A RECHERCHE NUCLÉAIRE
IUCLEAR RESEARCH
que des Particules
ticle Physics

ALEPH 90-15
DATACQ 90-05
January 1990

Author: H.Seywerd

ent

nay be used to store data sets
at they will be included on the
is intended to be used to save
line, for use by the reconstruction

ie cons_delete_files. Provide
: consistency check. Changes to
e run number at a time.

Dependent Constants

ıe of the information necessary for the reconstruction of ALEPH data roduced while those data are being collected. This information may ıde calibration constants, error records, slow control, and summary rmation; we call these data run records. They may be produced from esses running on any of the online computers. A mechanism must rovided by which, at the end of a run, these data may collected in place for storage on the output tape. In addition, it is desirable that e data be available at the beginning of the data tape, in spite of possibility that they are only determined during or at the end of a This is only possible if both event data and these constants are first ten to disk, and only sent to tape after the run is complete complete.

ıny time during a run up to the point of "Close" run, run records may tored for collection, by calling the set of routines discussed below. eneral the calls should be made either in the "Config" state, on the ısition through the "End" state. These routines expect data in the ı of BOS banks, in an array. They are stored in temporary files, in directory ALEPH$RUN_RECORDS, with identifiers as to the run partition identification number for which they are relevant, and the letector which produced them. The temporary directory used for this pose should not be accessed except by these routines. As part of the of run procedure these temporary files are collected together, written ıe data tape, and made available to the Falcon processors.

S

records are stored by calls to the following functions:

cons_store_sor(detec, name, data, size, run_number)

cons_store_eor(detec, name, data, size, run_number)

guments are discussed in the Reference Manual section.

each call to the routine a file is produced in ALEPH$RUN_RECORDS ı the name: detec_name_nnnnnnnn_mmmmmmm.type, where:

c	subdetector name
e	sub-name as mentioned above
ınnnn	partition id number
nmmmm	run number

1.3 Recovering Run Records

Run records are recovered and placed in a final output file by calls to the following functions:

- cons_get_all_sor(cons_status, total_size)
- cons_get_all_eor(cons_status, total_size)

A call to cons_get_all_sor of these will collect all records stored with calls to the routine cons_store_sor, while a call to cons_get_all_eor will collect all of those stored with cons_store_eor. Both of these routines also produce, within the files, a 'RUNR' bank, to inform the offline that that the banks following are to be kept for the entire runs worth of processing.

1.3.1 Files Produced

When used within a subdetector partition a BOS format fixed block size file: RUN_nnnnnn.SOR or RUN_nnnnnnn.EOR is produced in the directory specified in cons_status.directory.

1.4 Deleting Run Records

At the end of a run sequence, run record files that are no longer needed should be cleaned up. The routine cons_delete_files performs this function.

- cons_delete_files(run_number, partid, num_files)

A call to cons_delete_files will delete all run records files belonging to the given run number and partition id. The number of files deleted is returned in num_files.

1.5 Restrictions

The storage routines should be used within a partition context, otherwise quantities determined from the PCT will not be correctly set.

1.6 Implementation

To link these routines on the VAX include, into you link file a line with:

- a_datawr$dir:cons.olb/lib.

Include files provided for as follows:

- a_datawr$src:cons_ssdef.inc Contains return codes.
- a_datawr$src:cons_status.inc Contains Structure definition for cons_get_all_xxx routines.

2 Examples

2.1 Bank Storage

The following is an example for an application in a task running on VAX and written in FORTRAN. We assume the partition id is '103'X.

```
C --- Declarations
      INTEGER istat, cons_store_sor
      INTEGER ibuf(50000)
      INTEGER run_number /23/

C --- Build a BOS bank in the array ibuf, first four words are BOS header
      ibuf(1)   = 'ABCD'
      ibuf(2)   = 0
      ibuf(3)   = 0
      ibuf(4)   = 3
      ibuf(5)   = 1
      ibuf(6)   = 2
      ibuf(7)   = 3
C --- And another one
      IBUF(8)    = 'EFGH'
      ibuf(9)    = 0
      ibuf(10)   = 0
      ibuf(11)   = 2
      ibuf(12)   = 4
      ibuf(13)   = 5

C Will produce a file in aleph$run_records with name
C SATR_TEST_00000103_0000023.SOR
C
      istat = cons_store_sor('SOR', 'TEST', ibuf, 13, run_number)
```

2.2 Retrieval

```
      INCLUDE 'A_DATAWR$SRC:CONS_STATUS.INC'

      RECORD /cons_status_info/cons_status

      INTEGER size

      INTEGER istat

      cons_status.directory  = 'TEST.SOR'
      cons_status.run_number = 5687
      cons_status.partid     = '103'X
      cons_status.part_det   = pct.part_det

      istat = cons_get_all_sor(cons_status, size)

C Produces a file containing all start of run type records.
```

Examples

2.3 Deletion

```
      INTEGER run_number
      INTEGER partid
      INTEGER num_files

      INTEGER istat

      run_number = 5687
      partid     = '103'X
      istat = cons_delete_files(run_number, partid, num_files)

C Deletes files, and returns number of files deleted
```

3 Reference Manual

x *detector, sub_name, data, size, run_number*

alue
d (unsigned)
ly
e

alue returned in your hardware register. Conditions urned are described under RETURN VALUES.

er string

ly
riptor
letector producing this record.

er string

ly
riptor
defined uniquely within each subdetector, should that more than one run record file of this type.

rd
ly
ence
a sequence of one or more consecutive bos banks to be

size

VMS Usage: integer
type: longword
access: read only
mechanism: by reference
The total size of all the banks and bank headers within data.

run_number

VMS Usage: integer
type: longword
access: read only
mechanism: by reference
The run number for which these data are valid. Be certain which PCT values you use for which file at which point in the run sequence.

DESCRIPTION

These routines store a set of bos banks in a file of the type xxx, where xxx is either sor or eor. Calls to these two routines are otherwise identical. The set of banks must be continuous in the array, and be in proper BOS format.

RETURN VALUES

SS$_NORMAL	Success
CONS_SS_BAD_BOS	error BOS format of data array is inconsistent.
Any other value	System error from attempt to write intermediate file.

cons_get_all_xxx

Assemble all run records for writing to the output tape.

FORMAT

cons_get_all_xxx *cons_status, size*

RETURNS

VMS Usage: **cond_value**
type: **longword (unsigned)**
access: **write only**
mechanism: **by value**

Longword condition value returned in your hardware register. Conditions value that can be returned are described under RETURN VALUES.

ARGUMENTS

cons_status

VMS Usage: **longword integer**
type: **structure**
access: **read only**
mechanism: **by reference**
Structure specifying information on the data to be retrieved and where to put it. The structure has the following format.

CHARACTER*40	directory	Directory where to write data
INTEGER*4	run_number	Run Number of the data
INTEGER*4	part_id	Partition id for this data
INTEGER*4	part_det	Detectors in Partition (pct.part_det, used in filling the RUNR bank.)

run_number

VMS Usage: **integer**
type: **longword**
access: **read only**
mechanism: **by reference**
The run number from which to collect run record information.

size

VMS Usage: **integer**
type: **longword**
access: **write only**
mechanism: **by reference**
The total number of words stored.

DESCRIPTION

The routine cons_get_all_sor collects all run records produced with cons_store_sor into the bos formatted fixed block length file RUN_nnnnnnn.SOR. The routine cons_get_all_eor collects all run records produced with cons_store_eor, and produces a file RUN_nnnnnnn.EOR. These routines are intended to be called from the tape/disk tasks.

RETURN VALUES

SS$_NORMAL	Success
Any other value	System failure codes on reading or writing the files.

cons_delete_files

Delete intermediate files.

FORMAT

cons_delete_files *run_number, partid, num_files*

RETURNS

VMS Usage: cond_value
type: longword (unsigned)
access: write only
mechanism: by value

Longword condition value returned in your hardware register. Conditions value that can be returned are described under RETURN VALUES.

ARGUMENTS

run_number

VMS Usage: integer
type: longword
access: read only
mechanism: by reference
The run number for which files are to be deleted.

partid

VMS Usage: integer
type: longword
access: read only
mechanism: by reference
The partition identifier for which files are to be deleted.

num_files

VMS Usage: integer
type: longword
access: write only
mechanism: by reference
The number of files deleted.

DESCRIPTION

This routine deletes the intermediate files in ALEPH$RUN_RECORDS, produced by the calls to cons_store_sor/eor, corresponding to the give run number and partition identifier. It is intended to be called from the tape/disk task.

RETURN VALUES

SS$_NORMAL	Success
Any other value	System error code associated with the attempted delete operation.