ALE Days

SIS Table Name	ALE Days
SIS table Code	AL
Data Collection LEA Level	School
Cycles	3,5,6,7
# of Data Items	9
Source	eSchool

Source

These fields receive their values directly from eSchool table fields:

Column ID	Element	Column Name	Cycles	Source	Source Column
AL0030	LEA	lea	3,5,6,7	eSch: REG_BUILDING	state_code_equiv
AL0040	SSN	ssn	3,5,6,7	eSch: REG_PERSONAL	SSN
AL0045	State Reporting ID	uniq_stu_id	3,5,6,7	eSch: REG	student_id
AL0050	ALE Entry Date	entry_date	3,5,6,7	eSch: REG_PROGRAMS	start_date
AL0055	ALE Exit Date	exit_date	3,5,6,7	eSch: REG_PROGRAMS	end_date
AL0060	Consecutive Days Flag	consec_flag	3,5,6,7	Derived field (See below)	
AL0065	Number of Days in ALE	ale_days	3,5,6,7	eSch: ATT_STU_DAY_MEMB	membership_value
AL0070	Special Ed Status	sped_status	3,5,6,7	eSch: REG_PROGRAM_COLUMN	state_code_equiv
AL0080	ALE Program Type	ale_type	3,5,6,7	Derived field (See below)	

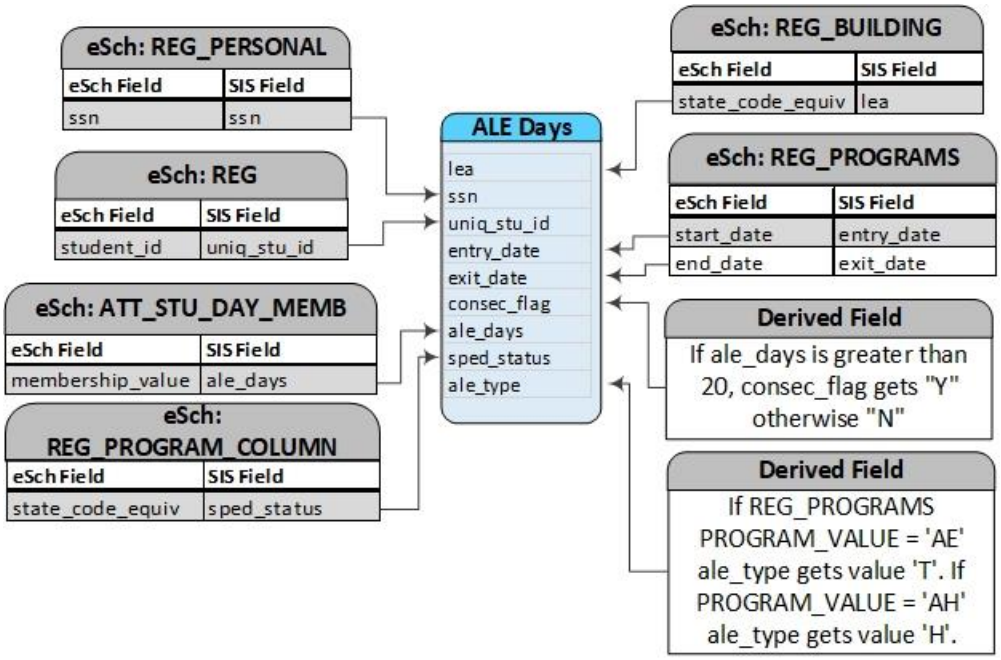
Derived Fields

Consecutive Days Flag

Field ID “AL0060”, Element “Consecutive Days Flag”, Column Name “consec_flag”, Cycles 3,5,6,7.
If AL0065 Number of Days in ALE is greater than 20, AL0060 gets “Y” otherwise “N”.

ALE Program Type

Field ID “AL0080”, Element “ALE Program Type”, Column Name “ale_type”, Cycles 3,5,6,7.
If student has a record in REG_PROGRAMS with PROGRAM_ID = 'ARCUR': If PROGRAM_VALUE = 'AE' this field gets value ‘T’ indicating Traditional ALE Program. If PROGRAM_VALUE = 'AH' this field gets value ‘H’ indicating Hybrid ALE Program.



Selection Criteria

```
FROM
  REG r
  INNER JOIN REG_PERSONAL rp
      ON REG.district = REG_PERSONAL.district
      AND REG.student_id = REG_PERSONAL.student_id
   INNER JOIN REG_PROGRAMS p
      ON REG.district = REG_PROGRAMS.district
      AND REG.student_id = REG_PROGRAMS.student_id
   INNER JOIN REG_BUILDING rb
      ON REG.district = REG_BUILDING.district
        AND REG.building = REG_BUILDING.building
        AND REG_BUILDING.state_code_equiv IS NOT NULL
        AND (last three characters of REG_BUILDING.state_code_equiv != "000")
        AND REG_BUILDING.building < 900
   INNER JOIN REG_CALENDAR cal
      ON REG_CALENDAR.district = REG.district
      AND REG_CALENDAR.building = REG.building

   INNER JOIN ATT_INTERVAL ai
       ON REG_BUILDING.building = ATT_INTERVAL.building
       AND ATT_INTERVAL.school_year = REG_CALENDAR.school_year
       AND ATT_INTERVAL.ATND_INTERVAL = Current Quarter
       AND ATT_INTERVAL.SUMMER_SCHOOL <> "Y"
WHERE
    REG_CALENDAR.school_year = Current Year
    AND REG_PROGRAMS.summer_school = "N"
```

```
    AND REG_PROGRAMS.program_value = "AE"
    AND REG_PROGRAMS.start_date != REG_PROGRAMS.end_date (or current date if
REG_PROGRAMS.end_date is null)
    AND REG.current_status = "A" OR "I"
    AND REG_PROGRAMS.program_id = "ARCUR"
    AND REG_PROGRAMS.start_date between REG_CALENDAR.first_day and REG_CALENDAR.last_day

ALEDays calculation:
SELECT SUM(membership_value)
FROM
   ATT_STU_DAY_MEMB m
WHERE
    ATT_STU_DAY_MEMB.district = REG.district
    AND ATT_STU_DAY_MEMB.school_year = REG_CALENDAR.school_year
    AND ATT_STU_DAY_MEMB.student_id = REG.student_id
    AND ATT_STU_DAY_MEMB.membership_date BETWEEN REG_CALENDAR.start_date AND
REG_CALENDAR.end_date
    AND ATT_STU_DAY_MEMB.building = REG.building
```