FORTRAN

Some facts and many myths

by

Kees Ampt

What?

- Fortran
- fortran
- FORTRAN
- USASI FORTRAN
- FORTRAN II
- FORTRAN IV
- FORTRAN 66
- FORTRAN 77
- Fortran 8X
- ….

The old days

Pioneers of the Fortran language

L to R: Richard Goldberg, Robert Nelson, Lois Haibt, Roy Nutt, Irv Ziller, Sheldon Best, Harlan Herrick, John Backus, Peter Sheridan.

THE FORTRAN AUTOMATIC CODING SYSTEM FOR THE IBM 704 EDPM®

This manual supersedes all earlier information about the FORTRAN system. It describes the system which will be made available during late 1956, and is intended to permit planning and FORTRAN coding in advance of that time. An Introductory Programmer's Manual and an Operator's Manual will also be issued.

APPLIED SCIENCE DIVISION
AND PROGRAMMING RESEARCH DEPT.
International Business Machines Corporation
590 Madison Ave., New York 22, N. Y.

WORKING COMMITTEE

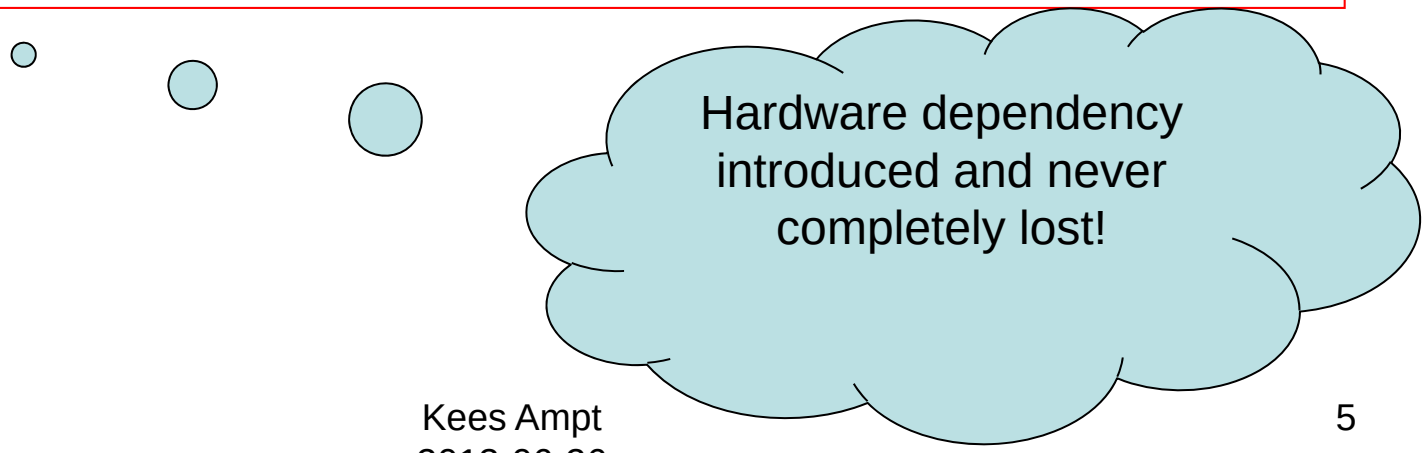
J. W. BACKUS L. B. MITCHELL

Efficiency of the Object Program

Object programs produced by FORTRAN *will be nearly as efficient as those written by good programmers.*

NOTE 1. There are two — signs. Only the 11-punch minus may be used in source program cards. Either minus may be used in input data to the object program; object program output has the 8-4 minus.

NOTE 2. The $ character can be used in FORTRAN only as Hollerith text in a FORMAT statement.



Coding

	CARD	BCD TAPE	704		CARD	BCD TAPE	704		CARD	BCD TAPE	704		CARD	BCD TAPE	704
1	1	01	01	A	12 1	61	21	J	11 1	41	41	/	0 1	21	61
2	2	02	02	B	12 2	62	22	K	11 2	42	42	S	0 2	22	62
3	3											T	0 3	23	63
4	4										44	U	0 4	24	64
5	5	05	05	E	12 5	65	25	N	11 5	45	45	V	0 5	25	65
6	6	06	06	F	12 6	66	26	O	11 6	46	46	W	0 6	26	66
7	7	07	07	G	12 7	67	27	P	11 7	47	47	X	0 7	27	67
8	8	10	10	H	12 8	70	30	Q	11 8	50	50	Y	0 8	30	70
9	9	11	11	I	12 9	71	31	R	11 9	51	51	Z	0 9	31	71
Blank		20	60	+	12	60	20	–	11	40	40	0	0	12	00
=	8-3	13	13	.	12 8-3	73	33	$	11 8-3	53	53	,	0 8-3	33	73
–	8-4	14	14	)	12 8-4	74	34	*	11 8-4	54	54	(	0 8-4	34	74

What is this all about?

All statements in early FORTRAN

STATEMENT	NORMAL SEQUENCING
a = b	Next executable statement
GO TO n	Statement n
GO TO n, (n_1,n_2,...,n_m)	Statement last assigned
ASSIGN i TO n	Next executable statement
GO TO (n_1,n_2,...,n_m), i	Statement n_i
IF (a) n_1,n_2,n_3	Statement n_1,n_2,n_3 as a less than, =, or greater than 0
SENSE LIGHT i	Next executable statement
IF (SENSE LIGHT i) n_1,n_2	Statement n_1,n_2 as Sense Light i ON or OFF
IF (SENSE SWITCH i) n_1,n_2	" " " as Sense Switch i DOWN or UP
IF ACCUMULATOR OVERFLOW n_1,n_2	" " " as Accumulator Overflow trigger ON or OFF
IF QUOTIENT OVERFLOW n_1,n_2	" " " as MQ Overflow trigger ON or OFF
IF DIVIDE CHECK n_1,n_2	" " " as Divide Check trigger ON or OFF
PAUSE or PAUSE n	Next executable statement
STOP or STOP n	Terminates program
DO n i = m_1,m_2 or DO n i = m_1,m_2,m_3	Next executable statement
CONTINUE	" " "
FORMAT (Specification)	Not executed
READ n, List	Next executable statement
READ INPUT TAPE i, n, List	" " "
PUNCH n, List	" " "
PRINT n, List	" " "
WRITE OUTPUT TAPE i, n, List	" " "
READ TAPE i, List	" " "
READ DRUM i, j, List	" " "
WRITE TAPE i, List	" " "
WRITE DRUM i, j, List	" " "
END FILE i	" " "
REWIND i	" " "
BACKSPACE i	" " "
DIMENSION v, v, v, ...	Not executed
EQUIVALENCE (a,b,c,...), (d,e,f,...), ...	" "
FREQUENCY n(i,j,...), m(k,l,...), ...	" "

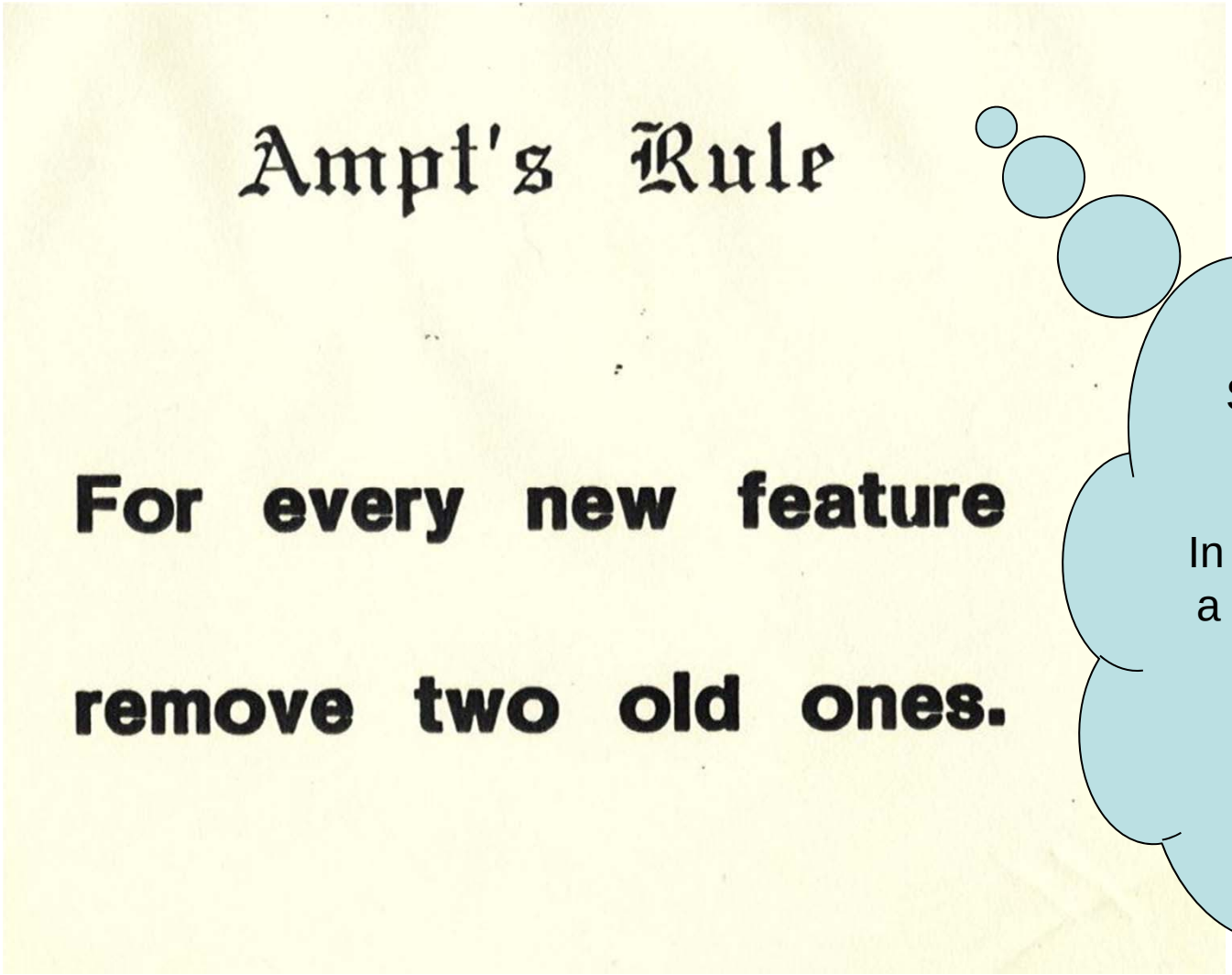
Information Processing?

- Why has success so many
- Upward compatibility: fathers and every error hardly of .. mothers

Information Processing?

- Why has success so many fathers and hardly mothers?
- Upward compatibility:
  every error of the past will be carried forward for ever.
- Still we want more features.

The simple solution



Consequence?
Still less features?
However, most is cosmetic.
In the past we needed a DO-loop to copy an array.
Just the = will do.

Standardization?

- Yes, let’s standardize!
- But do it my way
- Originally a prerogative of the United States of America

How did I got involved in standardization?

- The Netherlands Postal & Telecommunication services installed a new and very powerful Univac 1110.
- Programmers cursed the system, because FORTRAN programs they got from elsewhere did not run. Why?
- Univac had tried to inplement the standard as good as possible, IBM not.

NOVI

- NOVI was in the 70s the leading institute in the Netherlands for computer education.
- Also FORTRAN as module T4.
- Requirement: Standard FORTRAN.
- However, exam committee had never seen the standard as they believed their IBM compilers to be THE standard.

Part of Univac manual

PROPOSED USASI FORTRAN IV

The old trick, use a draft.

1. INTRODUCTION

1.1 PURPOSE. This standard establishes the form for and the interpretation of programs expressed in the FORTRAN language for the purpose of promoting a high degree of interchangeability of such programs for use on a variety of automatic data processing systems. A processor shall conform to this standard provided it accepts, and interprets as specified, at least those forms and relationships described herein.

Insofar as the interpretation of the form and relationships described are not affected, any statement of requirement could be replaced by a statement expressing that the standard does not provide an interpretation unless the requirement is met. Further, any statement of prohibition could be replaced by a statement expressing that the standard does not provide an interpretation when the prohibition is violated.

1.2 SCOPE. This standard establishes:

(1) The form of a program written in the FORTRAN language.

(2) The form of writing input data to be processed by such a program operating on automatic data processing systems.

(3) Rules for interpreting the meaning of such a program.

(4) The form of the output data resulting from the use of such a program on automatic data processing systems, provided that the rules of interpretation establish an interpretation

A *main program* is a set of statements and comments not containing a FUNCTION, SUBROUTINE, or BLOCK DATA statement (9.1.5).

A *subprogram* is similar to a main program but is headed by a BLOCK DATA, FUNCTION, or SUBROUTINE statement. A subprogram headed by a BLOCK DATA statement is called a specification subprogram. A subprogram headed by a FUNCTION or SUBROUTINE statement is called a procedure subprogram (9.1.3, 9.1.4).

The term *program unit* will refer to either a main program or subprogram (9.1.7).

Any program unit except a specification subprogram may reference an *external procedure* (Section 8).

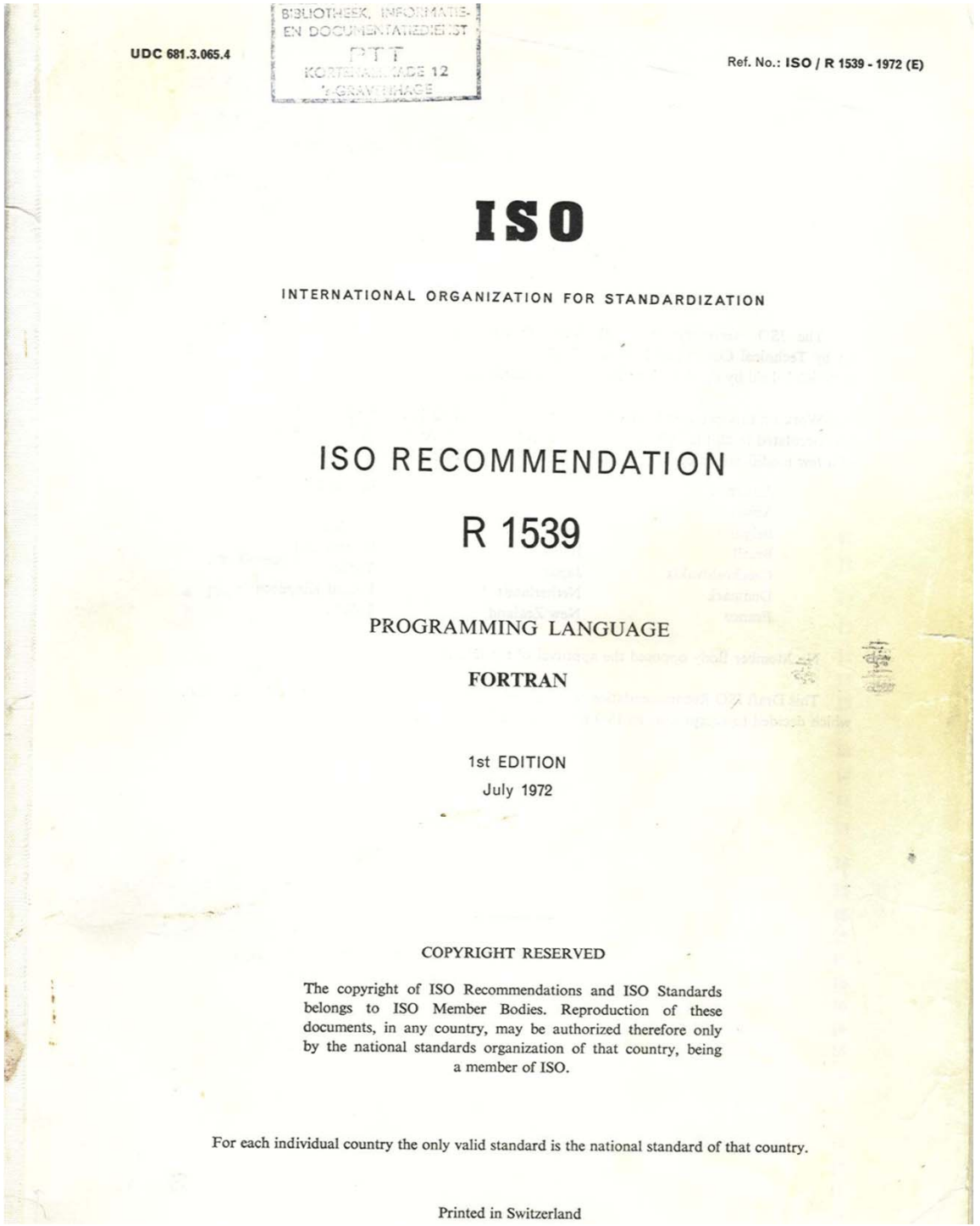
An external procedure that is defined by FORTRAN statements is called a *procedure subprogram*. External procedures also may be defined by other means. An external procedure may be an external function or an external subroutine. An external function defined by FORTRAN statements headed by a FUNCTION statement is called a *function subprogram*. An external subroutine defined by FORTRAN statements headed by a SUBROUTINE statement is called a *subroutine subprogram* (Sections 8 and 9).

Any program unit consists of *statements* and *comments*. A statement is divided into physical sections called *lines*, the first of which is called an *initial line* and the rest of which are called *continuation lines* (3.2).

Many people did not like FORTRAN

- At Netherlands universities it was never part of the curriculum, although everybody programmed in it.
- Hence, all those horrible programs.
- Just kicking against FORTRAN too easy.
- So I became a member of the Netherland FORTRAN Specialists Group.

Finally a genuine standard



UDC 681.3.065.4

Ref. No.: ISO / R 1539 - 1972 (E)

ISO

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION

ISO RECOMMENDATION

R 1539

PROGRAMMING LANGUAGE

FORTRAN

1st EDITION

July 1972

Printed in Switzerland

International Fortran Standardization Meetings, 1963-1999

Date	Place	Attendance Individual	Attendance Member Body	Notes
Ad-hoc Fortran meetings at TC97/SC5 meetings				
1963/06	Berlin, Germany	?	?	ad hoc meeting, details not known
1964	New York, USA			no information
1965/09	Vedboek, Denmark			no information
1967/11	Paris, France			no information
1972/11/28-12/01	Washington, USA	10	4	ad hoc meeting for one day (?)
1977/11/14-17	The Hague, Netherlands	15	7	ad hoc meeting for 3 half-day sessions on 11/15-16
1979/11/12-16	Turin, Italy	26	9	ad hoc meeting for 2 days on 11/12-13
1981/10/05-09	Windsor, UK	?	?	ad hoc meeting for half-day
1983/09/26-30	Ottawa, Canada	?	?	informal meeting for two hours on 09/28
SC5 Fortran Experts' Meetings				
1978/11/27-30	London, UK	36	7	
1980/10/20-23	Amsterdam, Netherlands	35	9	
1982/06/14-17	Vienna, Austria	38	10	

Date	Place	Agenda (Pre/Pro)	Mins	Resolutions
SC5/WG9				
1984/04/09-12	Geneva, Switzerland	36	9	X3J3/163
SC22/WG5				
1985/07/01-04	Bonn, Germany	38	9	X3J3/179 in mins

ISO/TC97/SC5

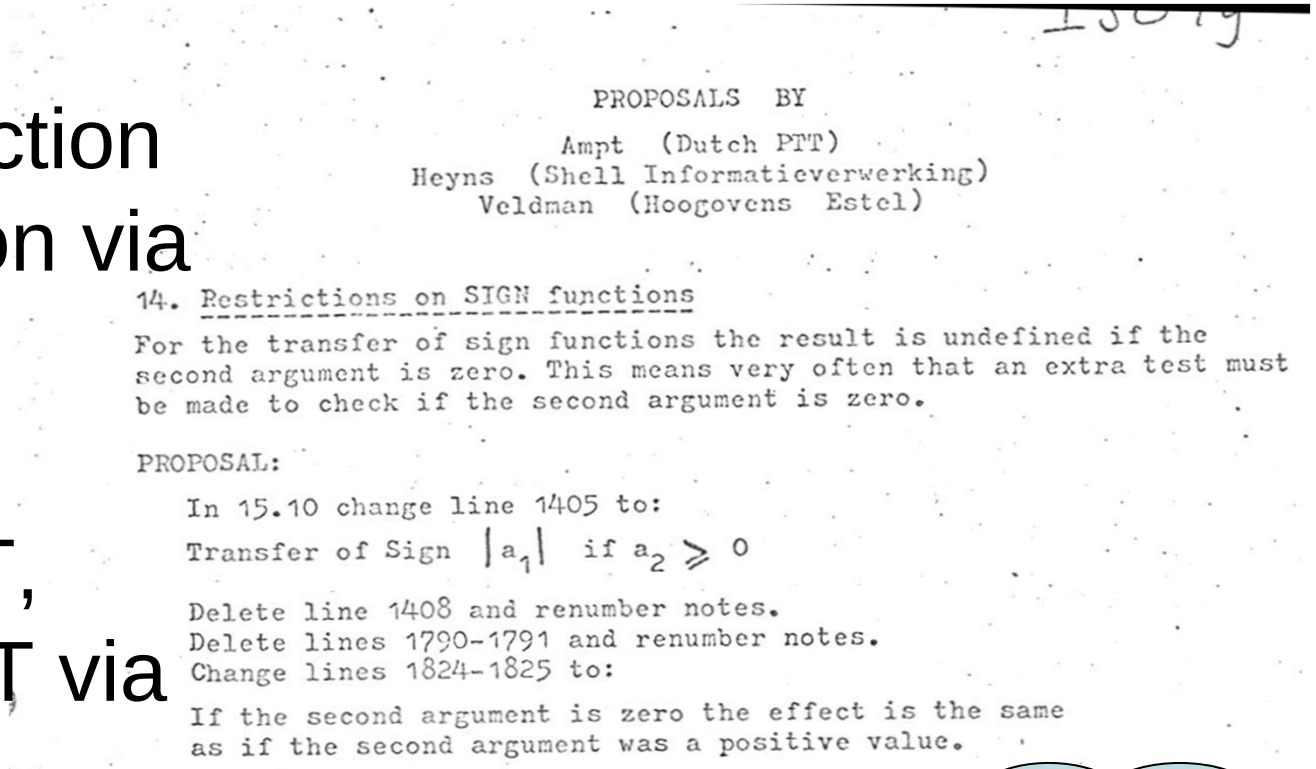
- The Programming Languages Committee had no meeting since 1972.
- Nobody wanted to host a meeting.
- Finally the Netherlands Standardization organization, NNI at that time, was convinced to host a meeting.

1977: Worldwide opposition?

- The ANSI X3J3 committee had prepared a new draft standard.
- Although more or less informal international comment might be taken into account.
- However, often too late due to surface mail.

Still some minor influence?

- SIGN function clarification via formal proposal
- LGE, LGT, LLE & LLT via informal contacts



PROPOSALS BY

Ampt (Dutch PTT)
Heyns (Shell Informatieverwerking)
Veldman (Hoogovens Estel)

14. Restrictions on SIGN functions

For the transfer of sign functions the result is undefined if the second argument is zero. This means very often that an extra test must be made to check if the second argument is zero.

PROPOSAL:

In 15.10 change line 1405 to:

Transfer of Sign $|a_1|$ if $a_2 \geqslant 0$

Delete line 1408 and renumber notes.
Delete lines 1790-1791 and renumber notes.
Change lines 1824-1825 to:

If the second argument is zero the effect is the same as if the second argument was a positive value.

Character handling has always been weak in FORTRAN.

Formal and informal

- Draft resolutions had been prepared for synchronization of the US domestic project for FORTRAN and the international ISO one.
- Informal: find the US FORTRAN people: there is a lot negative to say about your proposal. However, it is much better than what we have.

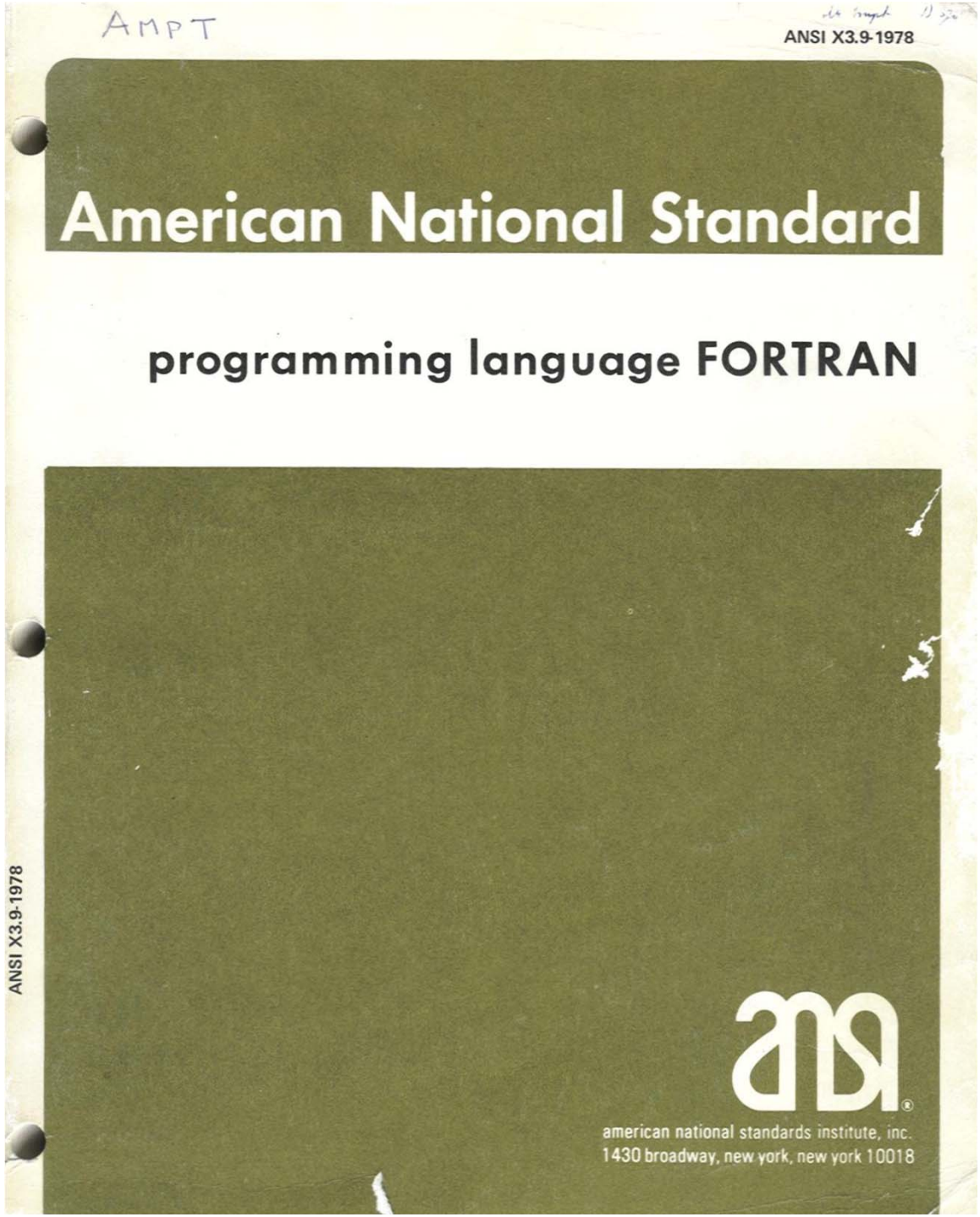
The deal

- As Netherlands we are even prepared to come forward with a motion to accept the US draft standard, provided:
- This is really the last time that you do it on your own without genuine international cooperation.

Experts

- The USA wanted an open organization.
- Therefore the text mentioned
  a group of experts.
- Due to this for several programming languages the **Experts Groups** were born. [The USA had suggested to take the FORTRAN road for all languages]

ANSI X3.9
FORTRAN
1978
known as
FORTRAN
77



The spirit of 77

12. Discussion of general schedules of X3J3

A. Program, scope & philosophy	1978
B. Proposals	1979
C. Preparation of draft std.	1980
D. - Processing of public comments	1981-1982
E. Approval of std.	1982

15. Solicitation of constructive criticism

A. Re scope

1) Is there a need for a bit data type

a) If so, what mechanization

b) Fortran already a large language

2) Attempt simplification by

a) Arithmetic statement function elimination

b) Assigned GOTO elimination

c) Arithmetic IF elimination

d) Increasing character set

e) Permitting 'reserved word' declarations

f) Add internal procedure

g) Unify repetitive execution

h) Simplify syntax

i) Add asynchronous I/O useful for large scientific applications and for industrial control applications.

The CDC challenge

- CDC had rather early a FORTRAN 77 compiler.
- I suggested that I would find an error in the compiler within 15 minutes.
- Just READ, WRITE, and loop.
- CDC had 60 bits words, thus N characters.
- A few times N+1 characters and BINGO.
- That N+1 character was interpreted as a control character. It took more than 3 months to fix it.

Industrial FORTRAN? Attempts by ISA

Matthew R. Gordon-Clark
Chairman ISA Standards
Committee SP 61
Scott Paper Company
Scott Plaza III
P H I L A D E L P H I A, PA 19113
U.S.A.

Your reference | Date | Our reference | * Telephone | Date 8th October 1976

Enclosures | * For information about this letter

Re

Dear Sir,

On behalf of the Dutch FORTRAN Study Group I have the pleasure to express our opinion upon your Draft Standard S 61.2. The Dutch FORTRAN Study Group is a subsection of the Dutch Normalisation Institute (NNI), a memberbody of ISO. The Group consists of representatives of both industrial and governmental organisations.
Unfortunately we have to vote for a disapproval, the reasons of which are in the appendix. However we appreciate your effort and we will be pleased to act in the Board of Review in the next review period and permit you to list the Group in the standard.

Yours faithfully,

C. Ampt

C.G.F. Ampt

Just rejection too simple?

- Cooperation between
- International Purdue Workshop [IPW]
- And
- European Workshop on Industrial Computer Systems TC1 [EWICS]

Ping-Pong

- Proposals went back and forward between Europe en the USA.
- People became friendly, accepting each other proposals.
- However, due to the snail mail letters were crossing each other.
- Finally…

I R T F

Industrial Real – Time

FORTRAN

WG 1

- IRTF was as new work item in ISO assigned to Working Group 1 Programming Languages for the Control of Industrial Processes.
- Germany had suggested this WG, taken the secretariat but had to withdraw due to financial cuts.
- Who else?

The European Commission

- At that time [early 80's] the European Commission had not the faintest idea about ISO standardization.
- EWICS TC1 jumped in.
- Funds needed to attend meetings.
- General EC rule: as a start only co-financing
- Due to huge success of Fortran Symposium in Amsterdam [thanks to X3J3 representatives] the Netherlands FORTRAN Specialists Group had some funds.

Organizers or Specialists?

- The ISO directives contained many shortcuts, e.g. have a resolution accepted at a SC meeting, thus avoiding a six month letter ballot.
- Classify comments as editorial or technical. Only in the latter case a new voting round required.
- Many X3J3 members still don’t understand this.

Organizers

- Get more votes

ISO/TC 97/SC 22 N 114

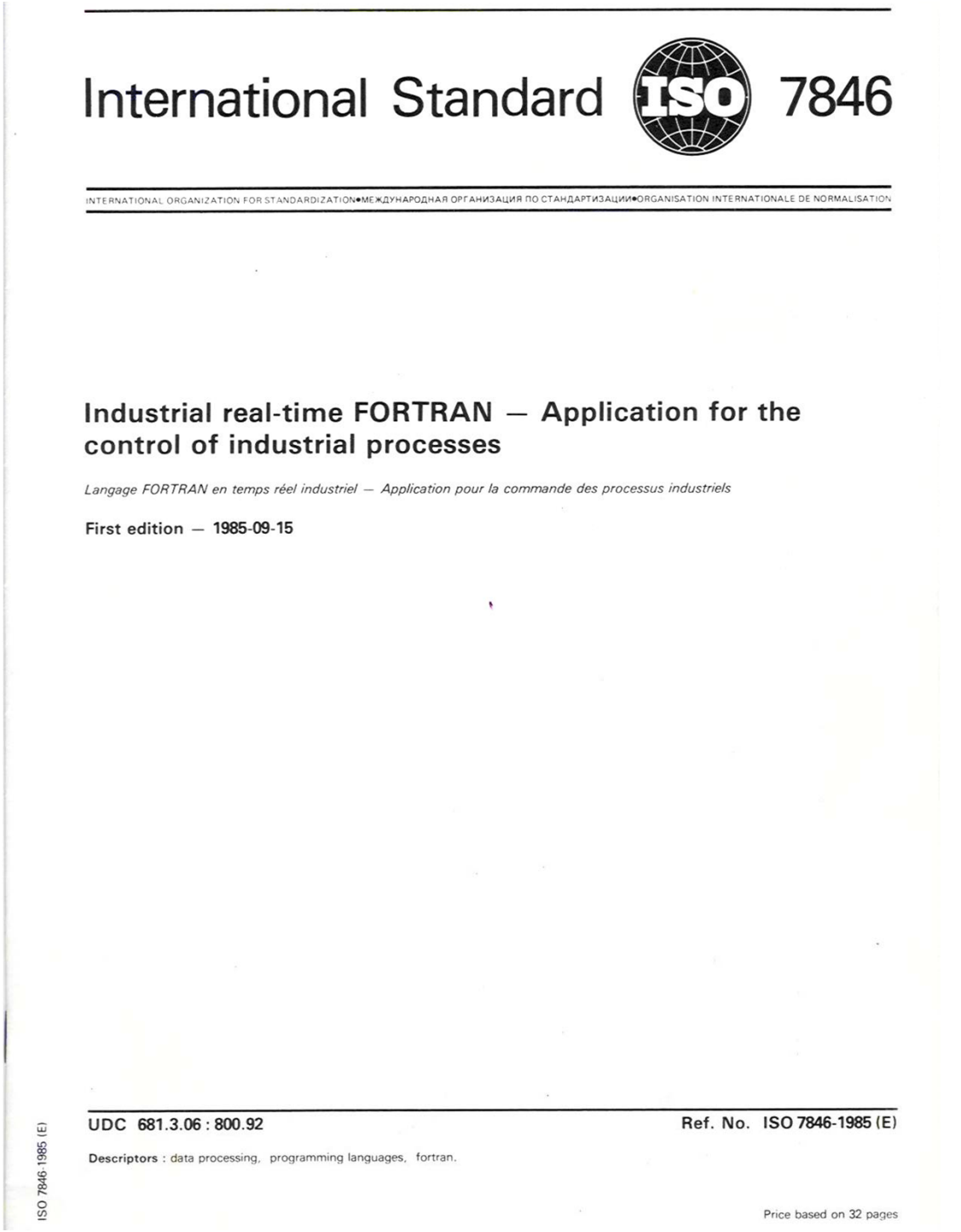
LIST OF DELEGATES
ISO/TC97/SC22
FIRST PLENARY MEETING - PARIS, FRANCE
1985-11-05/08

Please add * after name for head of delegation

Name	Member Country	Address
SIDI Jacqueline	FRANCE	BNI Tel : (1) 39 55 25 35 BP 105 F- 78153 LE CHESNAY CEDEX
LEROY Bertrand	FRANCE	BULL Tel : 33 (1) 39 02 52 69 68, route de Versailles 78430 LOUVECIENNES - FRANCE
HOLKA Tadeusz	FRANCE	BULL Tel : 33 (1) 39 02 42 47 68, route de Versailles 78430 LOUVECIENNES - FRANCE
GENUYS François	FRANCE	IBM FRANCE 36, AVENUE RAYMOND POÏNCARE 75116 PARIS
MAS Christian	FRANCE	IBM FRANCE Tel : 33 (1) 47 76 43 43 Tour Septentrion - CEDEX 9 92081 PARIS LA DEFENSE - FRANCE
JAAKKOLA Hannu	FINLAND	TAMPERE UNIVERSITY OF TECHNOLOGY P.O BOX 527 SF-33101 TAMPERE
BICKLE Tony	CANADA *	ENVIRONMENT CANADA, ASQME 5th Floor, Place Vincent Massey OTTAWA, CANADA, K1A-IC7
AMPT Kees	NETHERLANDS * BELGIUM *	Head of Delegation for 2 Member Bodies

ISO 7846

- Industrial real-time FORTRAN – Application for the control of industrial processes



International Standard ISO 7846

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Industrial real-time FORTRAN — Application for the control of industrial processes

Langage FORTRAN en temps réel industriel — Application pour la commande des processus industriels

First edition — 1985-09-15

UDC 681.3.06 : 800.92 **Ref. No. ISO 7846-1985 (E)**

Descriptors : data processing, programming languages, fortran.

Price based on 32 pages

ISO 7846-1985 (E)

Even a French version

- Really published by ISO Central Secretariat.
- So no canvas tricks.

Norme internationale ISO 7846

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION•МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ•ORGANISATION INTERNATIONALE DE NORMALISATION

Langage FORTRAN en temps réel industriel — Application pour la commmande des processus industriels

Industrial real-time FORTRAN — Application for the control of industrial processes

Première édition — 1985-09-15

CDU 681.3.06 : 800.92 Réf. nº : ISO 7846-1985 (F)

Descripteurs : traitement de l'information, langage de programmation, fortran.

6-1985 (F)

State Model & Transition Diagram

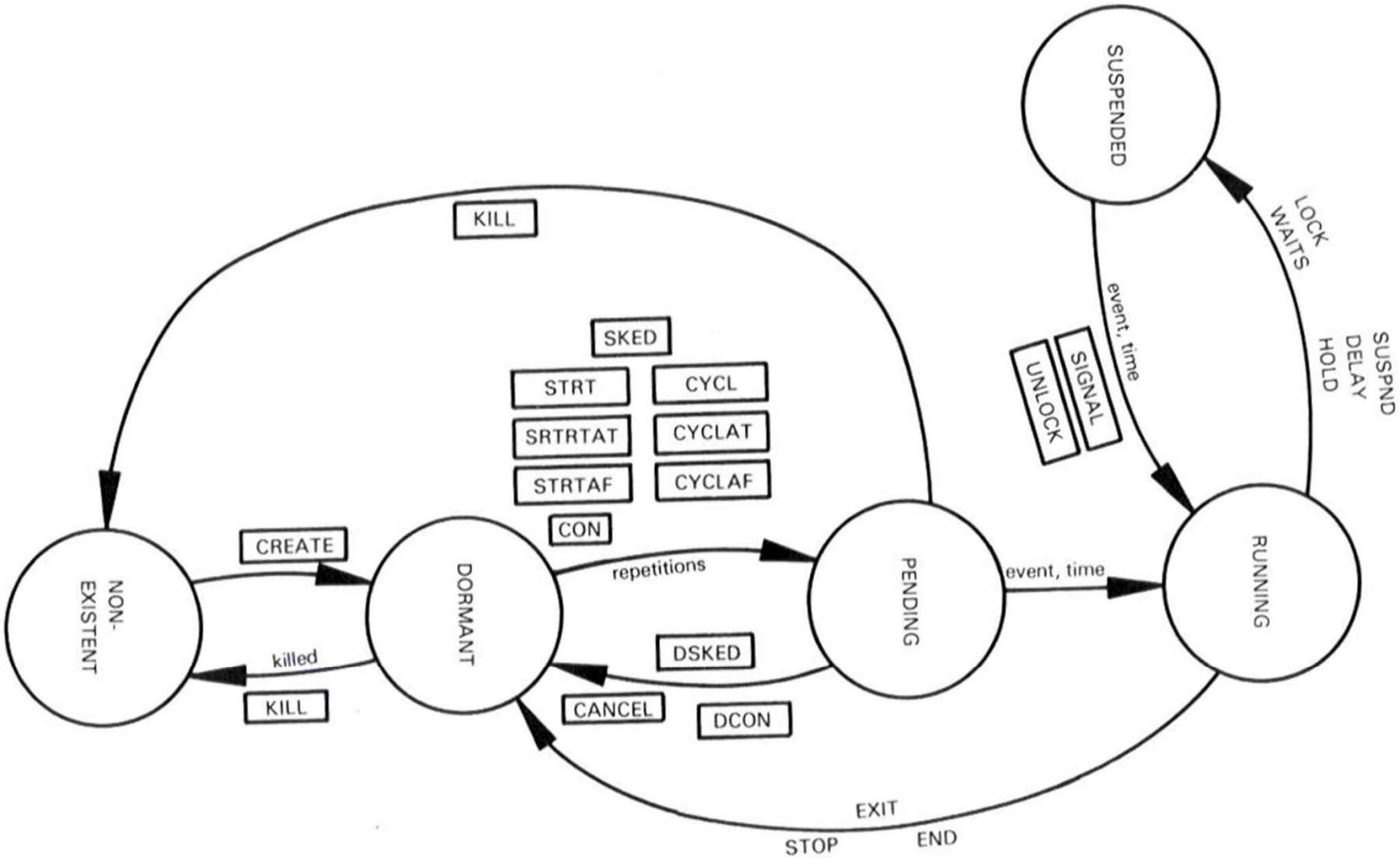


Figure — State model and transition diagram

New masters?

- ISO/TC97/SC5 had been converted into ISO/IEC/JTC1/SC22
- The Experts Groups should be converted into Working Groups
- X3J3 was still a roller coaster

16. Should blanks be significant? (16-8-7)

90th Mtg: Deprecate insignificant blanks. (16-0) Passed

91st Mtg: Delete the concept of significant blanks.
(11-11) Failed

92nd Mtg: Delete the concept of significant blanks.
(12-12) Failed

Issues Implemented

yes	no
10	6

Running Box Score

17. Should work continue to incorporate Event Handling in Fortran 8X? (20-0-3) (22-0-3)

91st Mtg: Replace Event Handling with Multitasking. (5-11) Failed
Endorse continued work on Events. (24-0) Passed
Endorse continued work on Multitasking. (24-0) Passed

92nd Mtg: Adopt exception handling via CONDITION/ENABLE.
(20-5) Passed
Delete Event Handling.
(24-0) Passed

Issues Implemented

yes	no
10	7

Running Box Score

History?

- Today there are no members trying to sabotage progress
- There are no members quarrelling for hours about nitty-gritty details
- The idea of standardization by implementation [if possible differently by various vendors] has been left behind for years
- Hardware issues are no longer debated.

Or?

Thank you for your attention!