Supplemental Digital Content 2

/\*

All the table names and variable/field names have been changed to obscure EPIC field names.

Code is written in the PL/SQL script used for Oracle™ systems.

\*/

/\* 1. Post-Operative Anti-emetic Medications. This query will report the unique case identification number and the number of anti-emetic medications administered during the recovery phase of a patient’s care. The medications detected are ondansetron, metoclopramide, dexamethasone, promethazine, diphenhydramine, and haloperidol. The returned value ranges from 0-5 depending on the number of medications administered. \*/

-- VERSION I -– Code Written Using Case Based Tables

SELECT procedure\_case\_identification\_number,ponv\_tx\_count

FROM case\_based\_pacu\_metrics\_table;

-- VERSION II -– Code Written Using Base Tables

SELECT procedure\_case\_identification\_number,sum(ponv\_tx\_pacu\_yn) AS ponv\_tx\_count

FROM

( SELECT procedure\_case\_list\_table.procedure\_case\_identification\_number,

/\* This will get the number of medications given if the dosage is greater than 0 units. \*/

MAX(CASE

WHEN recorded\_dosage > 0

AND medication\_administration\_base\_table.medication\_administration\_time <=

event\_description.stop\_recovery

THEN 1

ELSE 0

END) AS ponv\_tx\_pacu\_yn

FROM medication\_administration\_base\_table

/\* Will inner join with the procedure/surgical list to ensure that there are no non-surgical patient included \*/

INNER JOIN procedure\_case\_list\_table

ON procedure\_case\_list\_table.patient\_unique\_identification\_number =

medication\_administration\_base\_table.patient\_unique\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = recovery\_stop\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS stop\_recovery,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM mv\_schema\_case\_events

WHERE timestamp\_marker\_identifier IN

(in\_room\_identifier,procedure\_start\_identifier,recovery\_start\_identifier,

recovery\_stop\_identifier,paper\_chart\_identifier)

GROUP BY procedure\_case\_identification\_number

) event\_description

ON event\_description.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM all\_staff\_table

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_case\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM anesthesia\_staffing\_table

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

WHERE medication\_administration\_base\_table.medication\_administration\_time >=

least(COALESCE(procedure\_case\_list\_table.anesthesia\_procedure\_stop\_time,

event\_description.recovery\_start),COALESCE(event\_description.recovery\_start,

procedure\_case\_list\_table.anesthesia\_procedure\_stop\_time))

AND medication\_administration\_base\_table.medication\_administration\_time <=

event\_description.stop\_recovery

/\* These represent the identification numbers for all the different instances of the medications desired from the medication table. Current selected medications include: ondansetron,dexamethasone,haloperidol,diphenhydramine,promethazine,and metoclopramide. \*/

AND medication\_id IN

(ondansetron\_1,ondansetron\_2,ondansetron\_3,ondansetron\_4,ondansetron\_5,ondansetron\_6

dexamethasone\_1,dexamethasone\_2,dexamethasone\_3,dexamethasone\_4,dexamethasone\_5,

dexamethasone\_6,dexamethasone\_7,dexamethasone\_8,haloperidol\_1,haloperidol\_2,

haloperidol\_3,haloperidol\_4,haloperidol\_5,haloperidol\_6,diphenhydramine\_1,

diphenhydramine\_2,diphenhydramine\_3,diphenhydramine\_4,diphenhydramine\_5,

promethazine\_1,promethazine\_2,promethazine\_3,promethazine\_4,promethazine\_5,

promethazine\_6,promethazine\_7,metoclopramide\_1,metoclopramide\_2,metoclopramide\_3,

metoclopramide\_4,metoclopramide\_5,metoclopramide\_6)

/\* Case Inclusion criteria. A case/procedure must have either:

1. An anesthesia start and start time and stop time with an anesthesia attending physician assigned

2. Must be a paper chart used in a setting with no EMR access

3. Must have recorded times such as: in room time, procedure start time, or recovery start time

4. Must have a surgeon with documented minutes in a case. \*/

AND

(

( anesthesia\_procedure\_start\_time IS NOT NULL

AND anesthesia\_procedure\_stop\_time IS NOT NULL

AND att\_count >=1

)

OR COALESCE(paper\_chart,in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes >0

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( ucla\_loc = 'SM ANESTHESIA FLOOR'

OR ucla\_loc = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(procedure\_title\_description) LIKE '%labor%'

OR lower(procedure\_title\_description) LIKE '%cse%'

OR lower(procedure\_title\_description) LIKE '%l'||chr(38)||'d%'

OR lower(procedure\_title\_description) LIKE '%l '||chr(38)||' d%'

OR lower(procedure\_title\_description) LIKE '%l and d%'

OR lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%obstetric%'

OR lower(procedure\_title\_description) LIKE '%section%'

OR lower(procedure\_title\_description) LIKE '%c/s%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR lower(procedure\_title\_description) LIKE '%management%'

OR lower(procedure\_title\_description) LIKE '%intubation%'

OR lower(procedure\_title\_description) LIKE '%treatment%'

OR lower(procedure\_title\_description) LIKE '%new start%'

OR lower(procedure\_title\_description) LIKE '%epidural%'

OR procedure\_title\_description IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery. Only surgical gynecologic cases included. \*/

AND NOT

(

( ucla\_loc = 'SM OBOR'

OR ucla\_loc = 'RR OBOR'

)

AND

( lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR procedure\_title\_description IS NULL

)

)

GROUP BY procedure\_case\_list\_table.procedure\_case\_identification\_number

)

GROUP BY procedure\_case\_identification\_number;

-- VERSION III -– Code Written Using Only Clarity Tables

SELECT procedure\_record\_identification\_number, sum(ponv\_tx\_pacu\_yn) AS ponv\_tx\_count

FROM

( SELECT procedure\_record\_table.procedure\_record\_identification\_number,

MAX(CASE

WHEN medication\_administration\_data\_table.recorded\_dosage > 0

AND medication\_administration\_data\_table.medication\_administration\_time <=

event\_description.stop\_recovery

THEN 1

ELSE 0

END) AS ponv\_tx\_pacu\_yn

FROM medication\_information\_table

LEFT OUTER JOIN medication\_administration\_data\_table

ON medication\_administration\_data\_table.order\_med\_identification\_number =

medication\_information\_table.order\_med\_identification\_number

LEFT OUTER JOIN procedure\_record\_table

ON procedure\_record\_table.patient\_unique\_identification\_number =

medication\_information\_table.patient\_unique\_identification\_number

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,MIN(documented\_event\_time)

AS documented\_event\_time

FROM procedure\_review\_case\_times

WHERE tracking\_event\_description\_data = in\_room\_marker

GROUP BY procedure\_record\_identification\_number

) anesthesia\_procedure\_times\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_procedure\_times\_table.procedure\_record\_identification\_number

/\* During the validation phase of coding, it was discovered that the anesthesia\_review\_table was incomplete for many rows and had left a significant number of unmatched rows when joined on a single variable. For that reason, the table needs to be joined 3 times. The first join uses the anesthesia\_specific\_procedure\_record\_identification\_number. The second join uses the patient identification number and time of surgery. The last uses that patient identification number and the surgery date. Every subsequent join is more generalized and only matches rows that were unmatched previously. \*/

LEFT OUTER JOIN anesthesia\_review\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_2

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_2.anesthesia\_specific\_patient\_unique\_identification\_number

AND anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time =

COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time)

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_3

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_3.anesthesia\_specific\_patient\_unique\_identification\_number

AND trunc(anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) =

trunc(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time))

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM procedure\_review\_all\_staff\_table astaff

LEFT OUTER JOIN improved\_specification\_position\_table improved\_specification\_position

ON improved\_specification\_position.role\_description\_data = astaff.position\_description

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_record\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_record\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT encounter\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = recovery\_stop\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS stop\_recovery,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM event\_description\_table

LEFT OUTER JOIN event\_description\_table\_2

ON event\_description\_table\_2.event\_identification\_number =

event\_description\_table.event\_identification\_number

WHERE timestamp\_marker\_identifier IN

(in\_room\_identifier,procedure\_start\_identifier,recovery\_start\_identifier,

recovery\_stop\_identifier,paper\_chart\_identifier)

GROUP BY encounter\_identification\_number

) event\_description

ON COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_2.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_3.anesthesia\_procedure\_identification\_number) =

event\_description.encounter\_identification\_number

/\* Joining the procedure location information. \*/

LEFT OUTER JOIN location\_information\_table

ON procedure\_record\_table.location\_information\_identification\_number =

location\_information\_table.location\_information\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. Note that the system used to track anesthesia providers was updated and the data was moved to a new table, but the old data was never transferred. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM (

SELECT staff.procedure\_record\_identification\_number AS procedure\_case\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_identification\_number

AS anesthesia\_provider\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_position AS anesthesia\_provider\_position

FROM antiquated\_or\_staff\_table staff

JOIN antiquated\_staff\_table

ON antiquated\_staff\_table.record\_identification\_number=staff.anesthesia\_staff\_identification\_number

UNION

SELECT anesthesia\_review\_table.procedure\_record\_identification\_number

AS procedure\_case\_identification\_number,

staff.anesthesia\_specific\_prov\_identification\_number AS anesthesia\_provider\_identification\_number,

staff.anesthesia\_specific\_prov\_type\_description\_data AS anesthesia\_provider\_position

FROM anesthesia\_review\_table

INNER JOIN anesthesia\_specific\_staff staff

ON anesthesia\_review\_table.anesthesia\_specific\_episode\_identification\_number =

staff.summary\_block\_identification\_number

)

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

WHERE medication\_administration\_data\_table.medication\_administration\_time >=

least(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_stop\_time,

event\_description.recovery\_start),COALESCE(event\_description.recovery\_start,

anesthesia\_review\_table.anesthesia\_procedure\_stop\_time))

AND medication\_administration\_data\_table.medication\_administration\_time <=

event\_description.stop\_recovery

/\* These represent the identification numbers for all the different instances of the medications desired from the medication table. Current selected medications include: ondansetron,dexamethasone,haloperidol,diphenhydramine,promethazine,and metoclopramide. \*/

AND medication\_identification\_number IN

(ondansetron\_1,ondansetron\_2,ondansetron\_3,ondansetron\_4,ondansetron\_5,ondansetron\_6

dexamethasone\_1,dexamethasone\_2,dexamethasone\_3,dexamethasone\_4,dexamethasone\_5,

dexamethasone\_6,dexamethasone\_7,dexamethasone\_8,haloperidol\_1,haloperidol\_2,

haloperidol\_3,haloperidol\_4,haloperidol\_5,haloperidol\_6,diphenhydramine\_1,

diphenhydramine\_2,diphenhydramine\_3,diphenhydramine\_4,diphenhydramine\_5,

promethazine\_1,promethazine\_2,promethazine\_3,promethazine\_4,promethazine\_5,

promethazine\_6,promethazine\_7,metoclopramide\_1,metoclopramide\_2,metoclopramide\_3,

metoclopramide\_4,metoclopramide\_5,metoclopramide\_6)

AND

( /\* Case Inclusion criteria. A Case must have either:

1. An anesthesia start and start time and stop time with an anesthesia attending physician assigned

2. Must be a paper chart used in a setting with no EMR access

3. Must have recorded times such as: in room time, procedure start time, or recovery start time

4. Must have a surgeon with documented minutes in a case. \*/

( COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) IS NOT NULL

AND COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_stop\_time) IS NOT NULL

AND att\_count >=1

)

OR COALESCE(paper\_chart,in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes >0

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM ANESTHESIA FLOOR'

OR location\_information\_table.location\_description = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%labor%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cse%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l'||chr(38)||'d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l '||chr(38)||' d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l and d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%deliver%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%obstetric%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%section%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%c/s%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cesarean%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%management%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%intubation%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%treatment%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%new start%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%epidural%'

OR COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description) IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery. Only surgical gynecologic cases included. \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM OBOR'

OR location\_information\_table.location\_description = 'RR OBOR'

)

AND

( lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%deliver%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cesarean%'

OR COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description) IS NULL

)

)

GROUP BY procedure\_record\_table.procedure\_record\_identification\_number

)

GROUP BY procedure\_record\_identification\_number;

/\* 2. Same Day Cancellation. This query will report the unique case identification number and the original most recent scheduled procedure time. \*/

-- VERSION I -– Code Written Using Case Based Tables

SELECT procedure\_case\_identification\_number, date\_of\_service

FROM case\_based\_cancellation\_data\_table

WHERE case\_status = 'CANCELLED'

AND relative\_date\_changed = 0;

-- VERSION II -– Code Written Using Base Tables

SELECT procedure\_case\_identification\_number, original\_scheduled\_procedure\_time

AS date\_of\_service

FROM schedule\_history\_table

WHERE schedule\_history\_table.procedure\_case\_identification\_number NOT IN

/\* Using this nested select statement to make sure that the procedure\_case\_identification\_number is not one of the numbers that exists in the table of cases that have already occurred. \*/

( SELECT procedure\_case\_list\_table.procedure\_case\_identification\_number

FROM procedure\_case\_list\_table

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number, sum(duration\_in\_minutes) AS surg\_minutes

FROM all\_staff\_table

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_case\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number, count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM anesthesia\_staffing\_table

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM anesthesia\_procedure\_events\_table

WHERE timestamp\_marker\_identifier IN

(in\_room\_identifier,procedure\_start\_identifier,recovery\_start\_identifier,paper\_chart\_identifier)

GROUP BY procedure\_case\_identification\_number

) event\_description

ON event\_description.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

WHERE

( /\* Case Inclusion criteria. A Case must have either:

1. An anesthesia start and start time and stop time with an anesthesia attending physician assigned

2. Must be a paper chart used in a setting with no EMR access

3. Must have recorded times such as: in room time, procedure start time, or recovery start time

4. Must have a surgeon with documented minutes in a case. \*/

( anesthesia\_procedure\_start\_time IS NOT NULL

AND anesthesia\_procedure\_stop\_time IS NOT NULL

AND att\_count >=1

)

OR COALESCE(paper\_chart,in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes >0

)

)

AND schedule\_history\_table.original\_scheduled\_procedure\_time <

/\* This nested select statement looks for the largest value in log tables to ensure that the reason for a case not existing in the table is not because of lapsed updates. The times include: in room time, procedure start time, anesthesia start time, pre-procedure complete time, recovery start time, surgical log date. \*/

( SELECT MAX(COALESCE(trunc(in\_room),trunc(procedure\_start\_time),trunc(anesthesia\_procedure\_start\_time),

trunc(preprocedure\_comp),trunc(recovery\_start),procedure\_review\_date))

FROM procedure\_case\_list\_table

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart,

MAX(CASE

WHEN timestamp\_marker\_identifier = pre\_procedure\_complete\_identifier

THEN timestamp\_marker\_time

END) AS preprocedure\_comp,

MAX(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

END) AS procedure\_start\_time

FROM anesthesia\_procedure\_events\_table

WHERE timestamp\_marker\_identifier IN

(in\_room\_identifier,procedure\_start\_identifier,recovery\_start\_identifier,paper\_chart\_identifier,

procedure\_start\_identifier,pre\_procedure\_complete\_identifier)

GROUP BY procedure\_case\_identification\_number

) event\_description

ON event\_description.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

)

/\* Excludes cases that were never scheduled though the scheduling system. \*/

AND original\_scheduled\_procedure\_time IS NOT NULL

/\* Make sure that the case was not moved to another day. \*/

AND next\_sched\_time IS NULL

/\* Same day cancellation since the cancel date and scheduled date are the same. \*/

AND trunc(original\_scheduled\_procedure\_time)-trunc(or\_case\_cancel\_date) = 0

GROUP BY procedure\_case\_identification\_number,original\_scheduled\_procedure\_time;

-- VERSION III -– Code Written Using Only Clarity Tables

SELECT procedure\_case\_identification\_number, date\_of\_service

FROM

( SELECT procedure\_case\_description\_table.procedure\_case\_identification\_number,

scheduled\_procedure\_table.original\_scheduled\_procedure\_time AS date\_of\_service,

/\* This looks at the time that the case was moved to in the event that a case was rescheduled and not cancelled.\*/

lead (scheduled\_procedure\_table.original\_scheduled\_procedure\_time) OVER (

PARTITION BY procedure\_case\_description\_table.procedure\_case\_identification\_number

ORDER BY procedure\_case\_description\_table.procedure\_case\_identification\_number ASC,

/\* This combines that date and time into a single timestamp. \*/

CASE

WHEN scheduled\_procedure\_time IS NOT NULL

THEN to\_timestamp(to\_char(scheduled\_procedure\_date, 'mm/dd/yyyy')

||' '||to\_char(scheduled\_procedure\_time, 'hh24:mi'), 'mm/dd/yyyy hh24:mi')

ELSE NULL

END ASC, scheduled\_procedure\_table.original\_scheduled\_procedure\_time ASC) AS next\_case,cancel\_date

FROM procedure\_case\_description\_table

LEFT OUTER JOIN scheduled\_procedure\_table

ON procedure\_case\_description\_table.procedure\_case\_identification\_number =

scheduled\_procedure\_table.procedure\_case\_identification\_number

WHERE procedure\_case\_description\_table.procedure\_case\_identification\_number NOT IN

/\* Using this nested select statement to make sure that the procedure\_case\_identification\_number is not one of the numbers that exists in the table of cases that have already occurred. \*/

( SELECT procedure\_record\_table.procedure\_record\_identification\_number

FROM procedure\_record\_table

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,MIN(documented\_event\_time)

AS documented\_event\_time

FROM procedure\_review\_case\_times\_table

WHERE tracking\_event\_description\_data = in\_room\_marker

GROUP BY procedure\_record\_identification\_number

) anesthesia\_procedure\_times\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_procedure\_times\_table.procedure\_record\_identification\_number

/\* During the validation phase of coding, it was discovered that the anesthesia\_review\_table was incomplete for many rows and had left a significant number of unmatched rows when joined on a single variable. For that reason, the table needs to be joined 3 times. The first join uses the anesthesia\_specific\_procedure\_record\_identification\_number. The second join uses the patient identification number and time of surgery. The last uses that patient identification number and surgery date. Every subsequent join is more generalized and only matches rows that were unmatched previously. \*/

LEFT OUTER JOIN anesthesia\_review\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_2

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_2.anesthesia\_specific\_patient\_unique\_identification\_number

AND anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time = COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time)

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_3

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_3.anesthesia\_specific\_patient\_unique\_identification\_number

AND trunc(anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) =

trunc(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time))

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM procedure\_review\_all\_staff\_table astaff

LEFT OUTER JOIN improved\_specification\_position\_table improved\_specification\_position

ON improved\_specification\_position.role\_description\_data = astaff.position\_description

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_record\_identification\_number

) all\_staff\_data ON all\_staff\_data.procedure\_record\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. Note that the system used to track anesthesia providers was updated and the data was moved to a new table, but the old data was never transferred. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM

( SELECT staff.procedure\_record\_identification\_number AS procedure\_case\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_identification\_number

AS anesthesia\_provider\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_position AS anesthesia\_provider\_position

FROM antiquated\_or\_staff\_table\_table staff

JOIN antiquated\_staff\_table

ON antiquated\_staff\_table.procedure\_record\_identification\_number =

staff.anesthesia\_staff\_identification\_numberdentification\_number

UNION

SELECT anesthesia\_review\_table.procedure\_record\_identification\_number AS

procedure\_case\_identification\_number,

staff.anesthesia\_specific\_prov\_identification\_number AS anesthesia\_provider\_identification\_number,

staff.anesthesia\_specific\_prov\_type\_description\_data AS anesthesia\_provider\_position

FROM anesthesia\_review\_table

INNER JOIN anesthesia\_specific\_staff staff

ON anesthesia\_review\_table.anesthesia\_specific\_episode\_identification\_number =

staff.staff\_description\_identification\_number

)

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT encounter\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = recovery\_stop\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS stop\_recovery,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM event\_description\_table

LEFT OUTER JOIN event\_description\_table\_2

ON event\_description\_table\_2.event\_identification\_number = event\_description\_table.event\_id

WHERE timestamp\_marker\_identifier IN (in\_room\_identifier,procedure\_start\_identifier,

recovery\_start\_identifier,recovery\_stop\_identifier,paper\_chart\_identifier)

GROUP BY encounter\_identification\_number

) event\_description

ON COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_2.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_3.anesthesia\_procedure\_identification\_number) =

event\_description.encounter\_identification\_number

WHERE

( /\* Case Inclusion criteria. A Case must have either:

1. An anesthesia start and start time and stop time with an anesthesia attending physician assigned

2. Must be a paper chart used in a setting with no EMR access

3. Must have recorded times such as: in room time, procedure start time, or recovery start time

4. Must have a surgeon with documented minutes in a case. \*/

( COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) IS NOT NULL

AND COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_stop\_time) IS NOT NULL

AND att\_count >= 1

)

OR COALESCE(paper\_chart,in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes>0

)

)

/\* Ensuring that the case was scheduled through the case scheduling system. \*/

AND scheduled\_procedure\_table.original\_scheduled\_procedure\_time IS NOT NULL

AND scheduled\_procedure\_table.original\_scheduled\_procedure\_time <

/\* This nested select statement looks for the largest value in log tables to ensure that the reason for a case not existing in the table is not because of lapsed updates. The times include: in room time, procedure start time, anesthesia start time, pre-procedure complete time, recovery start time, surgical log date. \*/

( SELECT MAX(COALESCE(trunc(in\_room),trunc(procedure\_start\_time),trunc(

COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time)),trunc(preprocedure\_comp),

trunc(recovery\_start),surgery\_date))

FROM procedure\_record\_table

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,

MIN(documented\_event\_time) AS documented\_event\_time

FROM procedure\_review\_case\_times\_table

WHERE tracking\_event\_description\_data = in\_room\_marker

GROUP BY procedure\_record\_identification\_number

) anesthesia\_procedure\_times\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_procedure\_times\_table.procedure\_record\_identification\_number

/\* During the validation phase of coding, it was discovered that the anesthesia\_review\_table was incomplete for many rows and had left a significant number of unmatched rows when joined on a single variable. For that reason, the table needs to be joined 3 times. The first join uses the anesthesia\_specific\_procedure\_record\_identification\_number. The second join uses the patient identification number and time of surgery. The last uses that patient identification number and the surgery date. Every subsequent join is more generalized and only matches rows that were unmatched previously. \*/

LEFT OUTER JOIN anesthesia\_review\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_2

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_2.anesthesia\_specific\_patient\_unique\_identification\_number

AND anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time =

COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time)

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_3

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_3.anesthesia\_specific\_patient\_unique\_identification\_number

AND trunc(anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) = trunc(COALESCE(

anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time))

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM procedure\_review\_all\_staff\_table astaff

LEFT OUTER JOIN improved\_specification\_position\_table improved\_specification\_position

ON improved\_specification\_position.role\_description\_data = astaff.position\_description

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_record\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_record\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. Note that the system used to track anesthesia providers was updated and the data was moved to a new table, but the old data was never transferred. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM

( SELECT staff.procedure\_record\_identification\_number AS procedure\_case\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_identification\_number

AS anesthesia\_provider\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_position AS anesthesia\_provider\_position

FROM antiquated\_or\_staff\_table staff

JOIN or\_lnlg\_anes\_resp antiquated\_staff\_table

ON antiquated\_staff\_table.procedure\_record\_identification\_number =

staff.anesthesia\_staff\_identification\_number

UNION

SELECT anesthesia\_review\_table.procedure\_record\_identification\_number AS

procedure\_case\_identification\_number,

staff.anesthesia\_specific\_prov\_identification\_number AS anesthesia\_provider\_identification\_number,

staff.anesthesia\_specific\_prov\_type\_description\_data AS anesthesia\_provider\_position

FROM anesthesia\_review\_table

INNER JOIN anesthesia\_specific\_staff staff

ON anesthesia\_review\_table.anesthesia\_specific\_episode\_identification\_number =

staff.staff\_description\_identification\_number

)

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT encounter\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = recovery\_stop\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS stop\_recovery,

MAX(CASE

WHEN timestamp\_marker\_identifier = pre\_procedure\_complete\_identifier

THEN timestamp\_marker\_time

END) AS preprocedure\_comp,

MAX(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

END) AS procedure\_start\_time

FROM event\_description\_table

LEFT OUTER JOIN event\_description\_table\_2

ON event\_description\_table\_2.event\_identification\_number = event\_description\_table.event\_id

WHERE timestamp\_marker\_identifier IN (in\_room\_identifier,procedure\_start\_identifier,

recovery\_start\_identifier,procedure\_start\_identifier,pre\_procedure\_complete\_identifier)

GROUP BY encounter\_identification\_number

) event\_description

ON COALESCE(

anesthesia\_review\_table.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_2.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_3.anesthesia\_procedure\_identification\_number) =

event\_description.encounter\_identification\_number

)

)

/\* Excludes cases that were never scheduled though the scheduling system. \*/

WHERE next\_case IS NULL

/\* Make sure that the case was not moved to another day. \*/

AND trunc(date\_of\_service)-trunc(cancel\_date) = 0

/\* Same day cancellation since the cancel date and scheduled date are the same. \*/

GROUP BY procedure\_case\_identification\_number,date\_of\_service;

/\* 3. Last Intraoperative Train-of-Four Value. This query will report the unique case identification number, the value of most recent recorded train-of-four value, and the time that the value was recorded. These values are only entered during anesthesia cases. \*/

-- VERSION I -– Code Written Using Case Based Tables

SELECT procedure\_case\_identification\_number, last\_train\_of\_four\_time, last\_train\_of\_four\_value

FROM case\_based\_intraoperative\_metrics\_table;

-- VERSION II -– Code Written Using Base Tables

SELECT procedure\_case\_identification\_number, last\_train\_of\_four\_time, last\_train\_of\_four\_value

FROM

( SELECT procedure\_case\_list\_table.procedure\_case\_identification\_number,

first\_value(measurement\_data\_table.data\_entry\_time) OVER (

PARTITION BY procedure\_case\_list\_table.procedure\_case\_identification\_number

ORDER BY measurement\_data\_table.data\_entry\_time DESC) AS last\_train\_of\_four\_time,

first\_value(measurement\_data\_table.data\_entry\_value) OVER (

PARTITION BY procedure\_case\_list\_table.procedure\_case\_identification\_number

ORDER BY measurement\_data\_table.data\_entry\_time DESC) AS last\_train\_of\_four\_value

FROM measurement\_data\_table

LEFT OUTER JOIN procedure\_case\_list\_table

ON procedure\_case\_list\_table.procedure\_case\_identification\_number =

measurement\_data\_table.procedure\_case\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM anesthesia\_staffing\_table

WHERE anesthesia\_provider\_role = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

WHERE measurement\_data\_table.data\_item\_identification\_number = 'NERVE STIMULATION'

AND measurement\_data\_table.data\_entry\_value IS NOT NULL

AND

(

( anesthesia\_procedure\_start\_time IS NOT NULL

AND anesthesia\_procedure\_stop\_time IS NOT NULL

AND att\_count >=1

)

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( ucla\_loc = 'SM ANESTHESIA FLOOR'

OR ucla\_loc = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(procedure\_title\_description) LIKE '%labor%'

OR lower(procedure\_title\_description) LIKE '%cse%'

OR lower(procedure\_title\_description) LIKE '%l'||chr(38)||'d%'

OR lower(procedure\_title\_description) LIKE '%l '||chr(38)||' d%'

OR lower(procedure\_title\_description) LIKE '%l and d%'

OR lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%obstetric%'

OR lower(procedure\_title\_description) LIKE '%section%'

OR lower(procedure\_title\_description) LIKE '%c/s%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR lower(procedure\_title\_description) LIKE '%management%'

OR lower(procedure\_title\_description) LIKE '%intubation%'

OR lower(procedure\_title\_description) LIKE '%treatment%'

OR lower(procedure\_title\_description) LIKE '%new start%'

OR lower(procedure\_title\_description) LIKE '%epidural%'

OR procedure\_title\_description IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery. Only surgical gynecologic cases included. \*/

AND NOT

(

( ucla\_loc = 'SM OBOR'

OR ucla\_loc = 'RR OBOR'

)

AND

( lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR procedure\_title\_description IS NULL

)

)

)

GROUP BY procedure\_case\_identification\_number, last\_train\_of\_four\_time, last\_train\_of\_four\_value;

-- VERSION III -– Code Written Using Only Clarity Tables

SELECT anesthesia\_specific\_procedure\_record\_identification\_number, last\_train\_of\_four\_time, last\_train\_of\_four\_value

FROM

( SELECT anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number,

first\_value(measurement\_data\_table\_2.data\_entry\_time) OVER (

PARTITION BY anesthesia\_specific\_procedure\_record\_identification\_number

ORDER BY measurement\_data\_table\_2.data\_entry\_time DESC) AS last\_train\_of\_four\_time,

first\_value(measurement\_data\_table\_2.data\_entry\_value) OVER (

PARTITION BY anesthesia\_specific\_procedure\_record\_identification\_number

ORDER BY measurement\_data\_table\_2.data\_entry\_time DESC) AS last\_train\_of\_four\_value

/\* Joining the raw data tables. Each joined table has more information for each row of the previous table. \*/

FROM measurement\_data\_table

LEFT OUTER JOIN measurement\_data\_table\_2

ON measurement\_data\_table.fsd\_id = measurement\_data\_table\_2.fsd\_id

LEFT OUTER JOIN measurement\_data\_table\_3

ON measurement\_data\_table\_3.data\_item\_identification\_number =

measurement\_data\_table\_2.data\_item\_identification\_number

/\* Joining the table with information on the anesthesia encounter in order to access other anesthesia tables. \*/

LEFT OUTER JOIN encounter\_table

ON encounter\_table.inpatient\_data\_identification\_number = measurement\_data\_table.inpatient\_data\_id

LEFT OUTER JOIN anesthesia\_review\_table

ON anesthesia\_review\_table.anesthesia\_procedure\_identification\_number =

encounter\_table.encounter\_identification\_number

LEFT OUTER JOIN procedure\_record\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

/\* Joining the procedure location information. \*/

LEFT OUTER JOIN location\_information\_table

ON procedure\_record\_table.location\_information\_identification\_number =

location\_information\_table.location\_information\_id

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. Note that the system used to track anesthesia providers was updated and the data was moved to a new table, but the old data was never transferred. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM

( SELECT staff.procedure\_record\_identification\_number AS procedure\_case\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_identification\_number AS

anesthesia\_provider\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_role AS anesthesia\_provider\_role

FROM antiquated\_or\_staff\_table staff

JOIN antiquated\_staff\_table

ON antiquated\_staff\_table.procedure\_record\_identification\_number =

staff.anesthesia\_staff\_identification\_number

UNION

SELECT anesthesia\_review\_table.procedure\_record\_identification\_number AS

procedure\_case\_identification\_number,

staff.anesthesia\_specific\_prov\_identification\_number AS anesthesia\_provider\_identification\_number,

staff.anesthesia\_specific\_prov\_type\_description\_data AS anesthesia\_provider\_role

FROM anesthesia\_review\_table

INNER JOIN anesthesia\_specific\_staff staff

ON anesthesia\_review\_table.anesthesia\_specific\_episode\_identification\_number =

staff.staff\_description\_identification\_number

)

WHERE anesthesia\_provider\_role = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

WHERE measurement\_data\_table\_2.data\_item\_identification\_number = 'NERVE STIMULATION'

AND measurement\_data\_table.data\_entry\_value IS NOT NULL

/\* Case Inclusion criteria. A Case must have either:

1. An anesthesia start and start time and stop time with an anesthesia attending physician assigned

2. Must be a paper chart used in a setting with no EMR access

3. Must have recorded times such as: in room time, procedure start time, or recovery start time

4. Must have a surgeon with documented minutes in a case. \*/

AND

(

( anesthesia\_review\_table.anesthesia\_procedure\_start\_time IS NOT NULL

AND anesthesia\_review\_table.anesthesia\_procedure\_stop\_time IS NOT NULL

AND att\_count >=1

)

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM ANESTHESIA FLOOR'

OR location\_information\_table.location\_description = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%labor%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%cse%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%l'||chr(38)||'d%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%l '||chr(38)||' d%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%l and d%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%deliver%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%obstetric%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%section%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%c/s%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%cesarean%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%management%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%intubation%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%treatment%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%new start%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%epidural%'

OR anesthesia\_review\_table.procedure\_title\_description IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery.

Only surgical gynecologic cases included. \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM OBOR'

OR location\_information\_table.location\_description = 'RR OBOR'

)

AND

( lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%deliver%'

OR lower(anesthesia\_review\_table.procedure\_title\_description) LIKE '%cesarean%'

OR anesthesia\_review\_table.procedure\_title\_description IS NULL

)

)

)

GROUP BY anesthesia\_specific\_procedure\_record\_identification\_number, last\_train\_of\_four\_time, last\_train\_of\_four\_value;

/\* 4. Most Recent Creatinine Value. This query will report the unique case identification number, the value of most recent creatinine lab value prior to surgery, and the time that the value was recorded. Lab specimens resulting the morning of surgery are still included and lab values older than one year prior to surgery are ignored. \*/

-- VERSION I -– Code Written Using Case Based Tables

SELECT procedure\_case\_identification\_number, preop\_cr\_time, preop\_cr\_val

FROM case\_based\_preop\_metrics\_table;

-- VERSION II -– Code Written Using Base Tables

SELECT procedure\_case\_identification\_number, preop\_cr\_time, preop\_cr\_val

FROM

( SELECT procedure\_case\_list\_table.procedure\_case\_identification\_number,

first\_value(data\_entry\_time) OVER (

PARTITION BY procedure\_case\_list\_table.procedure\_case\_identification\_number

ORDER BY data\_entry\_time DESC,data\_entry\_value ASC) AS preop\_cr\_time,

first\_value(data\_entry\_value) OVER (

PARTITION BY procedure\_case\_list\_table.procedure\_case\_identification\_number

ORDER BY data\_entry\_time DESC,data\_entry\_value ASC) AS preop\_cr\_val

FROM procedure\_case\_list\_table

/\* Lab result criteria: anything before the case starts and greater than 365 days prior to the start of the case. \*/

LEFT OUTER JOIN lab\_result\_table

ON procedure\_case\_list\_table.patient\_unique\_identification\_number =

lab\_result\_table.patient\_unique\_identification\_number

AND trunc(lab\_result\_table.data\_entry\_time) <=

COALESCE(anesthesia\_procedure\_start\_time,procedure\_review\_date)

AND trunc(lab\_result\_table.data\_entry\_time) >=

(COALESCE(anesthesia\_procedure\_start\_time,procedure\_review\_date) - 365)

AND lab\_result\_specification\_identification\_number IN

('Creatinine\_1', 'Creatinine\_2','Creatinine\_3','Creatinine\_4','Creatinine\_5')

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM all\_staff\_table

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_case\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM anesthesia\_staffing\_table

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM mv\_schema\_case\_events

WHERE timestamp\_marker\_identifier IN

(in\_room\_identifier,procedure\_start\_identifier,recovery\_start\_identifier,paper\_chart\_identifier)

GROUP BY procedure\_case\_identification\_number

) event\_description

ON event\_description.procedure\_case\_identification\_number =

procedure\_case\_list\_table.procedure\_case\_identification\_number

WHERE

(

( anesthesia\_procedure\_start\_time IS NOT NULL

AND anesthesia\_procedure\_stop\_time IS NOT NULL

AND att\_count >=1

)

OR COALESCE(paper\_chart,in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes >0

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( ucla\_loc = 'SM ANESTHESIA FLOOR'

OR ucla\_loc = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(procedure\_title\_description) LIKE '%labor%'

OR lower(procedure\_title\_description) LIKE '%cse%'

OR lower(procedure\_title\_description) LIKE '%l'||chr(38)||'d%'

OR lower(procedure\_title\_description) LIKE '%l '||chr(38)||' d%'

OR lower(procedure\_title\_description) LIKE '%l and d%'

OR lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%obstetric%'

OR lower(procedure\_title\_description) LIKE '%section%'

OR lower(procedure\_title\_description) LIKE '%c/s%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR lower(procedure\_title\_description) LIKE '%management%'

OR lower(procedure\_title\_description) LIKE '%intubation%'

OR lower(procedure\_title\_description) LIKE '%treatment%'

OR lower(procedure\_title\_description) LIKE '%new start%'

OR lower(procedure\_title\_description) LIKE '%epidural%'

OR procedure\_title\_description IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery. Only surgical gynecologic cases included. \*/

AND NOT

(

( uprocedure\_case\_list\_tablea\_loc = 'SM OBOR'

OR uprocedure\_case\_list\_tablea\_loc = 'RR OBOR'

)

AND

( lower(procedure\_title\_description) LIKE '%deliver%'

OR lower(procedure\_title\_description) LIKE '%cesarean%'

OR procedure\_title\_description IS NULL

)

)

)

GROUP BY procedure\_case\_identification\_number, preop\_cr\_time, preop\_cr\_val;

-- VERSION III -– Code Written Using Only Clarity Tables

SELECT procedure\_record\_identification\_number, preop\_cr\_time, preop\_cr\_val

FROM

( SELECT procedure\_record\_table.procedure\_record\_identification\_number,

first\_value(lab\_result\_table.data\_entry\_time) OVER (

PARTITION BY procedure\_record\_table.procedure\_record\_identification\_number

ORDER BY lab\_result\_table.data\_entry\_time DESC,data\_entry\_value ASC) AS preop\_cr\_time,

first\_value(data\_entry\_value) OVER (

PARTITION BY procedure\_record\_table.procedure\_record\_identification\_number

ORDER BY lab\_result\_table.data\_entry\_time DESC,data\_entry\_value ASC) AS preop\_cr\_val

FROM procedure\_record\_table

LEFT OUTER JOIN anesthesia\_procedure\_times\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_procedure\_times\_table.procedure\_record\_identification\_number

AND anesthesia\_procedure\_times\_table.tracking\_event\_description\_data = in\_room\_marker

/\* During the validation phase of coding, it was discovered that the anesthesia\_review\_table was incomplete for many rows and had left a significant number of unmatched rows when joined on a single variable. For that reason, the table needs to be joined 3 times. The first join uses the anesthesia\_specific\_procedure\_record\_identification\_number. The second join uses the patient identification number and time of surgery. The last uses that patient identification number and the surgery date. Every subsequent join is more generalized and only matches rows that were unmatched previously. \*/

LEFT OUTER JOIN anesthesia\_review\_table

ON procedure\_record\_table.procedure\_record\_identification\_number =

anesthesia\_review\_table.anesthesia\_specific\_procedure\_record\_identification\_number

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_2

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_2.anesthesia\_specific\_patient\_unique\_identification\_number

AND anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time = COALESCE(

anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time)

LEFT OUTER JOIN anesthesia\_review\_table anesthesia\_review\_table\_3

ON COALESCE(procedure\_record\_table.patient\_unique\_identification\_number,

anesthesia\_review\_table.anesthesia\_specific\_patient\_unique\_identification\_number) =

anesthesia\_review\_table\_3.anesthesia\_specific\_patient\_unique\_identification\_number

AND trunc(anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) =

trunc(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_procedure\_times\_table.documented\_event\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time))

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT ct.procedure\_record\_identification\_number,

MAX(CASE

WHEN tracking\_event\_description\_data = recovery\_start\_identifier

THEN documented\_event\_time

END ) AS start\_recovery,

MAX(CASE

WHEN tracking\_event\_description\_data = in\_room\_identifier

THEN documented\_event\_time

END ) AS in\_room,

MAX(CASE

WHEN tracking\_event\_description\_data = procedure\_start\_identifier

THEN documented\_event\_time

END ) AS procedure\_start\_time

FROM procedure\_review\_case\_times ct

WHERE tracking\_event\_description\_data IN (in\_room\_identifier,procedure\_start\_identifier,

recovery\_start\_identifier)

GROUP BY procedure\_record\_identification\_number

) event\_description

ON event\_description.procedure\_record\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Joining the raw lab results tables. \*/

LEFT OUTER JOIN lab\_result\_table

ON lab\_result\_table.patient\_unique\_identification\_number =

procedure\_record\_table.patient\_unique\_identification\_number

/\* More specific information about the lab results in the table above. \*/

LEFT OUTER JOIN lab\_result\_specification\_table

ON lab\_result\_table.lab\_result\_specification\_identification\_id =

lab\_result\_specification\_table.lab\_result\_specification\_identification\_id

/\* Lab result criteria: anything before the case starts and greater than 365 days prior to the start of the case. \*/

LEFT OUTER JOIN procedure\_record\_table

ON procedure\_record\_table.order\_procedure\_identification\_number =

lab\_result\_table.order\_procedure\_identification\_number

AND trunc(lab\_result\_table.data\_entry\_time) <= COALESCE(COALESCE(

anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time),

COALESCE(trunc(in\_room),trunc(procedure\_start\_time),

trunc(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time)),trunc(start\_recovery),surgery\_date))

AND trunc(lab\_result\_table.data\_entry\_time) >= (COALESCE(COALESCE(

anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time),

COALESCE(trunc(in\_room),trunc(procedure\_start\_time),

trunc(COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time)),trunc(start\_recovery),surgery\_date))

- 365)

AND lab\_result\_table.lab\_result\_specification\_identification\_number IN

('Creatinine\_1', 'Creatinine\_2','Creatinine\_3','Creatinine\_4','Creatinine\_5')

/\* Sum of all the minutes for the primary surgeon assigned to a procedure. This is a prerequisite to classify as a case is having at least some surgeon minutes logged. \*/

LEFT OUTER JOIN

( SELECT procedure\_record\_identification\_number,sum(duration\_in\_minutes) AS surg\_minutes

FROM procedure\_review\_all\_staff\_table astaff

LEFT OUTER JOIN improved\_specification\_position\_table improved\_specification\_position

ON improved\_specification\_position.role\_description\_data = astaff.position\_description

WHERE physicianesthesia\_specific\_provider\_position = 1

AND physicianesthesia\_specific\_provider\_position\_2 IN ('PRIMARY','PROCEDURALIST')

GROUP BY procedure\_record\_identification\_number

) all\_staff\_data

ON all\_staff\_data.procedure\_record\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

/\* Calculating all the time events that occur during a case that help solidify that it occurred. This is one of the inclusion criteria for a procedure. \*/

LEFT OUTER JOIN

( SELECT encounter\_identification\_number,

MIN(CASE

WHEN timestamp\_marker\_identifier = in\_room\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS in\_room,

MIN(CASE

WHEN timestamp\_marker\_identifier = procedure\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS procedure\_start,

MIN(CASE

WHEN timestamp\_marker\_identifier = recovery\_start\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS recovery\_start,

MAX(CASE

WHEN timestamp\_marker\_identifier = recovery\_stop\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS stop\_recovery,

MAX(CASE

WHEN timestamp\_marker\_identifier = paper\_chart\_identifier

THEN timestamp\_marker\_time

ELSE NULL

END) AS paper\_chart

FROM event\_description\_table

LEFT OUTER JOIN event\_description\_table\_2

ON event\_description\_table\_2.event\_identification\_number =

event\_description\_table.event\_identification\_number

WHERE timestamp\_marker\_identifier IN (in\_room\_identifier,procedure\_start\_identifier,

recovery\_start\_identifier,recovery\_stop\_identifier,paper\_chart\_identifier)

GROUP BY encounter\_identification\_number

) event\_description

ON COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_2.anesthesia\_procedure\_identification\_number,

anesthesia\_review\_table\_3.anesthesia\_procedure\_identification\_number) =

event\_description.encounter\_identification\_number

/\* Joining the procedure location information. \*/

LEFT OUTER JOIN location\_information\_table ON

procedure\_record\_table.location\_information\_identification\_number =

location\_information\_table.location\_information\_identification\_number

/\* Counting the number of anesthesia attendings that are assigned to the case. This is a prerequisite to classify as a case is having at least 1 provider assigned. Note that the system used to track anesthesia providers was updated and the data was moved to a new table, but the old data was never transferred. \*/

LEFT OUTER JOIN

( SELECT procedure\_case\_identification\_number,count(anesthesia\_provider\_identification\_number)

AS att\_count

FROM (

SELECT staff.procedure\_record\_identification\_number AS procedure\_case\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_identification\_number AS

anesthesia\_provider\_identification\_number,

antiquated\_staff\_table.anesthesia\_provider\_position AS anesthesia\_provider\_position

FROM antiquated\_or\_staff\_table staff

JOIN antiquated\_staff\_table

ON antiquated\_staff\_table.procedure\_record\_identification\_number =

staff.anesthesia\_staff\_identification\_number

UNION

SELECT anesthesia\_review\_table.procedure\_record\_identification\_number AS

procedure\_case\_identification\_number,

staff.anesthesia\_specific\_prov\_identification\_number AS anesthesia\_provider\_identification\_number,

staff.anesthesia\_specific\_prov\_type\_description\_data AS anesthesia\_provider\_position

FROM anesthesia\_review\_table

INNER JOIN anesthesia\_specific\_staff staff

ON anesthesia\_review\_table.anesthesia\_specific\_episode\_identification\_number =

staff.staff\_description\_identification\_number

)

WHERE anesthesia\_provider\_position = 'ANESTHESIA ATTENDING'

GROUP BY procedure\_case\_identification\_number

) anesthesia\_staffing

ON anesthesia\_staffing.procedure\_case\_identification\_number =

procedure\_record\_table.procedure\_record\_identification\_number

WHERE

(

( COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_start\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_start\_time) IS NOT NULL

AND COALESCE(anesthesia\_review\_table.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_2.anesthesia\_procedure\_stop\_time,

anesthesia\_review\_table\_3.anesthesia\_procedure\_stop\_time) IS NOT NULL

AND att\_count >=1

)

OR COALESCE(paper\_chart,event\_description.in\_room,procedure\_start,recovery\_start) IS NOT NULL

OR surg\_minutes >0

)

/\* Procedure must not be a floor procedure (excluded from procedure list). \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM ANESTHESIA FLOOR'

OR location\_information\_table.location\_description = 'RR ANESTHESIA FLOOR'

)

/\* Excluding non-operative cases and deliveries. \*/

AND

(

lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%labor%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cse%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l'||chr(38)||'d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l '||chr(38)||' d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%l and d%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%deliver%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%obstetric%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%section%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%c/s%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cesarean%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%management%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%intubation%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%treatment%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%new start%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%epidural%'

OR COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description) IS NULL

)

)

/\* Remove all cases from labor and delivery operating room that have to do with a delivery. Only surgical gynecologic cases included. \*/

AND NOT

(

( location\_information\_table.location\_description = 'SM OBOR'

OR location\_information\_table.location\_description = 'RR OBOR'

)

AND

( lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%deliver%'

OR lower(COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description)) LIKE '%cesarean%'

OR COALESCE(anesthesia\_review\_table.procedure\_title\_description,

anesthesia\_review\_table\_2.procedure\_title\_description,

anesthesia\_review\_table\_3.procedure\_title\_description) IS NULL

)

)

)

GROUP BY procedure\_record\_identification\_number, preop\_cr\_time, preop\_cr\_val;