Subject: Further (ultimate?) generalization of rank remapping
From: Van Snyder

1 Introduction

Pointer rank remapping could be generalized beyond contiguous objects to any sequence of uniformly-spaced array elements. Rank-one arrays are uniformly spaced, but others would qualify.

2 Edits

Edits refer to 06-007r1. Page and line numbers are displayed in the margin. Absent other instructions, a page and line number or line number range implies all of the indicated text is to be replaced by associated text, while a page and line number followed by + (-) indicates that associated text is to be inserted after (before) the indicated line. Remarks are noted in the margin, or appear between [and] in the text.

[Editor: Add before the paragraph:] 164:22

An object is **uniform** if it is

   (1)   contiguous (5.3.6),

   (2)   of rank one,

   (3)   a dummy argument that is argument associated with an actual argument array that is uniform,

   (4)   a pointer that is associated with a uniform target, or

   (5)   an array section that

      (a)   has a uniform base object,

      (b)   does not have a vector subscript, and

      (c)   consists of elements of a subset of the base object whose array element-order (6.2.2.2) positions within the base object are uniformly spaced.

NOTE 7.47a

A scalar is uniform because it is contiguous. Assuming A has shape [4,4,4], A(1,:,2) is uniform because it is a rank one array; the following higher-rank sections are either uniform or not, as shown by the array-element order within A of the elements of the sections.

Uniform		Not uniform	
Section	Elements	Section	Elements
A(:,:,1)	1:16	A(1:2,:,1)	1, 2, 5, 6, 9, 10, 13 and 14
A(::2,:,1)	1:16:2	A(:,::2,1)	1:4, 9:11
A(4:1:-1,4:1:-1,4:1:-1)	64:1:-1	A(4:1:-1,1:2,1)	4:1:-1, 8:5:-1

[Then replace "contiguous ... one" by "uniform".]

[Editor: In the second line of the note, "either rank one or contiguous" ⇒ "uniform".] 165:Note 7.49