Stephen Tinsley's Blog: Export Anesthesia Records to a File

Posted by Stephen Tinsley Jan 28, 2016

Sometimes there's a need to export Anesthesia Records for billing, risk, printing, secure emailing, or quality purposes. There are many posts on asking about this functionality, but I've yet to see someone deliver on it or at least post a CCL solution.

There are solutions like clinicalXR that can export PDFs of the anesthesia record, and printing from Power note is a terrible pain. Thus there is a need to extract final records for the aforementioned purposes.

Here is my solution to this problem. The program below begins by calling the sa_get_anesthesia_records script to get a list of anesthesia records for a given date.

It then drops into a for loop and queries CE_blob to check for charts, and get blob lengths and such. It stores the data in a flat record structure called output. If the first query finds a blob, then it drops into the act of retrieving the record and outputting it to a file.

Anesthesia Records are TIFF documents stored in multiple rows on CE Blob. They are uncompressed. The script extracts the blobs, concatenates them and writes them to file in CCLUSERDIR (or a directory of your choosing). The script then loops back and does it again.

I modified the working with blobs document provided by the discern team to extract, concat() and strip the delimited data out of the blobs. Then I used CCLIO to write to a file.

The resulting tiff files can be sFTP'd off the backend for whoever needs them or converted to pdf by a 3rd party app. TIFF is an adobe standard so it should suffice with most billing companies.

I hope this assists the Anesthesia community.

-Steve

```
drop program  ANESREC_2_TIFF:DBA go
create program ANESREC_2_TIFF:DBA
/*
* @author stinsley - 1/14/2016
- purpose: query anesthesia records and output them to a TIFF file for billing purposes or later conversion.
- one tiff file can contain multiple pages and can later be converted to a .pdf by an external utility
*
* tables: Clinical_event
*  CE_BLOB
*  surgical_case
*  sa_anesthesia_record & child tables
```

```
*
* scripts used: SA_GET_ANETHESIA_RECORDS
*
*
*
*/
prompt
  "Output to File/Printer/MINE" = "MINE"   ;* Enter or select the printer or file name to send this report to.
  , "Event_start" = ""
  , "event_end" = ""



with OUTDEV, event_start, event_end









RECORD  CASES_REPLY  (
  1  ANESTHESIARECORDS [*]
  2  SA_ANESTHESIA_RECORD_ID  =  F8
  2  EVENT_ID  =  F8
  2  SURGICAL_CASE_ID  =  F8
  2  CREATED_BY_PRSNL_ID  =  F8
  2  CREATED_BY_PRSNL_NAME  =  VC
  2  CREATE_LOCATION_CD  =  F8
  2  CREATE_LOCATION_DISP  =  VC
  2  RECORD_DESCRIPTION  =  VC
  2  CREATE_DT_TM  =  DQ8
  2  LAST_DOC_DT_TM  =  DQ8
  2  LAST_DOC_PRSNL_ID  =  F8
  2  LAST_DOC_PRSNL_NAME  =  VC
  2  SA_REF_DOC_TYPE_ID  =  F8
  2  SA_REF_DOC_TYPE_NAME  =  VC
  2  SA_DOC_TYPE_CD  =  F8
  2  SA_DOC_TYPE_DISP  =  VC
  2  SA_DOC_TYPE_MEAN  =  VC
```

```
 2 FINALIZATION_TYPE_FLAG = I2
 2 ACTIVE_IND = I2
 2 UPDT_CNT = I4
 2 SUPERVISOR_REQ_IND = I2
 2 ANESTHESIARECORDLOCKS [*]
 3 SA_ANESTHESIA_RECORD_LOCK_ID = F8
 3 LOCKED_IND = I2
 3 LOCKED_BY_PRSNL_ID = F8
 3 LOCKED_BY_NAME_FULL_FORMATTED = VC
 3 LOCKED_BY_USERNAME = VC
 3 LOCKED_DT_TM = DQ8
 3 LOCKED_LOCATION_CD = F8
 3 LOCKED_LOCATION_DISP = VC
 3 UPDT_CNT = I4
 3 ACTIVE_IND = I2
 2 ANESTHESIARECORDSTATUSS [*]
 3 SA_ANESTHESIA_RECORD_STATUS_ID = F8
 3 STATUS_TYPE_FLAG = I2
 3 PRINT_STATUS_FLAG = I2
 3 PRSNL_ID = F8
 3 STATUS_DT_TM = DQ8
 3 ACTIVE_IND = I2
 3 UPDT_CNT = I4
 2 ANESTHESIARECORDSIGNATURES [*]
 3 SA_ANESTHESIA_REC_SIGNATURE_ID = F8
 3 SIGNED_BY_PRSNL_ID = F8
 3 SIGNED_BY_NAME_FULL_FORMATTED = VC
 3 SIGNED_BY_USERNAME = VC
 3 SIGNED_DT_TM = DQ8
 3 ACTIVE_IND = I2
 3 UPDT_CNT = I4
 2 ANESTHESIARECORDFIELDVALUES [*]
 3 SA_ANESTHESIA_REC_FIELD_VAL_ID = F8
 3 PERSON_ID = F8
 3 PERSON_NAME = VC
 3 ENCNTR_ID = F8
 3 ANESTHESIA_TYPE_CD = F8
 3 ANESTHESIA_TYPE_DISP = VC
 3 ANESTHESIOLOGIST_ID = F8
 3 ANESTHESIOLOGIST_NAME = VC
 3 ASA_CLASS_CD = F8
 3 ASA_CLASS_DISP = VC
 3 NPO_DT_TM = DQ8
```

```
  3  OR_CD  =  F8
  3  OR_DISP  =  VC
  3  PRE_OP_DIAGNOSIS_TXT  =  VC
  3  ADMIT_REASON  =  VC
  3  PROCEDURE_NOMENCLATURE_ID  =  F8
  3  PROCEDURE_NAME  =  VC
  3  DIAGNOSIS_NOMENCLATURE_ID  =  F8
  3  DIAGNOSIS_NAME  =  VC
  3  PROBLEM_NOMENCLATURE_ID  =  F8
  3  PROBLEM_NAME  =  VC
  3  PHYSICIAN_ID  =  F8
  3  PHYSICIAN_NAME  =  VC
  3  CARDIOLOGIST_ID  =  F8
  3  CARDIOLOGIST_NAME  =  VC
  3  ACTIVE_IND  =  I2
  3  UPDT_CNT  =  I4
  1  STATUS_DATA
  2  STATUS  =  C1
  2  SUBEVENTSTATUS [1 ]
  3  OPERATIONNAME  =  C25
  3  OPERATIONSTATUS  =  C1
  3  TARGETOBJECTNAME  =  C25
  3  TARGETOBJECTVALUE  =  VC
  )


record FREC(
  1 FILE_DESC = I4
  1 FILE_OFFSET =I4
  1 FILE_DIR = I4
  1 FILE_NAME = VC
  1 FILE_BUF = VC
)


if(not(validate(isopsjob)))
  declare isopsjob = i2 with noconstant(0), protect
  if(validate(request->batch_selection))
    set isopsjob = 1
  endif
endif

if(not(isopsjob))
```

```
  record request(
  1 batch_selection = vc
  1 output_dist = vc
  1 ops_date = dq8
  )
endif

if(not(validate(reply->status_data)))
  record reply(
  1 ops_event = vc
  1 status_data
    2 status = c1
    2 subeventstatus[1]
    3 operationname = c25
    3 operationstatus = c1
    3 targetobjectname = c25
    3 targetobjectvalue = vc
  )
endif


Record output(
1 cnt = i4
1 list[*]
  2 event_id = f8
  2 blob_seq = i4
  2 case_num = vc
  2 bloblen = i4
  2 img = gc
  2 finalized_rec = i4
  2 personId = f8
  2 pat_first = vc
  2 pat_last = vc
  2 pat_mi = vc
  2 MRN = vc
  2 encntr_id = f8
  2 dos = vc
  2 fn = vc
)
/*************************************************************
; DVDev DECLARED SUBROUTINES
*************************************************************/
declare EchoOut =vc
```

```
;default max for a VC field is 1048576 but this limit can be extended SET MODIFY MAXVARLEN command
set modify maxvarlen 50000000 ;set new max var size
/************************************************************
; DVDev DECLARED VARIABLES
************************************************************/
;***** declare blob working variables
declare good_blob = gc
declare print_blob = vc
declare outbuf = c32768
declare blobout = vc
declare retlen = i4
declare offset = i4
declare newsize = i4
declare finlen = i4
declare xlen=i4
;declare blobseqmax = i4
declare filepath = vc with protect, constant("CCLUSERDIR:")
declare AnesPrintedEventCd = f8 with protect,
constant(uar_get_code_by("DISPLAYKEY",72,"ANESTHESIAPRINTEDRECORD"))
declare AUTHVERIFIEDCD = f8 with protect, constant(uar_get_code_by("DISPLAYKEY",8,"AUTHVERIFIED"))
declare sbm = i4 with noconstant(768)
set runDate = build2(Replace(format(cnvtdatetime(curdate,curtime), "YYYY/MM/DD;;d"),"/",""))
;declare refDocTypeCD = f8 with protect, constant(198152095.00)
/*
* adjust params for backend use/opsjob
*/
  if ($EVENT_START = "")
  set event_start = cnvtdatetime(curdate-10,000000);look three days back per the specification
  set event_end = cnvtdatetime(curdate,235959)

  set manual_run_flag = 0
  call echoout("null event if")
  else
  set event_start = cnvtdatetime(build2($EVENT_START, "00:00:00"))
  set event_end = cnvtdatetime(build2($EVENT_END, "23:59:00"))

  set manual_run_flag = 1
  call echoout("pop tm if")
  endif

call echoout(build2("event_start: ", $event_start, " en: ", $event_end))
```

```
free record request
record REQUEST (
  1 debug_ind = i2
  1 actives_flag = i2
  1 unassociated_only_ind = i2
  1 search_by_out_of_or_ind = i2
  1 person_id = f8
  1 sa_anesthesia_record_id = f8
  1 surgical_case_id = f8
  1 created_by_prsnl_id = f8
  1 create_location_cd = f8
  1 sa_ref_doc_type_id = f8
  1 sa_doc_type_cd = f8
  1 last_doc_prsnl_id = f8
  1 start_dt_tm = dq8
  1 end_dt_tm = dq8
)

;use the stock cerner script to get the anesthesia cases to reduce descrepant records.
call EchoOut("Starting Anes Cases QUERY...")
  set request->end_dt_tm = event_end
  set request->start_dt_tm = event_start
  set request->actives_flag = 1
  set request->unassociated_only_ind = 0
  set request->search_by_out_of_or_ind = 0
  set request->person_id = 0
  set request->sa_anesthesia_record_id = 0
  set request->surgical_case_id = 0.0
  set request->created_by_prsnl_id = 0
  set request->create_location_cd = 0
  set request->sa_ref_doc_type_id = 0 ; refDocTypeCD ;DOC TYPE CODE for Facility 0 pulls all facilities
  set request->sa_doc_type_cd = 0
  set request->last_doc_prsnl_id = 0
  execute sa_get_anesthesia_records with replace("REPLY",CASES_REPLY)
  ;call echorecord(REPLY)

if(size(cases_reply->ANESTHESIARECORDS,5)>0)


  set stat = Alterlist(output->list, size(cases_reply->ANESTHESIARECORDS,5))

  for (ii = 1 to size(CASES_REPLY->ANESTHESIARECORDS,5))
```

```
 call echoout(build2("eventid: ", cases_reply->ANESTHESIARECORDS[ii].RECORD_DESCRIPTION, "
surgcaseid: ", cases_reply->
 ANESTHESIARECORDS[ii].SURGICAL_CASE_ID))



 select into "nl:"
 from
 surgical_case sc
 ,sa_anesthesia_record sar
 ,sa_anesthesia_rec_status sars
 ,clinical_event ce
 ,ce_blob cb



 plan sc where sc.SURG_CASE_ID = cases_reply->ANESTHESIARECORDS[ii].SURGICAL_CASE_ID
 join sar where sar.surgical_case_id = sc.surg_case_id
 join sars where sars.sa_anesthesia_record_id = sar.sa_anesthesia_record_id
 and sars.active_ind = 1
 and sars.status_type_flag = 0;finalized only
 join ce where ce.ENCNTR_ID = sc.ENCNTR_ID
 and ce.PERSON_ID = sc.PERSON_ID
 and ce.SUBTABLE_BIT_MAP = sbm
 and ce.CATALOG_CD = 0.0
 and ce.ORDER_ID = 0.0
 and ce.EVENT_CD = AnesPrintedEventCd
 and ce.RESULT_STATUS_CD = AUTHVERIFIEDCD
 and CE.VALID_UNTIL_DT_TM > cnvtdatetime(curdate, curtime3)
 join cb where cb.EVENT_ID = ce.EVENT_id
   and  cb.VALID_FROM_DT_TM < cnvtdatetime(curdate,curtime3)
  and  cb.VALID_UNTIL_DT_TM > cnvtdatetime(curdate,curtime3)

 order by cb.event_id, cb.blob_seq_num
 head  cb.event_id
 output->cnt = 0



 detail
 output->list[ii].case_num = sc.surg_case_nbr_formatted
 output->list[ii].blob_seq = cnvtint(max(cb.blob_seq_num))
 output->list[ii].event_id = ce.event_id
 output->list[ii].bloblen = cb.blob_length
```

```
 output->list[ii].personId = sc.person_id
 output->list[ii].encntr_id = sc.encntr_id
 ;doslogic
 if(sc.SURG_START_DT_TM != null)
 output->list[ii].dos = build2(Replace(format(cnvtdatetime(sc.surg_start_dt_tm), "YYYY/MM/DD;;d"),"/",""))
 elseif(sar.CREATE_DT_TM != null)
 output->list[ii].dos = build2(Replace(format(cnvtdatetime(sar.CREATE_DT_TM), "YYYY/MM/DD;;d"),"/",""))
 else
 output->list[ii].dos = build2(Replace(format(cnvtdatetime(sars.STATUS_DT_TM), "YYYY/MM/DD;;d"),"/",""))
 endif
 with nocounter
 if(curqual >0)
 ;if a blob exists then continue with the querying
 ;get info for file name
 select into "nl:"
 from
 person p
 ,encounter e
 ,person_alias pa

 plan p
 where p.person_id = output->list[ii].personId
 join e
 where e.person_id = p.person_id and e.encntr_id = output->list[ii].encntr_id
 and e.beg_effective_dt_tm <= outerjoin(cnvtdatetime(curdate,curtime3))
   and e.end_effective_dt_tm >= outerjoin(cnvtdatetime(curdate,curtime3))
   and e.active_ind = outerjoin(1)
 join pa where pa.person_id = p.person_id
 and pa.beg_effective_dt_tm <= outerjoin(cnvtdatetime(curdate,curtime3))
   and pa.end_effective_dt_tm >= outerjoin(cnvtdatetime(curdate,curtime3))
   and pa.active_ind = outerjoin(1)
 detail
 output->list[ii].pat_first = trim(p.name_first)
 output->list[ii].pat_last = trim(p.name_last)
 output->list[ii].pat_mi = trim(p.name_middle)
 output->list[ii].MRN = TRIM(pa.alias)
 ;output->list[ii].dos = cases_reply->ANESTHESIARECORDS[ii].CREATE_DT_TM
 with nocounter

 call echoout("cb query starting...")
 ;extract blob info & output to file
 select into "nl:"
 from
```

```
 ce_blob cb
 where cb.event_id = output->list[ii].event_id ;and cb.blob_seq_num = jj
 head report
 cnt = 0
 detail
 retlen = 1
 offset = 0
 cnt = cnt+1


 call echoout("BLOB Getting...")
 retlen = blobget(outbuf, offset, cb.blob_contents)
 offset = offset + retlen

 ;strip ocf_blob off the end
 xlen = findstring("ocf_blob",outbuf,1)-1

 ;concat into good_blob with out OCF delimiter
        good_blob = notrim(concat(notrim(good_blob), notrim(substring(1,xlen,outbuf))))
;is detail done detailing?
if(size(good_blob,4) > 0 and cnt = output->list[ii].blob_seq+1)
;name file depending on existance of middle initial
call echoout(build2("middle I: ",output->list[ii].pat_mi))
output->list[ii].pat_mi = concat(trim(output->list[ii].pat_mi),"_")
if(output->list[ii].pat_mi != "_")
FREC->FILE_NAME= build2(filepath,"_",output->list[ii].dos,"_",output->list[ii].pat_last,"_",output-
>list[ii].pat_first,"_",
output->list[ii].pat_mi,output->list[ii].case_num,"_",output->list[ii].MRN,"_",trim(runDate),".TIF")
output->list[ii].fn = value(FREC->FILE_NAME)
else;person has a middle initial
FREC->FILE_NAME= build2(filepath,"_",output->list[ii].dos,"_",output->list[ii].pat_last,"_",output-
>list[ii].pat_first,"_",
output->list[ii].case_num,"_",output->list[ii].MRN,"_",trim(runDate),".TIF")
output->list[ii].fn = value(FREC->FILE_NAME)
endif
      frec->file_buf = "w" ;write to beginning of new file
      stat = CCLIO("OPEN",FREC)
call echoout(build2("FCW: ", ii))
call echoout(build2(" FN: ", FREC->FILE_NAME, " Bsz: ", size(good_blob,4) ))
  frec->file_buf = good_blob
    stat = CCLIO("WRITE",FREC)
    stat = CCLIO("FLUSH",FREC)
    stat = CCLIO("CLOSE",FREC);Close File
```

```
    good_blob = null

  endif


  with  format=undefined,RDBARRAYFETCH = 1
  endif

endfor ;end of main loop
  ;output to screen for user
  set reply->status_data.status = "S"
SELECT INTO $outdev


FROM
  (DUMMYT   D1  WITH SEQ = VALUE(SIZE(OUTPUT->list, 5)))


PLAN D1


head report

"Files written to sFTP Directory: " filepath
row + 1

detail
if(trim(output->list[d1.seq].fn) !=" ")
col 0 output->list[d1.seq].fn
row+1
endif




WITH MAXCOL = 32000, MAXROW = 1, format=variable, formfeed=none, noheading


elseif(size(cases_reply->ANESTHESIARECORDS,5)=0);no anesthesia cases
  set reply->status_data.status = "Z"
```

```
  select into $outdev
  from
  dummyt dt
  head report
  "NO QUALIFYING RECORDS FOUND"
else
  set reply->status_data.status = "F"
endif;end of program control structure


/*************************************************************
; DVDev DEFINED SUBROUTINES
*************************************************************/
SUBROUTINE EchoOut(echo_str)
  call echo(concat(echo_str,"  @",format(cnvtdatetime(curdate,curtime3),"MM/DD/YYYY HH:MM:SS;;D")))
END ;EchoOut
SET LAST_MOD = "@STINSLEY - VERSION 1.0  - 01/26/16"
CALL ECHO (LAST_MOD)


end
go
```

358 Views Tags: ccl, anesthesia, extract, blob, ce_blob, printed record

There are no comments on this post