Informal Summary

Meeting of ISO/TC97/SC22/WG5 - Fortran

University of Liverpool, England

August 3-7, 1987

and

Documents Distributed since August 3

Documents N230 - N266

X3J3/214

CONTENTS

MEETING OF ISO/TC97/SC22/WG5 - FORTRAN

University of Liverpool, England

August 3-7, 1987

Informal summary of the meeting

Present:

Jeanne T. Martin	(USA)	
Gerhard J. Schmitt	(Austria)	
Klaus Plasser	(Austria)	
Fausto Milinazzo	(Canada)	
Graham Warren	(Canada)	
James H. Matheny	(USA)	
Leonard J. Moss	(USA)	
E. Andrew Johnson	(USA)	
Leo G. F. ter Haar	(Netherlands)	
Richard W. Weaver	(USA)	
Neldon H. Marshall	(USA)	
Jerrold L. Wagener	(USA)	
T. Miles R. Ellis	(UK)	
Akira Ohwada	(Japan)	
Yasumasa Kenada	(Japan)	(Monday and Tuesday)
Hiroshi Ina	(Japan)	
Michael Geary	(UK)	
Brian T. Smith	(USA)	
Katsumi Yamamoto	(Japan)	
Kohmei Kuroda	(Japan)	
David T. Muxworthy	(UK)	
Brian L. Meek	(UK)	
Fred Hopper	(UK)	
Meinolf Munchhausen	(Germany)	
Christian Mas	(France)	
John Wilson	(UK)	
David M. Vallance	(UK)	
Aurelio A. Pollicini	(Italy)	
Karl-Heinz Rotthauser	(Germany)	
Richard Hendrickson	(USA)	
Carl Burch	(USA)	
Kurt W. Hirchert	(USA)	
Lawrie Schonfelder	(UK)	
Steve Morgan	(UK)	
Alan Wilson	(UK)	
Jeanne Adams	(USA)	
Colin K. Mackinnon	(UK)	(From Tuesday)
Jeremy Du Croz	(UK)	(Thursday and Friday)

1. OPENING OF THE MEETING

The WG5 Convener, Jeanne Martin, opened the meeting at 9.00 a.m. August 3, 1987.

2. WELCOME OF THE DELEGATES

The delegates were welcomed to the University and to the City of Liverpool by the Vice-Chancellor of the University, Professor Graham Davis.

3. ADOPTION OF THE AGENDA

The provisional Agenda, dated June 10, 1987, was adopted with one change:

Delete item 8 (Approval of the minutes of the Halifax Meeting)

as these minutes were not yet available.

The following items were added to the Agenda:

13. (j)	Discussion of Passed-on Precision
17.5	Discussion of S8 Copyright
(Under 18)	Discussion of PHIGS binding

4. ELECTION OF THE CHAIR

Jeanne Adams was proposed, seconded and elected unanimously to the Chair for the meeting.

5. NATIONAL ACTIVITY REPORTS

AUSTRIA The Austrian delegates had reviewed the X3J3 progress and comments on the WG5 ballot.

CANADA The Canadian Fortran Working Group has 8 members and holds 4 meetings per year. Canada had voted Yes in the WG5 letter ballot so that a wider audience could comment on S8. Canada had produced a position paper for the Liverpool WG5 meeting.

FRANCE The French Fortran Working Group has 8 members. The Group had followed the progress of S8 and had sent a letter ballot voting Yes with comments. The French Group had produced a position paper which now effectively votes No with comments which would be treated by X3J3 in the same way as any other comments.

It was noted that the next WG5 meeting has been

provisionally scheduled for September 19-23, 1988 in Paris but that these dates might conflict with the meeting of the SC22 Advisory Committee in Tokyo.

GERMANY

The Fortran Working Group of DIN has 7 members. The Group had discussed the S8 document and comments with No votes together with their X3J3 responses. A German position paper had been prepared. WG5 members' attention was called to the document called "Fortran SC" which is a language similar to Fortran 8X suitable for the development of scientific and numerical programs. The Germans have a compiler for this language.

ITALY

Agreement has been reached with the Italian Standards Organisation. An official letter will shortly be sent to SC22 and WG5. No report has yet been produced because there is no Italian Working Group. There is an express intention to have an Italian National Report in 1988. Italy needs to build a group because Aurelio Pollicini cannot express a personal opinion as a national position.

Pollicini is planning to organise a Fortran 8X course for existing Fortran 77 programmers so that programmers will be able to see in practice what Fortran 8X looks like.

JAPAN

The Japanese Group has been examining the Fortran 8X draft for mnay years. The Japanese position is Yes with comments. The Japanese are strongly proposing multibyte character codes (e.g. for Kanji).

NETHERLANDS

Activities have been restricted to discussions on the S8 proposals. The Netherlands Group consists of 8 members and meetings of the Group are convened approximately every 3 months. The Group did not take part in the WG5 letter ballot because of time constraints. The Dutch position is Yes with comments. The Netherlands delegate has a position paper to present.

Fortran 8X is of interest to Shell. S8.104 is a better document than the previous versions and X3J3 are to be congratulated on their work.

A major reservation is the way in which C is becoming a very popular language because of its functionality. The Dutch Group did not agree with the removal of Fortran 8X language features to an Appendix.

UK

The UK Group has tabled an activity report. The UK position can be informally stated as "get the draft standard out to public comment as soon as possible".

The UK Group drew X3J3's and WG5's attention to the forthcoming British Standard: **Method for Specifying Fortran Language Processors**. This standard will be published by BSI in November 1987 as BS 6832. A set of special concessionary order forms will be sent by the BSI to Brian Meek and to Lawrie Schonfelder so that X3J3 and WG5 members can obtain copies of the standard at a discount. (Note added later: The forms were available during the meeting and stated September as the publication date.)

USA

(Report of the X3J3 committee) X3J3 had met 5 times since the Halifax meeting. 5 more meetings are planned for 1987/88. Four Fortran Forums have been favourably received. One of the main causes for objections to Fortran 8X in these Forums is the presence of obsolescent features.

The main documents that had been produced were:

X3J3/S8.104	
X3J3/S9	(Comments on Fortran 8X)
X3J3/S10	(Fortran 8X historical document)

The following ballots had been held:

X3J3 letter ballot 1986	NO	(44% voted Yes)
WG5 letter ballot	YES	(93% Yes)
X3J3 letter ballot	YES	(81% Yes)
X3J3 Roll Call vote	YES	(74% Yes)

6. APPOINTMENT OF THE DRAFTING COMMITTEE

The following delegates were appointed to the drafting committee by the Chair:

Graham Warren
Gerhard Schmitt (Drafting Committee Chair)
Aurelio Pollicini
David Muxworthy
Dick Hendrickson
Karl-Heinz Rottnauser
Hiroshi Ina

7. STATUS OF THE X3J3 RESPONSES TO THE HALIFAX RESOLUTIONS (J WAGENER)

The latest version of the status of these resolutions is in document ISO/TC97/SC22/WG5-N227 (Standing Document 6). This represents a first draft of the wording of the X3J3 responses. Jerry Wagener's personal feeling regarding the X3J3 attitude to each resolution is summarised as follows:

1a	SF	Key:
1b	MF	

2	MU	
3	?	
4	SF	
5	SF	
6	SF	
7	SF	
8	SF	
9	SF	
10	SF	
11	MU	
12	SF	
13	SU	Two levels of deprecated features
14	SU	Blanks made insignificant
15	SF	RANDOM intrinsic subroutine adopted
16	SF	
17	Failed in WG5	
18	Failed in WG5	
19	SF	
20	MU	Investigated by X3J3: Proposal failed
21	Failed in WG5	
22	SU	Replaced by NAMELIST
23	MU	Also a Bonn resolution. Not possible in confines of current syntax.

SF Strongly Favourable
MF Mildly Favourable
MU Mildly Unfavourable
SU Strongly Unfavourable
? Neutral

Items 1 to 9 are procedural; items 10 to 23 are technical.

Strongly favourable technical issues are:

Processor conformance
Data Abstraction
RANDOM intrinsic
Character intrinsics
Structure constructors

Strongly unfavourable technical issues are:

Deprecated features
Significant blanks
Name-directed I/O

Mildly unfavourable technical issues are:

Pointers
Operator renaming
Procedure interfaces

7.1 Discussion

MEEK suggested that the texts of the resolutions should be interspersed with responses so it was always obvious what any response referred to. **WAGENER** expressed his agreement to this suggestion.

MARTIN noted that Document N205 contained the text of the resolutions.

MEEK also felt that, if a WG5 resolution was not accepted by X3J3, the text of the failed resolution should appear with the reasons for its non-acceptance.

HIRCHERT remarked that a negative vote does not necessarily mean that a technical reason is given.

SCHMITT suggested, and the Chair agreed, that, if a resolution is re-affirmed by WG5, a new resolution with new text should be prepared so that old resolutions do not need to be carried forward.

Concern was expressed at the time taken to get the draft standard out to public comment. The Chair noted that public review in the USA is only possible when the X3J3 parent X3 committee allows it.

SCHMITT reminded the meeting of the Halifax resolution "send out the draft for public review prior to the next (Liverpool) WG5 meeeting".

ELLIS noted that the review was delayed and that the Halifax resolution was also aimed at the X3 committee.

8. STATUS OF FORTRAN 8X IN X3J3 (JEANNE ADAMS)

The ballot on the question of S8 public comment had passed. A response had been prepared based on the votes cast which contained both positive and negative votes as well as the final roll call vote. The SPARK committee had checked the draft for compliance with drafting standards in early July. On July 20, 1987, a ballot had been sent out to X3 members asking them to vote on whether S8 could now be sent out for public review. If the ballot is favourable, S8 will be available, bound together with the response document, from Global Engineering of Washington D.C. at a cost of maybe $100 per copy.

8.1 Discussion

It was still hoped to keep the ANSI and ISO public reviews in step. It was noted by **WEAVER** that the timescales involved in procedural rules will probably mean that the US review cannot start until October 1987.

As a result of the need to circulate the S8 document for public comment, there was discussion amongst the delegates (notably **SCHMITT, MEEK, WILSON, HIRCHERT, VALLANCE, WEAVER** and **ADAMS**) about the copyright status of the S8 document both in the USA and elsewhere in the world. **ADAMS** noted that S8 was an internal X3J3 document. **MEEK** noted that S8 was also an SC22 document. Andy Johnson was asked to obtain a statement on the copyright of S8 from CBEMA. The response was allocated to Agenda item 17.5.

MEEK wanted the drafting committee to re-affirm any outstanding resolutions from Halifax. The Chair instructed Gerhard Schmitt to renumber and re-present all such items.

SCHONFELDER proposed that the drafting commitee should prepare a resolution along the lines: "A set of WG5 resolutions from Halifax was

put before X3J3 and a set of responses have been received. Does WG5 think these responses are adequate?" The Chair noted that, as X3J3 had considered the Halifax resolutions formally, WG5 should make a formal response to X3J3.

9. PROJECT STAGES IN ISO (JEANNE MARTIN)

The ISO standardisation process consists of the following stages:

Stage 1: Development of WD (Working Draft)

Stage 2: Registration of DP (Draft Proposal)
- 3 month vote/comment period amongst P members of SC22
- 2 month vote period for successive drafts
- SC Secretariat decides when to proceed

Stage 3: Registration of DIS (Draft International Standard)
- 6 month vote period in TC97
- 75% of votes cast must approve or revert to Stage 2

Stage 4: Four months allowed to prepare (possibly revised) DIS and report of processing history for Central Secretariat.

Stage 5: If no more than 2 negative votes cast in Stage 3, go to Stage 7. Otherwise Council decides outcome within 6 weeks.

Stage 6: Council decides to publish or revert to Stage 2

Stage 7: Publication of IS (International Standard)

There was confusion over the number of P Members in TC97 and the countries that they represented. The Chair nominated Jeanne Martin to present the list later in the meeting. (The list in N235 inadvertantly omitted China and the Netherlands.)

Jeanne Martin had sent the S8 document to Stage 2.

9.1 Discussion

ELLIS asked what happened if the X3J3 review period changes S8 during the ISO review.

MEEK noted that the US comment period merely produces US comments for the ISO review.

SCHMITT noted the following:

(a) The name of TC97 will shortly change to Joint Technical Committee Number 1. The first meeting of this committee is in November 1987.

(b) Discussion had so far left out the problem of the French translation of S8.

(c) The S8 document is a revision of an existing standard and will be allocated the same number as its predecessor. ISO will normally print a suitable cover page for the ANSI standard text.

(d) We may need special wording in the ISO Standard that is not in the ANSI Standard.

MARTIN noted that the wording in the S8 foreword is inadequate for an international standard.

MAS said that the French Group were waiting until the end of the current WG5 meeting to decide what to do about translation. Translation would take about a year, but this would not be a major obstacle to voting within SC22.

There was discussion between the Chair and **HIRCHERT** regarding the status of the SC22 TAG in relationship to X3.

MEEK noted that the existing S8 layout did not conform to ISO drafting standards, thus it is likely that ISO will prepare a one page standard which simply references the ANSI Standard.

10. ISSUES ARISING FROM THE WG5 BALLOT COMMENTS

(These issues were not necessarily discussed in the order presented here.)

10.1 Pointers (Carl Burch)

Burch referred to paper 21, page 243 of the upcoming X3J3 premeeting distribution. Burch presented the contents of that paper.

SCHONFELDER: The proposal is still not a completely general pointer facility. It doesn't distinguish a pointer from the object that it points to. I dislike having both keywords (ALIAS and ALLOCATABLE) - we need only one of them.

MATHENY: I question whether a pointer facility that doesn't say "pointer" anywhere will sell to people who must have pointers.

MOSS: I think it will.

JOHNSON: The facility is what a Fortran programmer would expect to find.

SMITH: What do you use for actual and dummy arguments?

MOSS: The advantage is that it doesn't look pointer-like. My users like things to be simple.

HOPPER: I want to see procedure variables referred to by pointers.

HIRCHERT: The framework described could support procedure variables. The rules needed are rules for ALLOCATABLE attributes.

SCHONFELDER: There is no pointer to pointer facility in the proposal as it stands.

MARTIN: Putting ALLOCATABLE and ALIAS together is a good idea. I like "VIRTUAL".

SCHONFELDER: The dereferencing problem becomes critical.

ADAMS: Has any non-X3J3 member a view?

SCHMITT: Against pointers in F8X.

MOSS: Right way to approach a problem is to try to implement it without altering the language.

TER HAAR: Apologise for lack of knowledge of subject area. Understand that F77 does not have certain functionality. Community wants those facilities. Do we need to give facilities to Fortran community to solve this sort of problem. If so, how do we do it? Pointers introduce dangerous things. If needed have to accept the danger in use. Would vote yes if needed.

MEEK: The difficulty of storing records with pointers within so they can be retrieved later is not new. Nothing inherently dangerous about pointer if used sensibly (no worse than the dangers of unconditional GOTO).

MOSS: Pointers will not degrade performance with an optimising compiler if there are rules for the use of pointers.

MUNCHHAUSEN: If we want pointers, they should be restricted to certain objects. Like the way pointers are implemented in C.

MAS: In favour of Munchhausen view of pointers.

MILINAZZO: Proposal goes a long way very cheaply. In favour of it.

MOSS: Can do the Burch proposal by a very small change to current draft.

HENDRICKSON: Doesn't think the concept is an easy thing to add to the language.

MOSS: Meant that it looks the easiest way to proceed not that pointers were easier.

WARREN: Seems to provide functionality at little cost. Schmitt will produce a resolution.

10.2 Significant Blanks (Andy Johnson)

The proposal for significant blanks affects only the new free-form source.

(1) The syntax of the language does not require significant blanks

(2) Many vendors do not require significant blanks even in free-form source.

TER HAAR commented that S8 should not advocate the use of non-significant blanks (e.g. in labels, page 3-4 line 21 of S8).

A straw vote on the proposal that a resolution be prepared on significant blanks was 10-12-5. As the vote was close, Schmitt would still prepare a resolution.

10.3 Multibyte Characters (Jim Matheny)

KANJI problem

It is very desirable that new capabilities be within the existing extension mechanisms in 8X. When Matheny started to write a Kanji module, he felt that no further I/O was needed. The problem can be solved with a derived type module without impacting the people that don't need it. An intrinsic module of derived type can be made available as an Appendix to the Standard or published elsewhere. A module is not necessarily inefficient.

Matheny has prepared a document for the X3J3 pre-meeting distribution (see X3J3/212, page 259).

SCHMITT: How do you handle a substring?

MATHENY: I haven't tried to solve that problem.

SCHMITT: What about the source form?

MATHENY: I'm not yet sure how to handle the escape sequences.

SCHMITT: In your model, 5 one-byte characters and 5 multibyte characters do not have the same length. A-format is not suitable for such strings.

MATHENY: The output routine can get the length of any concatenated string. This is not a problem on output but might be a problem on input.

MOSS: On output, you don't want type CHARACTER, but type KANJI. Matheny has not answered all of his own questions.

HIRCHERT: Escape sequences are not going to keep tabs, etc. aligned.

BURCH: There is a requirement to allow an extended, but limited, mixture. I cannot see how tab left with a Kanji character will ever work.

ADAMS: Detailed discussion of the technical issues should be deferred until the X3J3 meeting.

HOPPER: How will ICHAR work?

HIRCHERT: We can't decide what we are doing unless we understand the technical alternatives. Either we make Fortran processors work completely or we provide a complete module solution. The Japanese want their programmers to be able to freely use their own character set. I wonder if a processor option would solve the problem.

INA: A comment on the Chinese paper (N251). The whole problem has been discussed about 10 years ago. The Chinese paper leaves the user to take care of the escape sequences and does not solve the substring problem.

We need a facility that will be useful for old Egyptian and old Indian characters - in fact any characters.

10.3.1 Presentation on Kanji (Akira Ohwada)

The Japanese have tabled a paper on Kanji (Document N238). About one billion people in the world do speak non-English. Fortran needs to make it easier for such people. There are many other nations as well as Japan who would benefit. If the S8 draft does not accept multibyte characters, Japan is against the draft. The Japanese delegation would like WG5 to recommend the feature to X3J3.

The Japanese delegation gave a demonstration of the use of Kanji in Fortran on a PC.

WEAVER: There is no doubt that the Japanese need a mixture of Latin and Kanji characters. Kanji takes 2 character positions on a line.

INA: For reasons of efficiency, the current Japenese processors do not handle mixtures: a character is either Kanji or EBCDIC for reasons of efficiency.

MOSS: Thanked the Japanese for the examples which had clarified the issue for him. Noted that KIND=2 means a second type of character not a 2-byte character.

INA: A set of character codes could be given to a compiler as an option. Currently, each character in the NCHARACTER implementation takes two places on output. In the future, possibly each Kanji character would take only one place.

SCHMITT: Has seen two solutions to the problem. The Japanese soltion requires a change to the language. Noted that the use of Aw with the KIND= implementation causes problems.

HIRCHERT: I suggest that we have been characterising the problem backwards. What we really have is one very large character set then we take subsets of this major character set. The user should not need to be aware of the use of any escape sequences, etc.

WAGENER: One-byte/multi-byte emphasis is practical but does not necessarily capture the strategic issues. If we implemented Fortran

characters as a 32 bit word, would this problem arise? Is Kanji a special case of integral signs, square root signs etc? Does the use of KIND= trigger a data type that is not type CHARACTER?

SCHMITT: Typesetting facilities and the need for a large character set are not the same. Typesetting requires that each character has attributes.

MOSS: You shouldn't map all possible characters into one huge character set. How do you collate italic letter A and roman letter A?

HENDRICKSON: In some abstract sense characters are characters but, practically, switching to bigger characters increases storage requirements and reduces performance.

ADAMS: I have a note from Rich Ragan saying that CDC strongly supports extensions to Fortran 8X that support Kanji.

VALLANCE: We once implemented F77 on a 24 bit word/ 6 bit character machine by using only 8 bits of a word for each ASCII character.

POLLICINI: Will character size change as KIND= changes the character set?

HIRCHERT: KIND= does not cause a different data type. Whether we allow assignment or conversion depends on whether we define appropriate conversions. Note in passing the possibility of KIND=* to declare character dummy arguments. Note also that there is nothing in 8X that defines a relationship between how characters are held internally and how they are held on a disc.

MEEK: I do not like the KIND=2 concept as you can run the same program on 2 systems and get 2 different sets of results.

I am also concerned about the problem that KIND=2 means different things on different machines. There are registration mechanisms for character sets. There is a committee in TC97. I would recommmend that any WG5/X3J3 proposal should not conflict with anything done in that committee.

The reference to using an option to invoke the Kanji mechanism makes me think that the standards committee cannot avoid the concept of options in standards for ever.

Has anyone considered putting the Kanji proposal forward not as part of the 8X standard but as an incremental standard (not a module)?

It is unwise and probably counterproductive if X3J3 tried to look at this problem only in relation to Fortran.

INA: I will summarize the Japanese proposal. Within our proposal you cannot concatenate different KIND=; you can only concatenate within the same KIND. The operation of concatention is the same. You don't need to change any specification and can still use the Aw formatting for I/O. The compiler can distinguish between different KINDs. By using KIND, everything else in the standard is the same.

Matheny's simple module example was not complete or perfect but the Japanese feel that this approach could not properly support Kanji. How is a substring handled?

Demand for different kinds of character sets will increase in the coming decades. China will eventually require its own character set on a computer.

WEAVER: What forums exist to address Kanji as a cross language problem? (remark addressed to Meek)

MEEK: I was saying that you don't solve the problem in isolation but you see what other languages are doing. We will have to rely on the Japanese delegates to other language working groups. The Forum is SC22. Each national body will have to give consideration to the general problem. Something should be done in the Fortran arena as soon as possible. Kanji could be raised at the SC22 meeting in September.

SCHMITT: We need to address the problem at the Fortran level and also get the issue tabled at SC22.

HIRCHERT: We need to pursue both options for implementation. There is no way that we can predict all characters that will be used on a Fortran processor.

POLLICINI: I remind the meeting of Halifax resolution 1b.

A straw vote on a resolution to investigate Kanji was 32-0-2.

MEEK: Would the Japanese delegation consider an incremental standard to be acceptable?

SCHMITT: We want a resolution to ask SC22 to take the issue to TC97.

10.4 Exception Handling (Kurt Hirchert)

Standard Fortran is only defined when things work well. X3J3 looked at adding a standardized error handling facility in the language. Initially, this resulted in "event handling". There was concern in X3J3 at the cost of implementing this, particularly as the proposal allowed a program to continue (optionally) in the event of a failure. A second approach was proposed ("condition enable") which was simpler and easier to implement. This proposal was not as thoroughly integrated with the rest of the language as it might have been because of time constraints. There was "nagging doubt" on the part of X3J3 that the proposal was not technically sound. The proposal was moved to Appendix F of S8.

MOSS felt that the actual responses in the ballot were not the real reasons.

MUNCHHAUSEN asked if there were any plans to support multitasking in the model. **HIRCHERT** noted that the design was for error handling and not to aid synchronisation.

SMITH had presented the proposal to two members of the IFIP 2.5 committee who had not understood the proposal until he explained the underlying models. Both then thought it was an excellent framework to assist numerical calculation and both wanted the ability to substitute a default value if something went wrong during a calculation.

There was further discussion (**MEEK, SCHMITT, HIRCHERT, VALLANCE**) followed by a straw vote on whether there should be a resolution to put exception handling back into S8. The result of this was 10-15-6.

10.5 Bit Data Type (Lawrie Schonfelder)

Bit data type is in Appendix F. Anyone looking at F8X sees masked arrays with logical as in F77 but not bit data. The bit proposal was really a one-bit logical supported by a small number of instructions. Other proposals have wanted bit-strings. Such bit-strings have a different application.

HOPPER: Desirable especially for the array processing associated with image processing.

MEEK: You need multi-dimensional bit arrays for crystallography.

MOSS: Main problem is with the format of bit strings. The best choice is logical bits.

HIRCHERT: Apple brought out Mac 2 which uses bit strings to describe the screen.

MAS: Had written signal processing program both in Fortran and PL/1. Very efficient in Fortran using extensions but harder to program. Easier to use bit strings in PL/1.

ADAMS: If bit goes back in the language is the language too large?

MOSS: Wait for public comment.

MEEK: Should be reconsidered during public comment period.

HENDRICKSON: Easy to overestimate how useful bit date type is especially when concerned with 2-d arrays.

TER HAAR: Important to have bit type.

WEAVER: In letter ballot, 6 of 7 no votes mentioned missing bit data type. There is evidence from implementation of bit intrinsics by vendor that bit data type is wanted.

HOPPER: Need to be able to write and read back.

Schmitt will prepare a resolution.

Straw Vote	Logical bits	16
	Bit string	3
	Undecided	16

WAGENER: Maybe bit doesn't do what it is intended to do. It has a lot of new operators. Perhaps it is better as a parameterised logical. Maybe the Appendix F version is ill-advised.

10.6 Vector Valued Subscripts (Dick Hendrickson)

Vector valued subscripts had been in S8 for a long time. They were removed 18 months ago. The main problems were the different rules regarding many-to-one mapping that applied depending whether an assignment was involved or not and the implied gathering when a vector valued subscripted array element was used as an argument.

No straw vote was taken.

10.7 Deprecated Features (Andy Johnson, Carl Burch)

Appendix B of S8 lists all these features.

(1) There are no deleted 77 features.

(2) Obsolescent features are those features that X3J3 recommends should be removed at the next revision of Fortran.

(3) Deprecated Features carry the recommendation that they are removed two revisions later. This introduces the concept of language evolution. To date, there is no such mechanism. It is inevitable that mistakes are made and that things will need to be removed sooner or later. However, even allowing for the possibility of removing things from the language is politically fraught.

    Most Code presented to 8X compilers will be 77 code in transition. Appendix B tries to deal with anticipated public comments.

MUXWORTHY: Deprecated features should not be mentioned as such in the body of the Standard. You should identify items that might be deleted next time round.

MARSHALL: Feels that there are a lot of dinosaurs in Fortran. Language is becoming a graveyard of archaic statements. If we don't mark something as deprecated, the next committee cannot do anything. I would like to see a single list with a few more features.

WARREN: Agrees with Marshall - deprecated features should be identified in the text.

JOHNSON: Mandate had come from one of the parallel committees to X3J3 that decremental features should be identified in the way of Appendix B.

VALLANCE: Bit-users determine what is obsolescent.

TER HAAR: In Shell, we would have an edict not to use certain features.

HIRCHERT: Important distinction between obsolescent and deprecated was

that obsolescent features could be got rid of now in 77.

MOSS: Realised that assigned GOTO in obsolescent has internal procedure as replacement.

BURCH: Need to change.

ADAMS: May need editorial change.

WAGENER: Tremendous controversy in X3J3. Better to err on the side of going slow. Would personally prefer one list, but the mechanism in Appendix B allows us to go slow.

HIRCHERT: Possible to replace ASSIGN/ASSIGNED GOTO with safer F77 features.

MOSS: I don't think that computed GOTO can replace use of assigned GOTO to simulate internal procedures.

Straw vote on whether to retain resolution Halifax/13 on one list of deprecated features. (6-13-11).

Further discussion. Noted that X3J3 cannot bind its successor. Some feeling (Moss, Muxworthy) that shouldn't have deprecated features

Straw vote	Like S8 description	13
	Live with S8	15
	Dislike S8 description	2

10.8 Syntax Charts (Miles Ellis)

Something other than words is needed to describe the language. The current Fortran 77 Standard has "railroad charts" which seem to be popular. Miles Ellis had looked at the problem of generating railroad charts from the syntax productions. After discussion (**BURCH, ELLIS, MEEK**) there was no straw vote.

10.9 Presentation and Readability of the Document (Jerry Wagener)

The following is a summary of the style and readability comments from the WG5 ballots:

- Too much duplicate text
- Text too verbose, tutorial, tautological
- Always have an introductory paragraph at the start of a section
- Include a new appendix with railroad charts
  Use railroad instead of BNF
  Use ISO standard form of BNF
- Forward references are too numerous and are poorly handled
- No feeling of consistency or top down design
- Section notes should be integrated with the text
- Should have a one page appendix on the history of Fortran
- Section 2 is neither a glossary nor an overview
- distinction between program unit, subprogram, procedure

is confusing (changes made since)
- need standard organization for each concept (general, syntax, examples, semantics)
- the Standard "feels" too big

X3J3 had attempted to improve S8 in the following ways:

- many more examples had been added
- very many minor editorial enhancements had been made
- Section 4 introduction had been completely rewritten
- Overview had been added to foreword
- General organization was now considered good
- noted that the index was neither complete nor adequate
- a glossary had been added
- the BNF terms in Appendix D were cross referenced

There was discussion on the size of the line numbers at the side of the text and on the quality of the index. It was important not to overload volunteers with the work of manually preparing a completely comprehensive index.

HIRCHERT suggested that the text of S8 might be more readable if a Times Roman or similar typeface were used.

It was agreed that a resolution would be prepared which appreciated the work done by X3J3 members in the area of improving the readability of the document.

10.10 Passed-on Precision (Karl-Heinz Rotthauser)

Delegates were referred to document N245, the proposed changes for user-overloading to replace passed-on precision on page 2.

MUNCHHAUSEN: Is an interface block necessary for a (*,*) argument.

BURCH: Yes

SMITH: Implementers would not want to adopt the German solution because of the potential need to copy in/copy out where array precisions do not match. Performance problem.

There are problems with the concept of a standard-conforming processor. The German method is OK on IBM but not on CDC (say).

SCHONFELDER: For a very large proportion of numerical algorithms, you can write a generic procedure. As long as you satisfy the necessary condition that actual and dummy argument precision match. In order to get sufficient condition, the remainder of the (*,*) definition is required so that programs are both portable and standard conforming across many processors. The problem was looked at in great detail and proposals like the DIN proposal were rejected.

HIRCHERT: Model of how you could allow the compiler to remove 'clear' code for a particular precision.

Some features in standard are really intended for IMSL, NAG etc. who prepare tools for other average programmers.

Typically the professional programmer will not use (*,*)

It allows overloading only amongst procedures that are nested (e.g., in a module). External procedures must all have unique names.

MUNCHHAUSEN/MAS: Discussion on the establishment of an interface for a procedure. The outcome - the calling procedure either has an interface block or the procedure can be in a module or be an internal procedure.

MUNCHHAUSEN: Would like editorial clarification of the interface problem.

SCHONFELDER: Can define implementation model that doesn't need interface block. No need for dynamic behaviour. Can implement by a locational replacement pre-processor.

SMITH: S8 defines the interface problem.

MUNCHHAUSEN: Now satisfied.

HIRCHERT: There may be things that can be done to clarify.

Straw vote on whether a resolution that X3J3 should reconsider passed on precision and ensure that it is compatible with F77 (3-13-15).

MOSS: Some WG5 people should look carefully at the (*,*) proposals as X3J3 believe that it is now correct.

BURCH: C-16 lines 18-26 of the Section Notes describes how the passed on precision ought to work.

ADAMS: I will make a proposal to X3J3 that notes should be interspersed with text with an indication that notes are not part of the standard maybe a resolution.

11. STATUS OF HALIFAX RESOLUTIONS

The drafting committee had prepared an initial draft of resolutions. Delegates were asked to consider which resolutions should be brought forward from the Halifax resolutions. **SCHMITT** wondered whether it would be necessary to word some resolutions so that two votes were possible: (a) on the technical content of the resolution and (b) on whether the resolution proposed something that might delay the Standard. The committee wanted to carry forward Halifax resolutions 11, 13, 14, 20, 22, and 23.

MEEK suggested that some if not all of the Halifax resolutions brought forward should be reworded in the form: "WG5 urges X3J3 to reconsider ... when public comments are being processed".

11.1 Operator Renaming

There was a brief discussion on the technical issues raised by this proposal (**MOSS, HIRCHERT, WAGENER**). A straw vote was taken on whether a resolution on the inclusion of operator renaming was necessary (1-24-10).

11.2 Name-directed I/O

X3J3 had spent a long time on this issue. X3J3 felt that the syntax/semantics of name-directed I/O was better, but that existing practice had NAMELIST.

BURCH: Current practice varied so name-directed I/O was invented. Over the years, different vendors' ideas had converged so giving a de facto standard for NAMELIST.

MEEK: Concerned about the "minor variations" amongst implementations. The original reason for WG5 not wanting NAMELIST was these minor variations.

HIRCHERT: The NAMELIST defined in S8 does not correspond exactly to anyone's existing practice. Vendors could support both the standard and their own variant. NAMELIST provides a major feature (the list) which is not available in name-directed I/O.

MATHENY: The NAMELIST statement does not restrict a list to one statement.

SCHMITT: NAMELIST is not consistent with the syntax of other I/O components.

HIRCHERT: To some, the confusion already exists: e.g., in H-format and quoted strings in format where part of the format appears as output.

BURCH: Agree with Meek that current practice is not the only reason for adopting a construct into a standard language.

A straw vote on whether there should be a resolution in favour of name-directed I/O was taken (11-15-8)

11.3 Procedure Interfaces (Bonn Resolution 24)

WAGENER: X3J3 haven't seen a way to do it. It would be a nice feature but it is not possible. Any resolution should suggest a way to do it.

HIRCHERT: The X3J3 subgroup could not find a way.

MOSS: In order to use the facility, you would need access to the source code. If you had the source code, you can use a module then there is no need to use a procedure interface block.

TER HAAR: The interface block is a good idea if the same block can be used both in defining and referencing the procedure. You could, of

course, use the unacceptable INCLUDE!

HIRCHERT: I can envision reasons why a procedure would be external rather than in a module. The disadvantage of the textual include is that identical text may have non-identical meanings.

ADAMS: Is there any non-X3J3 group that will provide a proposal?

MUNCHHAUSEN: The problem has been solved in C.

JOHNSON: INCLUDE will redefine an object. You can have a procedure interface block followed by an internal procedure that exhibits the same properties. C has an equivalent to the procedure interface block.

SMITH: Misunderstands what has been said. Has a simpler model. The routine being called is either in a Fortran program or it is not. If it is Fortran, then it can be enclosed in MODULE/END MODULE statements. If it is not Fortran, then the user must specify the procedure interface block.

MUNCHHAUSEN: The problem is that you specify the procedure interface twice; this is error prone.

SCHONFELDER: You could do all that Smith says. On the other hand, Fortran allows you to define the interface twice.

MOSS: You can define the interface if you are in a support environment then alter the procedure body. The environment will then say what has to be recompiled if anything changes.

ADAMS: Any resolution must be complete in technical detail.

TER HAAR: It may be that X3J3 has to clarify the current S8 document.

SCHMITT: I suggest that no straw vote be taken but that anyone who is interested should attempt to draft some wording for a resolution.

11.4 Use of [] Characters

SCHMITT: On a system without the ASCII character set, say DIN or French, you do not have the [and] characters. Thus you have to use the alternate forms (/ and /) for programs because you cannot write a standard conforming processor otherwise. Either X3J3 or WG5 has to do something. We need to word an appropriate resolution.

SCHONFELDER: How do you cope with Algol in the same circumstances?

SCHMITT: There are alternate forms for Algol symbols in systems with limited character sets. In the past, people normally used ASCII. There is now an increasing use of own language character sets. The next step is to change from a 7- to an 8-bit code.

ADAMS: Wishes it were possible for 8X to go to an 8-bit code.

HIRCHERT: [ ] could be treated in the same way as lower case so, quite

naturally, [goes to (/ and] to /). Then (/ and /) could be used in the primary definition.

SMITH: I have a problem with the statement "you cannot write a standard-conforming processor"

SCHMITT: Refer to p3-2 of S8.

SMITH: The alternatives (/ and /) are provided for portability.

MEEK: Discussion shows the problems caused by a number of things, most importantly ignoring the existence of other relevant standards. Recommends that whoever looks at this problem should please read the guidelines technical report which is currently being updated. Belives it possible to alter the wording in S8.

HIRCHERT: I disagree with Smith on what the portable programmer has to do. I might use [followed by blank and] preceded by blank then use a global edit to change to (/ and /) if necessary later.

WEAVER: [and] were introduced to avoid a single ambiguity viz A = [1,2] where [1,2] is <u>not</u> a complex constant.

It seems amazing that there is all this confusion/discussion for one use of [ ]. Perhaps S8 should specify the use of ((and)) for this one ambiguous case.

SCHMITT: The drafting committee will prepare a resolution asking X3J3 to resolve the problem.

11.5 S8 Review Period

SCHMITT: Felt that he and his National Group would restate any differences of opinion with X3J3 during the review period. We should look for ways to support WG5 convenor efficiently to respond to X3J3.

ADAMS: It is very convenient for X3J3 subgroup heads to have some collection of comments from WG5 members via the WG5 convenor. Is it reasonable for WG5 to define a collating procedure.

MEEK: WG5's job is to liaise with X3J3 to ensure that the final US Standard is acceptable to the International Community. It is the job of WG5 members to inform members in own country what (and why) things are in the standard. If there is any thought that there might be a national NO vote, members should immediately inform the WG5 convenor. The convenor could then circulate objections to other members of WG5.

POLLICINI: You should also send any communication to each national head of delegation.

MEEK: It is important to establish a mechanism that delivers a YES vote. It is not our job to "sell" the white book. We need to inform why it is the way it is and why other countries feel the way they do.

BURCH: What does WG5 want taken out of S8?

TER HAAR: ALIAS/RANGE/IDENTIFY.

MILINAZZO: I'm not sure what SET RANGE does.

ADAMS: It limits size of array during a calculation.

MILINAZZO: I wouldn't probably use it. I would prefer to use sections. I might forget that SET RANGE had been used.

HENDRICKSON: It is a notational convenience and also guarantees conformance at compile time.

SMITH: Every element has to be same shape so every array reference has a section qualifier. If one SET RANGE is used, then the remainder of the code is easier to read.

WEAVER: Many problems have array codes where element neighbours are of interest which means there is a boundary problem. SET RANGE is a better way to represent the problem.

POLLICINI: Don't like merging of old and new DO.

MOSS: Could you use EXIT and CYCLE in the old DO? Could you nest new and old DO?

BURCH: It is a problem of maintenance of existing code. If both forms of the DO are not the same statement, you would have a significant list of illegal statement uses, so old code would not run when compiled with 8X compilers.

TER HAAR: (1) I want to remove object-oriented declaration . (2) Also remove the character declaration that uses LEN= (3) I would finally remove CASE with LOGICAL type variables. A block-IF is a better alternative.

SCHONFELDER: (1) LEN= is included for internal consistency with REAL declarations. (2) Ability to be able to declare attributes was for readability of the code. Current scattered method is bad practice. There is no different functionality.

HIRCHERT: There are two instances of SELECT CASE:

(1) Statement.
(2) Part of the variant structure declaration.

No alternative such as block-IF is available in the (2) instance so we preferred to keep both the same.

12. COPYRIGHT (AGENDA ITEM 17.5 - ANDY JOHNSON)

Johnson had spoken to CBEMA's Gwendy Phillips on August 4, 1987. The response was, that as long as the copies were not publicized, not sold, and used only for the purpose of review then they could be copied without restriction.

MEEK recommended that the cover page of public review material should say where comments should be sent to.

WEAVER suggested that a resolution should be drafted to contain the wording of the CBEMA comment on copying.

The latter suggestion was not adopted but Andy Johnson was asked to clarify the CBEMA position with regard to copying of S8 (see N253).

13. DISCUSSION ON DRAFT LIVERPOOL RESOLUTIONS

A first draft of re-affirmed and new resolutions was distributed without a number as the document was not yet complete.

13.1 DR3: Content of the Resolution Response Document

There was discussion on draft resolution DR3 which asked X3J3 to give reasons if a WG5 resolution is rejected.

13.2 DR10: Program Size and Complexity

There was discussion on the problems of reporting program size and complexity limits (**WEAVER, SCHMITT, VALLANCE, HIRCHERT, HENDRICKSON**). **MEEK** discussed reasons for the BSI Standard for specifying Fortran processors. This Standard is both for users of Fortran for implementation and for those purchasing an implementation. The word 'limits' frightens people. There are lower and upper limits. Perhaps standards should specify lower limits. Users would like no upper limit on program size and complexity.

WEAVER: What about what is commonly called overflow? This is not a static check.

MATHENY: A vendor might lose business because of a poor choice of typical values if such values were specified in a user document.

ROTTHAUSER: Would like to restrict the limitation to reporting to the compiler rather than the run time system.

TER HAAR claimed that as a user he could deceive every compiler.

The result of a straw vote on whether the changed resolution should be adopted was 18-7-7.

13.3 DR11: Envelope for resolution 4 and 5 brought forward from Halifax

There was discussion as to whether or not it was relevant or valid to note that any resolution was a reaffirmation of a Halifax resolution. Danger that one resolution would appear to have more importance than another. Meek apologised if his proposed wording implied criticism of X3J3.