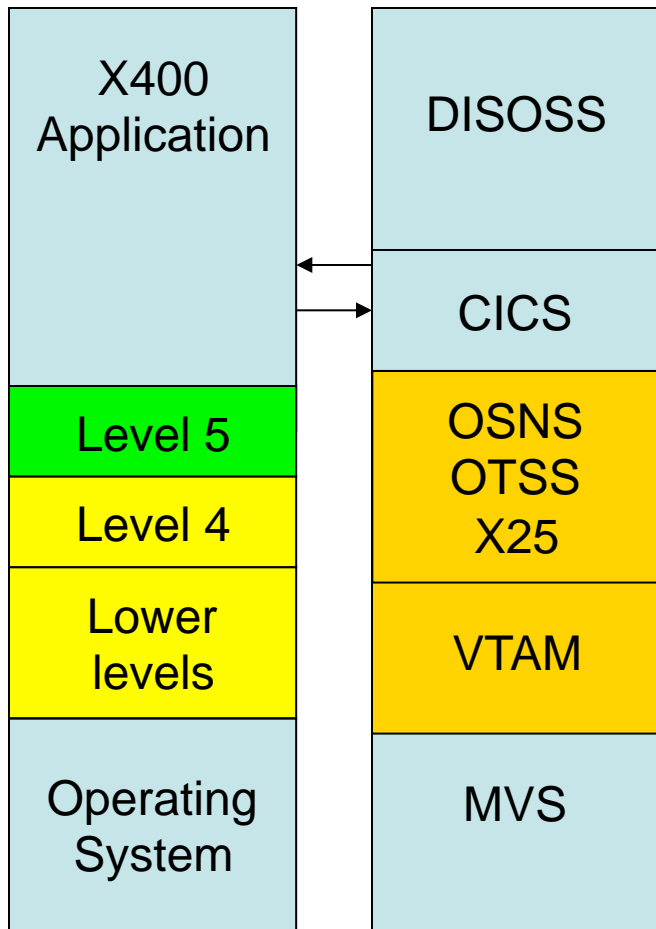


OSI and SNA

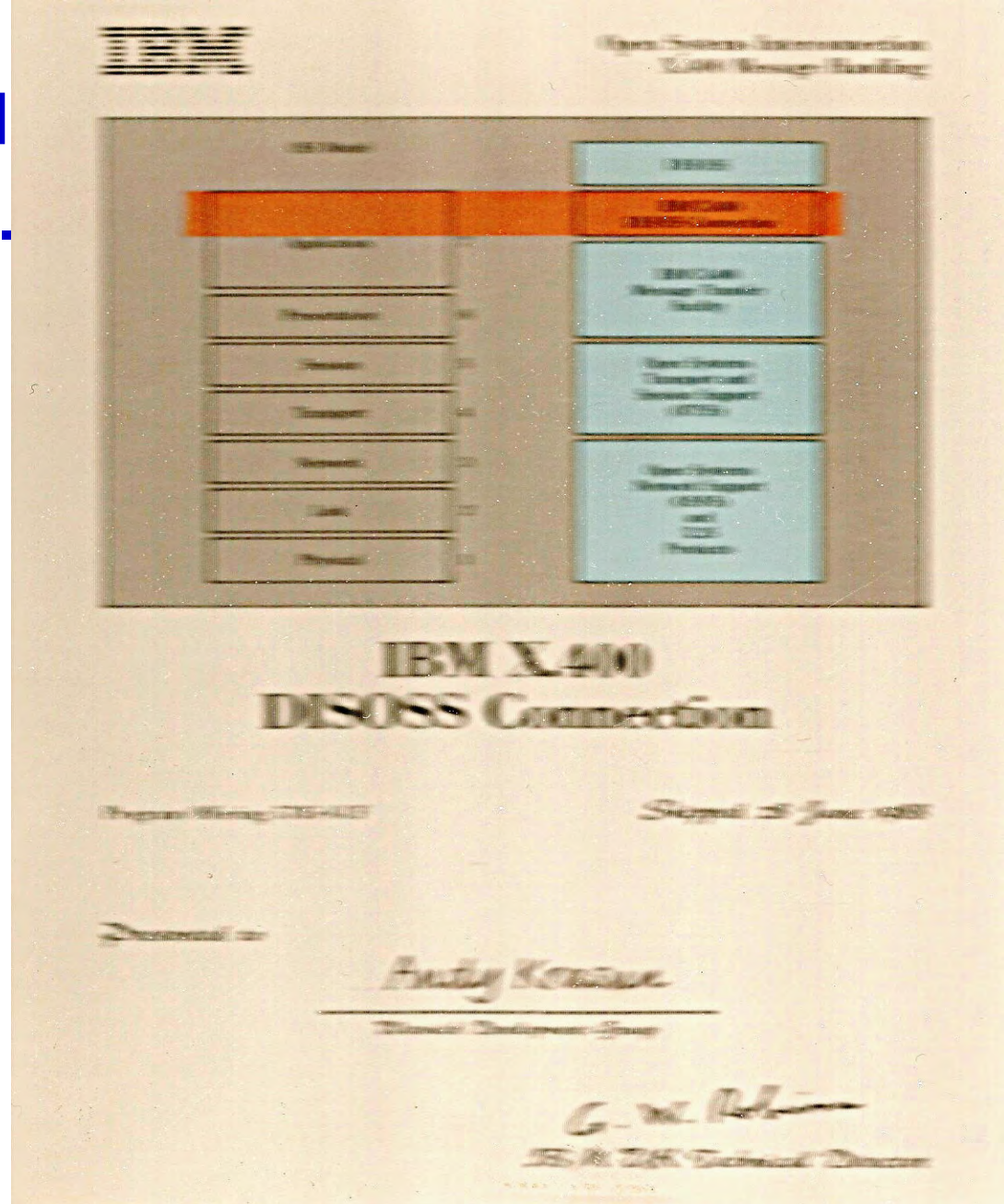
Presentation	Transaction services
Application	Functional management
Session	Data Flow
Transmission	Transmission
Network	Path control
Data Link	Data Link
Physical	Physical

- Bottom layers very similar at top of level 3
- Top of level 5 is where CICS is
- Each OSI level architected with defined interfaces
- OSI connectivity needed for government tender
- Levels 1, 2 and three existed
- Bought in software for levels 4 and 5
 - Beautifully architected as per the standard
 - A performance nightmare

Eventu



It worked
Wasn't very pretty
We were able to bid



GWRobinson and CICS managers and senior techies



BCS October 2006

OS/2 Presentation Manager

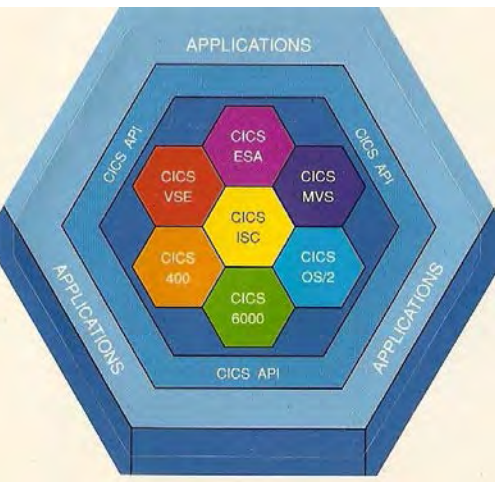
- In the beginning was DOS with many of the same challenges as an early CICS
 - Memory constraints
 - Not enough cycles
 - Co-operative program design
- IBM and Microsoft decided to work on a follow-on OS/2 which would alleviate the original deficiencies
- Development in three places
 - Boca Raton, FL – hardware and part of the OS
 - Hursley – Presentation Manager
 - Redmond, WA – software
- Design of PM had learnt from GDDM and associated hardware
- Applications needed to be designed round an interrupt loop so mouse movements were ‘immediately’ handled
- OS/2 was late...
- Meanwhile Windows 3 was being developed quietly in Redmond and was ready when the Intel386 hit the market
- So the rest is history!

Message and Queuing

- TCAM was widely used in the Brokerage business with messages being written to disk for integrity
- IBM decided to stabilise TCAM to reduce development cost because much of the function duplicated the VTAM function
- The standard Task Force was assembled
- After customer visits to Wall Street the question was “Where is the TCAM follow-on?”
 - so MQ was born

The rest of the CICS family and CICS OS/2

- CICS family expanded in the late 80' s
 - CICS AIX, CICS 400, CICS OS/2 and later the CICS client
 - Different technologies used for the implementation
 - AIX – existing software blocks SFS, DCE etc.
 - AS 400 – built from scratch
 - CICS DOS soon to be CICS OS/2 bought in from a small vendor
 - Written in COBOL
 - BTRIEVE as file manager
 - Hardware protection between applications
 - Conflicting requirements for a distributed server or client
 - » Fell between the two stools at the beginning
 - » Led to a cut down version becoming the CICS client in long a weekend in Millbrook
 - Most successful, in number terms, over a third of a million sold



Summary

- There is so much I have had to leave out
 - VDL, SNATAM, SL1 (SIGMA), SAA, DPS, PC3270 G and GX etc.
 - IBM5444, Functional memory, Model 44, IBM3250, Model 135, IBM8775 etc.
 - Service teams, test teams, planning teams and the marketing teams
- We should have taken notice of Moore's law
 - 20/20 Hindsight is a wonderful thing
- The cartwheel of technology and the ensuing cartwheel of repeated errors
 - One can and should learn from history
- Hursley was, and is, a great place to have a career

Moore's Law

- Moore's original statement can be found in his publication "Cramming more components onto integrated circuits", [*Electronics Magazine* 19 April, 1965](#)
- “The complexity for minimum component costs has increased at a rate of roughly a factor of two per year ... Certainly over the short term this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. That means by 1975, the number of components per integrated circuit for minimum cost will be 65,000. I believe that such a large circuit can be built on a single wafer.”