<p style="text-align:center">Minutes of DPCE Meeting
23-25 March 1994</p>

Sheraton Inn
5115 Hopyard Road
Pleasanton, CA 94588

Attendees

Frank Farance, Jamie Frankel, Phil Hatcher, Joe Peters, Linda Stanberry

Farance is chairman, Frankel is vice chairman. Stanberry served as acting meeting chair, sergeant-at-arms, and secretary.

Meeting Goals

The overall goal was to approve changes to the document to reflect decisions made, to complete as much as possible of the remaining document, and to devise a plan for completing other parts before the 2nd mailing for J11. The issues raised at the Kona meeting were addressed, including editorial suggestions received from the committee as a whole, and those provided separately to the technical editor from Rex Jaeschke.

There was discussion of exactly what the mandate was regarding separation of context and layout from shape as a result of the Kona meeting. Frankel believes it was recommended that we investigate alternatives, but that there was no requirement for us to actually force a separation of these concepts. Farance will present his proposal for using APL-like definitions of these concepts which will include their separation; he wants feedback on his proposal, and plans to pursue this approach individually if it is rejected by the DPCE committee, just as CRI is doing with their iterator proposal. Stanberry expressed concern that all such proposals should remain compatible. Hatcher wants the subgroup to make a final decision on the separation of concepts and then to proceed without revisiting the decision at each meeting; rationale should be added to the proposal to explain the committee's decision. Frankel fears that Thinking Machines might lose interest in DPCE if the decision were made to separate the concepts.

There was some discussion of whether or not to include "slicing" with the DPCE proposal. Farance has an action item to develop proposal extensions to the DPCE proposal for slicing. Hatcher asked if the extensions would cover arrays as well as parallel objects. Farance indicated that would be determined by the committee's decision on his APL extensions proposal.

There was some discussion about the features that exist in Fortran 90 and HPF. It was noted that HPF allows separable, dynamic determination of layout. That is, HPF does not require that operands to operations have the same layout, so long as the data is "conformable." The HPF where statement does not support dynamic contextualization, however, and limits what kind of procedures can be called from it; context is not passed into procedures except for generic functions.

APL/VLA proposal

Farance presented a "Monday" paper detailing a set of extensions for C that would use APL-like concepts of shape and rank to define attributes of array types. His proposal includes layout as another attribute for array objects that are distributed (discontiguous

arrays or ALO's). His proposal also includes a VLA extension, using the fat pointers approach as in the Ritchie/Prosser/Meissner VLA proposal. Finally, his proposal would define context in terms of parallel control flow rather than as an attribute of the data. He believes this set of extensions preserves the concepts and functionality of C*; that C* is an optimized implementation of this proposal.

The committee had many questions and concerns with the proposal. The consensus was that there were insufficient details presented on how to fold these changes into the base document: without these details, it was believed that the effect of such changes were basically unknown and not thought through. The proposal was incomplete as presented, and had inconsistencies as evidenced by the numerous questions. Specific objections/concerns over the proposal included:

(1) The scope of the proposed change is not restricted to ALO's or parallel objects. It is essentially a proposed change to treat arrays as first class objects: promotions and operations are to be defined for arrays as well as ALO's. This is beyond the scope of the current DPCE proposal.

(2) The inclusion of the VLA extension is also beyond the scope of the current DPCE proposal. These proposals should remain separable.

(3) The whole issue of mixing pointers to C arrays and pointers to ALO's is glossed over by the proposal. There are very serious problems that arise in trying to define this type of interoperability.

(4) The use of type qualifiers to annotate pointer behavior is expected to have problems being accepted by the C community. The placement of these qualifiers was an unresolved issue.

(5) The layout "attribute" was incompletely specified; the CRI proposed syntax was preferred. The issue of what the layout is if it is not specified was unresolved. The issue of how layout is determined for arguments (if not passed as a part of the argument) was unresolved.

(6) Context was (admittedly) not addressed by the proposal. The intent is that it will be defined by parallel control flow, but it is unclear what this entails.

Farance believes that this proposal simplifies the DPCE proposal, but the rest of the committee believes that it is too incomplete to simplify the proposal, and would actually end up making the proposal more complex if the details were all worked out. Farance claims that the advantage of his proposal is that it separates concepts, and allows arrays and ALO's to interact, requiring only shapes but not layouts to be conformable.

The committee provided Farance with numerous suggestions on possible solutions to some of the issues identified. Farance will continue working on the proposal and will try to address these issues in a revised proposal. He plans on presenting the proposal at the next J11 meeting. Stanberry stated a preference that it be presented as part of a "technical session" on DPCE issues rather than to the full committee. There was some discussion on whether there would now be two DPCE proposals. The status of the iterator proposal was discussed--Frankel believes it has not been implemented on a parallel machine. Frankel further stated that he believes standards groups should codify proven practices, not propose research items.

Layout and context

The committee continued a general discussion of separating layout and/or context from the current DPCE notion of shape. Performance is the real issue that the committee would like to address in considering the advantages and disadvantages of keeping these concepts bundled together as at present.

The rationale for keeping layout as part of shape is the transparency of operations.

The rationale for keeping context as part of shape is that it is always/only used together with shape, and it is reasonable therefore to keep them together. Farance argued that contextualization only occurs in parallel-if and where statements. Hatcher pointed out that contextualization occurs in all statements: all positions active is also a contextualization.

The committee then attempted to itemize pro's and con's for the following proposals, and voted on each proposal as noted.

Proposal: separate layout from the concept of shape

    Pro's

        Conceptually cleaner, elegance of expressability
        Allows operations on objects of different layouts (Note: this can be done
            with [.]object explicit references in C*)
        No significant performance cost on non-distributed memory

    Con's

        Disallows optimizations knowing objects are the same layout--significant
            performance hit
        Transparency of performance lost
        Must be coupled conceptually on distributed memory architectures
            anyway

    In favor: 1
    Opposed: 3
    Abstained: 1

Proposal: separate context from shape

    Pro's

        Conceptually separate: storage versus execution time concepts

    Con's

        Artificially separates concepts always used together
        Complicates explanation of what is affected by context

    In favor: 1
    Opposed: 4
    Abstained: 0

<u>Technical editing</u>

Numerous issues with the proposal document were addressed and will be reflected in the next version of the document, including:

> Specification of compatible shapes, parallel types, and composite types (3.1.2.6).
> Specification of same shape using structural equivalence (3.2.3).
> Completion of contextualization specification (3.2.3 and 3.6.7).
> Array subscripting by parallel int (3.3.2.1).
> Specification of indexing an array of parallel; & and * operators (3.3.3.2).
> Parallel indexing (3.3.3.5).
> Constraints and semantics for pointer arithmetic (3.3.6).
> Specification of shape assignment as copying, not aliasing; and specification of shape assignment constraints (3.3.16).
> Constraints on expressions used in shape type specifiers, and clarifications of layout specification (3.5.2.4).
> Require shape-expression in everywhere statement to evaluate to a pointer to a shape (3.6.7.2).
> Removed intrinsic functions--define as general utilities instead (4.1.7).
> <dpce.h> will declare "overloaded" functions as elemental (4.5, 4.10, 4.11).
> Add salloc() and sfree() to replace allocate_shape(), allocate_detailed_shape(), and deallocate_shape(), returning/taking pointers to shapes (4.10).

Farance will investigate possible solutions for handling errno in the math elemental functions.

Stanberry will produce a cover letter for the revised document to explain what has changed, and to explain change bars in the document.

<u>Technical presentation</u>

Peters presented an overview of the work being done at David Sarnoff Research Center on a parallel dialect of C. His presentation included background information on the visualization project there. The secretary was awake during the presentation, but didn't take any notes and won't try to reconstruct the details from memory.

Peters indicated that they are very interested in coming up to speed on the DPCE proposed extensions, and in tracking the standardization process.

<u>Commitments and Schedules</u>

In order to complete the revision of the document for the second J11 mailing due 4/29, work was identified and parcelled out to committee members. It was agreed that this work must be completed and posted to the "editorial" DPCE mail reflector by 4/15. This will allow until 4/22 for discussion to occur on the reflector of everyone's proposed edits. The technical editor will fold in the changes and submit them to Plauger for the mailing.

Hatcher - will provide edits for including nodal functions

Frankel - will provide edits for including parallel pointer handles, and wording for "clarifying" the single thread programmer's model (2.1).

Farance - will provide edits for ALO slicing

Stanberry - will provide edits for the library, an outline for an Appendix for "All things considered but not included", an index, and the edits adopted at this meeting.

Peters - will review the document!

There was discussion of what should be presented at the next J11 meeting. It was agreed that the technical editor should present a summary of the changes to the document and of the decisions from this meeting. The consensus was that it would not be worthwhile to go through these changes or the document in detail before the whole committee. Rather, it was suggested that we request an evening "technical session" in which we discuss the details of the proposal with interested parties, and continue editorial work on the revised document from input received at the technical session.

Farance will request appropriate agenda time for the DPCE summary and for an evening technical session.

Stanberry will try to contact other DPCE editors for contributions to the document by the deadline.

Slicing

Farance presented some ideas on how to incorporate an array/ALO slicing extension. The problems identified with trying to do this include having to define a temporary shape (allocated, freed?) for the object that results from the slice. One approach is to put the burden of declaring and managing such shapes on the user. Another issue is whether multi-dimensional slicing is done using cross product or dot product. Currently, the dot product is used for parallel indexing in C*. Frankel suggested that alternatively slicing could either define a mapping of cross product to dot product, or use the equivalent dot product to get the same behavior "as if" the cross product was used.

Straw vote: Is cross product slicing desirable?

> 3 Yes
> 1 No
> 1 Abstain

Future Meetings

We selected 9/26-28/94 as the next DPCE meeting. The location is still to be determined, but will probably be on the east coast.

Farance will contact Maya Gokhale about hosting the September meeting.

Stanberry expressed concern over the progress of the group, which is not aided by the absenteeism during the meetings. If the commitments made for the document revision are not kept, she will not be able to justify to her management spending additional resources to attend meetings, and will have to resign as technical editor.

Farance will send email to encourage more participation in DPCE meetings.

Frankel will reply to Tom Plum's request for information on DPCE/C++ issues.