

## Miscellaneous Conversion Information

### Inactive SASI Students are not converted

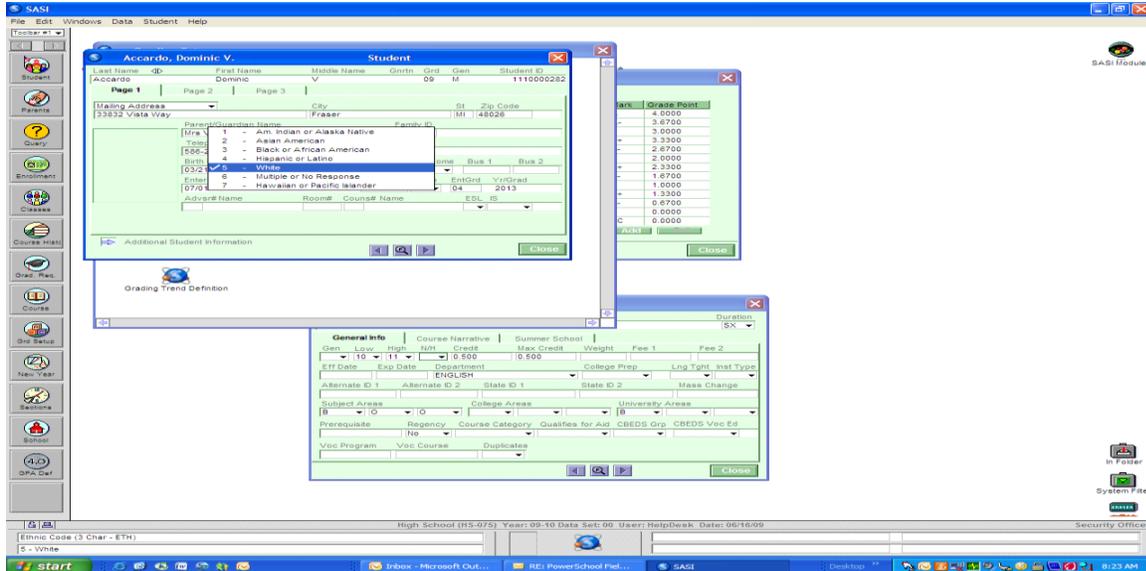
#### Time commitment:

- A significant amount of your summer will be filled with: converting data, importing data, validating data, scheduling training, attending training, leading training.

#### Ethnicity

- We decided to setup the ethnic codes 1 through 6 in PowerSchool to match the MSDS manual. Since we had a Multi-racial code in SASI, we decided to use 7 as Multi-racial and you will have to add this code into PowerSchool.
- Brian wrote conversion program to transfer codes as follows:

- 1 → 1 American Indian or Alaska Native
- 2 → 2 Asian American
- 3 → 3 Black or African American
- 4 → 6 Hispanic or Latino
- 5 → 5 White
- 6 → 7 Multiple or No Response
- 7 → 4 Hawaiian or Pacific Islander



## Pre-Conversion Cleanup and Prep in SASI

### SASI Data

- Have your most current SASI Data for importing.
- Reorganize all files
- Delete orphan records
- Delete temp files
- It is best to just copy your entire DATAFILE folder for the conversion process
- Student count - get a student count in SASI by running STU98 or a query. Remember to include inactive students. This count will allow you to compare numbers before and after import.

### SASI Teacher File

- Must contain teachers employee number. This is necessary for Data Director.
- Fix duplicate teachers. For example AM/PM kindergarten, have to change the teacher number in the section.

### SASI Courses and Sections

- Powerschool has one district file, so duplicate courses in each of your schools will have to be addressed.
- Brian has written a SASI enhancement that will add the prefix HS MS to a course – this is optional. This enhancement program can be ran in the rolled over year in SASI.
- The conversion program also has a built-in option that will add a prefix of your choice to a course number – this is also optional.
- Course names cannot contain any punctuation; examples - / &. Courses will not import with special characters.  
Course ID examples: Sasi – HS1001      PowerSchool – HS1001  
Section ID examples: Sasi – 0027PE9      PowerSchool - .1
- PowerSchool sections cannot contain leading 0's and cannot contain any alpha characters. Our conversion program will remove these characters.
- If students in SASI only had a teacher number on the Student screen then you will need to create courses and sections in SASI – one AM section (period 1) and one PM section (period 2). Note: you cannot use period 0 in PowerSchool. We found this problem by running ATT701x which showed period 2 all greyed out.
- If students were scheduled into only one course such as 1<sup>st</sup> grade, 2<sup>nd</sup> grade, etc., for the whole day, then you need to duplicate that course and create a second period. This is easy to do using the Duplicate Section option in Sections. After 2<sup>nd</sup> period section is created, you can open both sections, select all students and drag those students from the original section to the newly created section. Don't worry about the class start dates as the conversion program will use the enter date from the Student screen for this date.
- Print a MST03 before changes.
- Delete sections that have no students scheduled
- Course numbers with characters example 00K+ and changed to 00KP
- Max class sizes of zero
- Update class seat counts (Section atom)
- Print MST03 after changes

- CLS19 – students with less than a full schedule. Ran again after done schedule for less than two classes

### SASI Scheduling Tab in School Atom

- Must set to course and manual.
- Length field must be set

### SASI Rotation Schedules

- The conversion program now allows for schedules with rotation days. It will allow the user to select the proper start date for the rotation day(A or B for example) based on the term code. Example – A day starts on 9/9/09 for first semester and B day starts on 9/10/09. A day starts on 1/7/2010 for second semester and B day starts on 1/8/2010.
- Once these are coded, all rotating schedules will import correctly.
- Must set to course and manual.
- Length field must be set

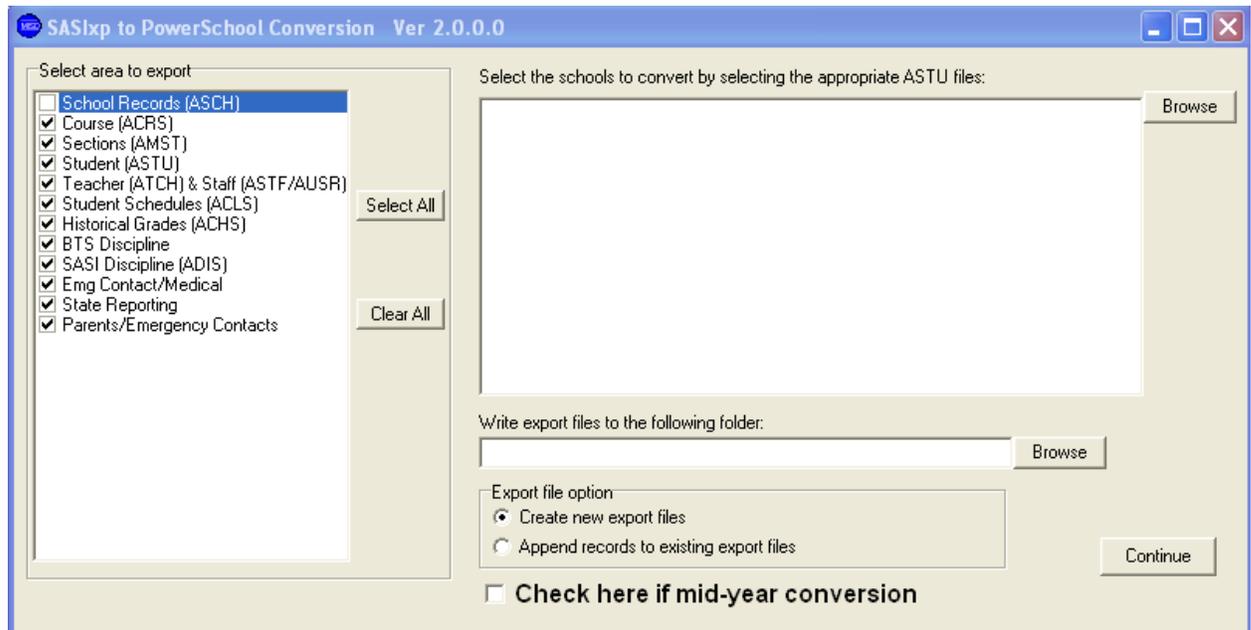
### Mid-Year Elementary Conversion Checklist – Utica and MISD Schools (the changes listed below were done in a copy of the current school year in SASI)

- In PowerSchool we have to have an AM and PM period in order for PowerTeacher to work and for our attendance conversion program to work.
- Teacher File
  - Teacher records must contain teachers employee number for Data Director
  - Fix duplicate teachers. For example AM/PM kindergarten, have to change the teacher number in the section.
- School Atom
  - expand section number to 5 characters (depends on length of section #s)
  - expand section number to 5 characters (depends on length of section #s)
  - period range should be set to 0-2. Having a range of 0-3 can cause period 2 not to convert (the spreadsheet will only have a 1 under Period Abbrev and no 2s.
- Sections/Courses/Schedules
  - If students in SASI only had a teacher number on the Student screen then you will need to create courses and sections in SASI – one AM section (period 1) and one PM section (period 2). Note: you cannot use period 0 in PowerSchool.
  - If students were scheduled into only one course such as 1<sup>st</sup> grade, 2<sup>nd</sup> grade, etc., for the whole day, then you need to duplicate that course and create a second period. This is easy to do using the Duplicate Section option in Sections. After 2<sup>nd</sup> period section is created, you can open both sections, select all students and drag those students from the original section to the newly created section. Don't worry about the class start dates as the conversion program will use the enter date from the Student screen for this date.
  - Print a MST03 before changes.

- Delete sections that have no students scheduled
  - Course numbers with characters example 00K+ and changed to 00KP
  - Max class sizes of zero
  - Update class seat counts (Section atom)
  - Print MST03 after changes
  - CLS19 – students with less than a full schedule. Ran again after done schedule for less than two classes
- 
- Run query from each elementary school to identify students who have multiple lines of enrollment within the same school year. We need to look for students who exited and re-entered, not those with teacher changes. Use this query: ASTU, AENR, fn In effdate, transyear, if transyear = 09 and \$count > 2. The enter dates for these students will have to be change on the spreadsheet otherwise attendance will not be converted for the entire year. This will cause the Transfer Info screen to not be accurate so you will have to correct each student by transferring student out and back in.

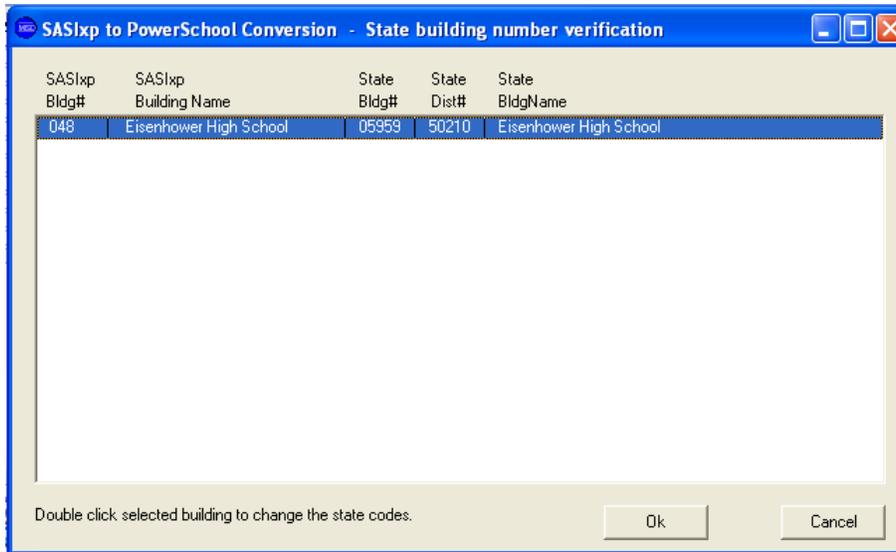
## Running The Conversion Program

- Open the cvtMain.exe program.

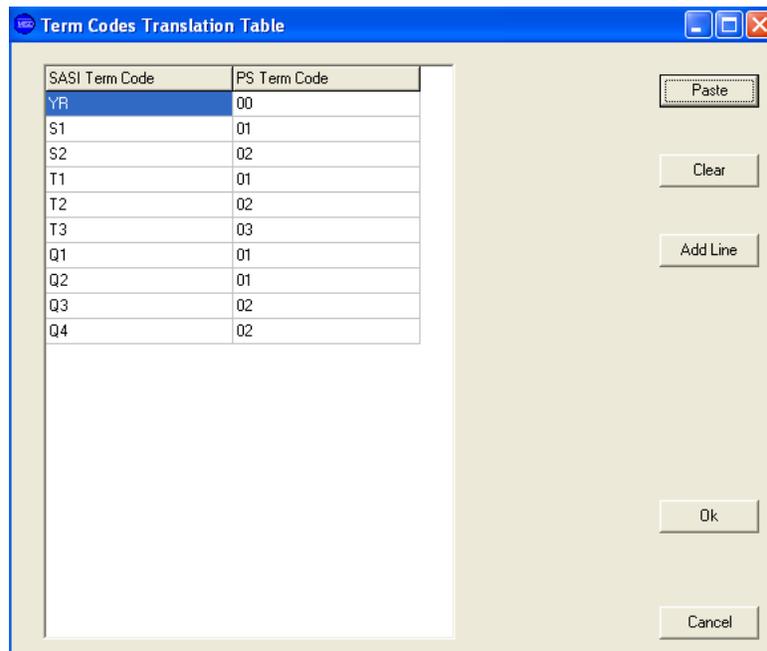


- Select areas to export. DO NOT choose School Records (ASCH). This information was keyed in manually to PowerSchool. If you are not sure about some areas, it's okay to select them anyway since the program will not convert data if it does not find data in SASI. You will get an error message later on stating that there was no data to convert.
- Select the school to convert by browsing to the SASI data and selecting the appropriate ASTU files. For example, ASTU9075 for the year 2009-2010 and the school 075. Note about converting multiple schools at once. Although the conversion program allows you to select multiple schools at once, it creates one large file which is difficult to convert from since conversions at the District level in PowerSchool do not work well. We found the best practice to be to select each school's ASTU individually and then import that file at the School level in PowerSchool.
- Write export files to a converted data folder: (browse to converted data folder)
- Select Create new export files option unless you forgot to choose an area to export which then you will select the Append records to existing export files.
- If doing a mid-year conversion, be sure to select the mid-year conversion checkbox.
- Click Continue.

Verify State building numbers.

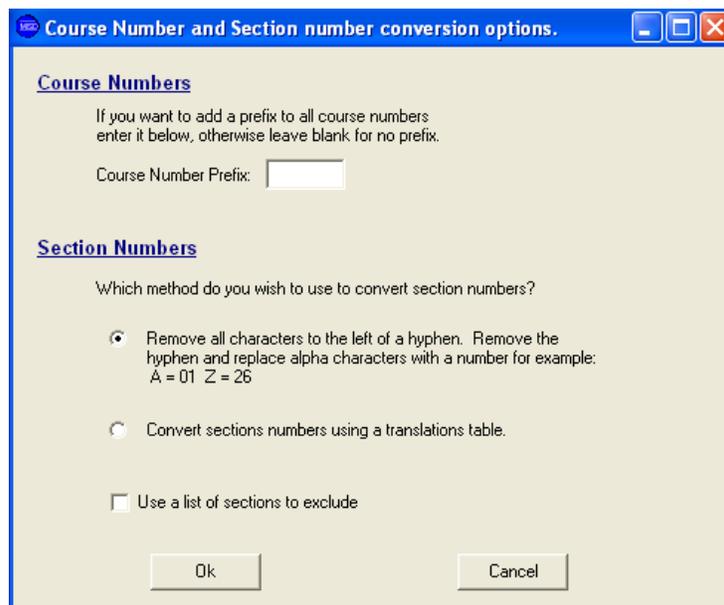


Choose the PowerSchool Term Codes. This school has Semesters & Quarters and so we removed the trimesters and associated the correct PS Term Code. If we were choosing Trimesters and not Semesters & Quarters, they would be: T1 for termcode 01; T2 for termcode 02; T3 for termcode 03 and remove the Semesters & Quarters. Important: Do not change YR=00 to Y1 or YR since 00 is a numerical representation for 1900. If you are converting multiple schools such as middle/jr and high schools at one time, you have to include all appropriate term codes.

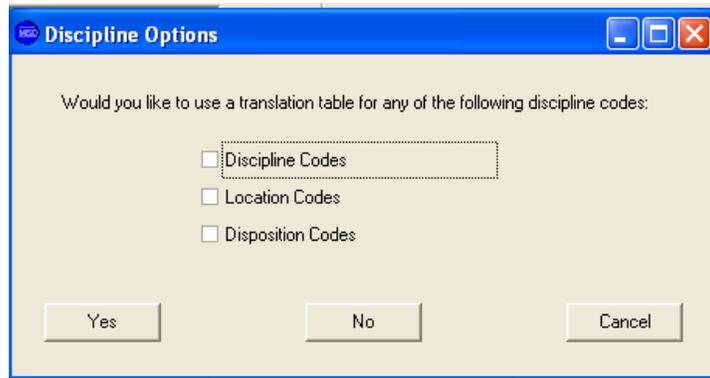


Course Numbers – since PowerSchool has one big district course file, some liaisons wanted to identify their courses by adding an HS before the course number for their high schools, an MS for their middle and an EL for their elementary schools. Some liaisons made this change in SASI, others made the change in the spreadsheet. If you wish to use this conversion program to put a prefix before the course number you can choose it here.

Section Numbers – the conversion program exports SASI’s courses and sections together so most liaisons chose to use the Remove all characters... option. Brian wrote other options to help convert section number since PowerSchool does not like leading zeroes. Make the selection that best appropriate for your school data.

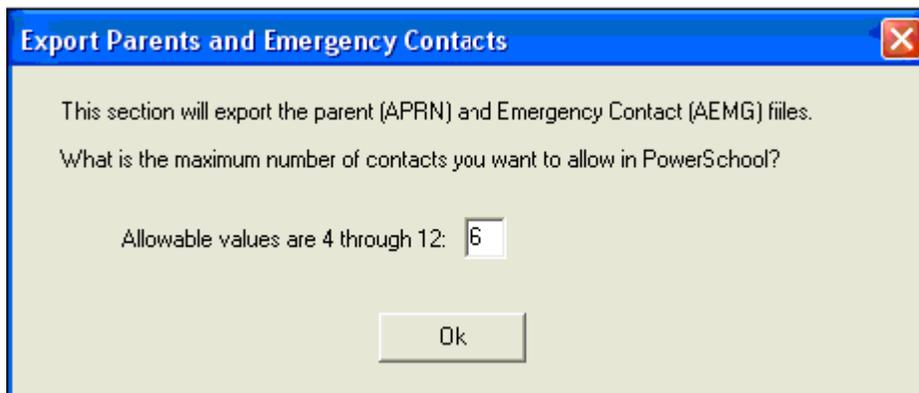


If you wish to create a translation table for these Discipline Options select it here.

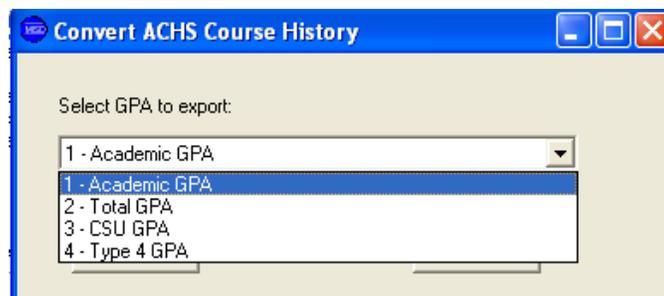


This section will export the parent (APRN) and Emergency Contact (AEMG) files. The CVTMAIN program reads the APRN first then looks at the AEMG and compares and tries to only take one record. It will sort mother, then father.

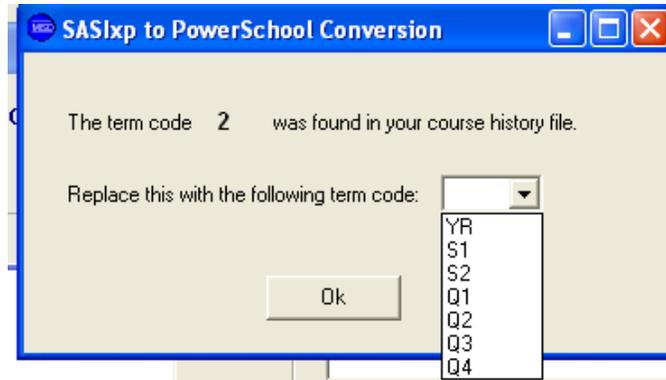
Need to determine the maximum number of contacts you want to allow in PowerSchool. Most of the time we chose 6.



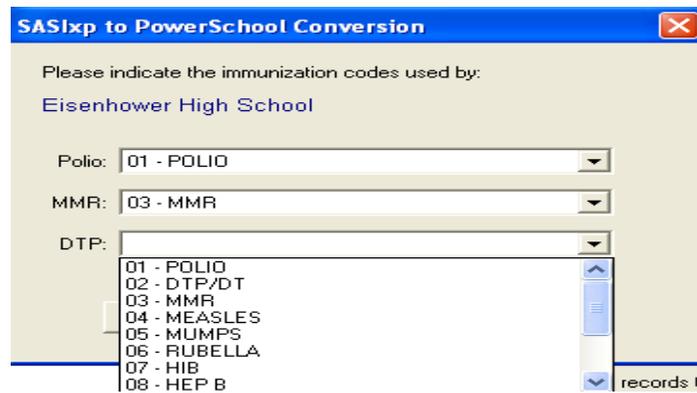
If you are converting Course History you will get the following screen.



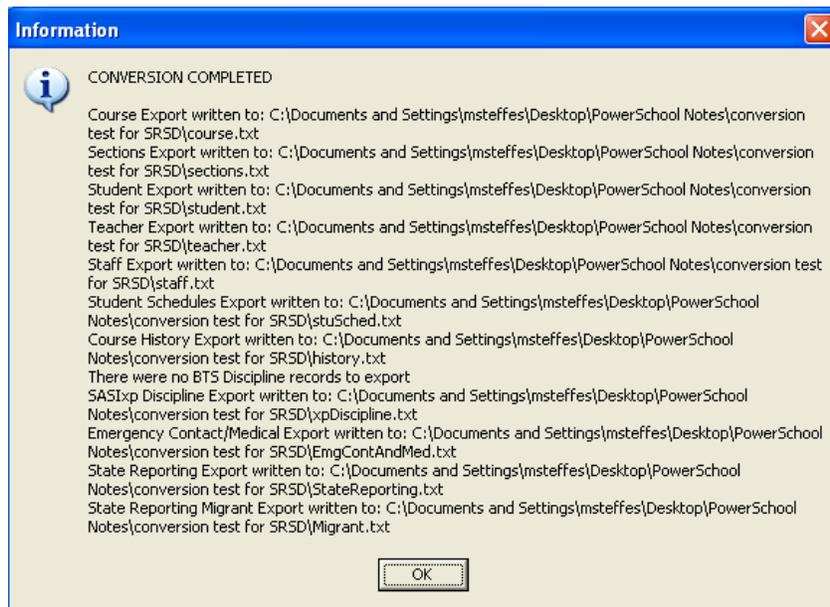
Your files will begin to process; subsequently this screen will ask which GPA to export.



Choose the immunization codes for each of the 3. Note: not many people chose to convert Immunizations since they keep this data in MCIR, the Health Dept's site.



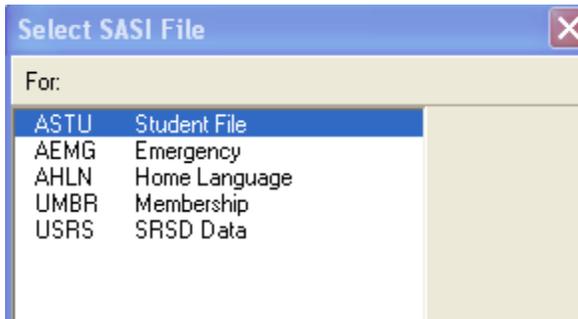
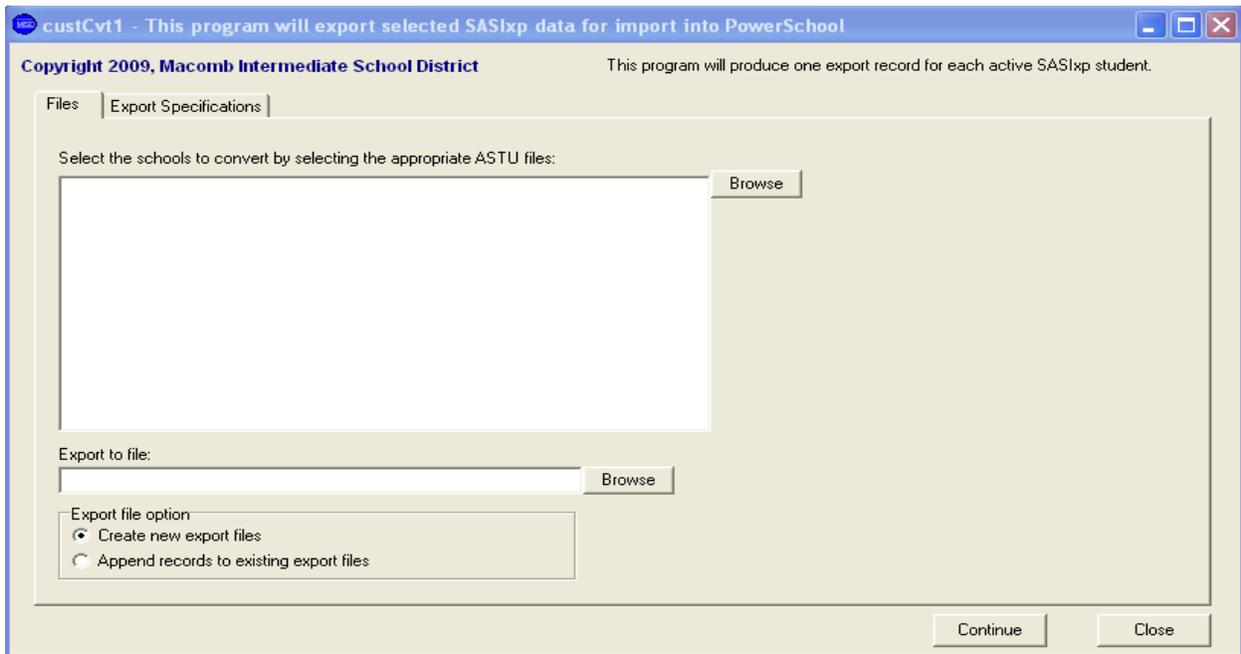
The program gives you a report on what was just converted.



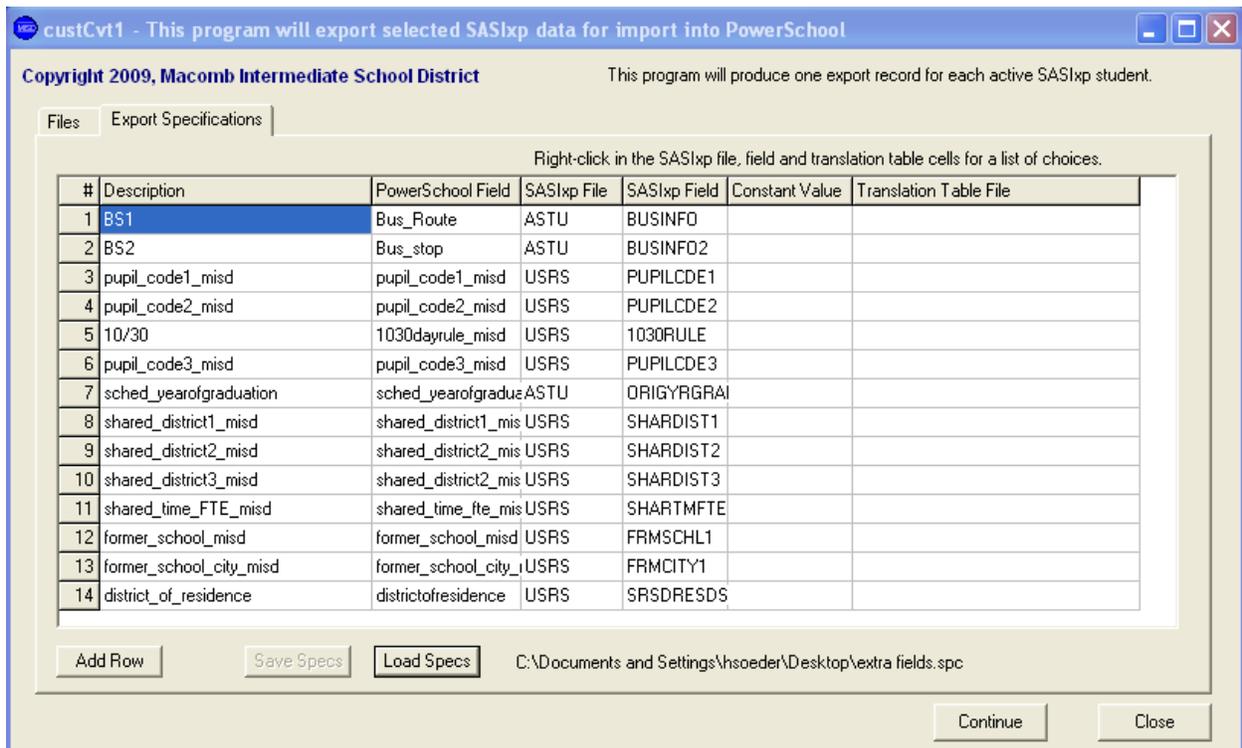
Here is an example of the files that were created and you will import into PowerSchool.

- btsDiscipline.txt (BTS Discipline)
- course.txt (Course ACRS)
- EmgContAndMed.txt (Emg Contact/Medical)
- Migrant.txt (State Reporting)
- Parent.txt (Parents)
- sections.txt (Sections AMST),
- staff.txt (Teacher (ATCH) & Staff (ASTF/AUSR)
- StateReporting.txt (State Reporting)
- stuSched.txt (Student Schedules (ACLS),
- teacher.txt (Teacher (ATCH) & Staff (ASTF/AUSR)
- xpDiscipline (SASI Discipline (ADIS),
- history.txt (ACHS)

**Running The custCvt1 Conversion Program to convert additional fields from SASI's ASTU, AEMG, AHLN, UMBR, USRS files. These would be fields that are not converted with the cvtMain.exe program.**



The screenshot below is an example of the fields that we converted for our ISD schools.



**Important!**

Some of these fields had a table relation in SASI so you have to change data in the spreadsheet prior to importing.

For example, the 10/30 day rule information exports as numerical value of either a 10 or a 30.

In PowerSchool, this field information is selected from a drop-down menu as 10 Day Rule.

So you have to change 10 to 10 Day Rule and 30 to 30 Day Rule.

This is the same for Pupil Codes. The data converts a RS, but you have to change the data in the spreadsheet to be RS-Reduced Schedule.

After adding rows and choosing all your field information, be sure and select Save Specs so you can Load at a later time when converting other schools.

## Post-Conversion Cleanup in the Excel Spreadsheet

### Excel Spreadsheet Formatting

- Once the data is converted you can pull it into excel and check your data prior to importing the data into PowerSchool. After creating your data in excel the file must be saved as a text file or it will not import into PowerSchool. See the PowerSchool Importing Directions documentation for details.
- If there are leading zeros in the text file and you open it in Excel, then save it, the leading zeros will be removed. This will create issues on the matching of student data. You will need to make the column a text column, then do a search and replace to keep the leading zeros.  
Ex: Find 400 and replace with '0400. The single quote must be there to retain the lead zero.
- Spreadsheet formatting – check your converted data and make sure that your excel spreadsheet is saved as a text file to import the data.

### Student Count

- check lines in spreadsheet to see if number matches numbers from SASI. Note: depending on the spreadsheet you are looking at not all rows in the spreadsheet will match since some conversions result in multiple records for students.

### Entry Date – student.txt - shows in the Transfer Info screen in PS

- We had a lot of problems with ADA/ADM reports because this field converted with a 7/1/2009 date as it was pulled from SASI's Enrollment History record and PowerSchool needs this date to be the first day of school. Carefully check this date on the spreadsheet to avoid having to mass change later in PS.

### Exit Date – student.txt - shows in the Transfer Info screen in PS

- This date pulled from the last day active school day in the SASI calendar. In PowerSchool we decided to use the day before count (June 29) so we mass changed that date.

### Expressions

- Watch for period zero. The periods that are zero will not import if the school does not have a zero period in their bell schedule in PowerSchool.
- Elementary setup - - in SASI most elementary schools periods were setup as 0 period for AM, 1<sup>st</sup> period for lunch, and 2<sup>nd</sup> period for PM. This setup was necessary in order for our enhancements to work. In PowerSchool the setup for an elementary should be: 1<sup>st</sup> period=period 1, leave time gap for lunch, 2<sup>nd</sup> period=period 2.
- If the class meets am and pm at the elementary the expression should be: 1-2(A), based on the periods being 1 and 2. Track schedules would be: 1(A) 2(B) for example.
- In PowerSchool we do not use a 0 period. In the section.txt spreadsheet, you may have to change expression 0(A) to 1(A) but 2(A) should be okay as is. When they attend for 2 periods the expression in the section should say 1-2(A). This would happen if you have the same section for

AM and PM, same teacher teaches all day. Kindergarten AM would be 1(A), Kindergarten PM would be 2(A).

## Grades Level

- Many Pioneer schools chose to use -6 for their grade 30 students because they didn't want to open their PowerSchool grade range to 30. PowerSchool will not import student data if the associated grade level in the Grade\_Level field is less than negative two (-2). This issue affects Quick Import, Import Using Template, AutoComm. If you chose to use -6 for your Early Childhood grade of 30, keep in mind that the conversion program will convert a grade 30 to a -6 but PowerSchool won't import a -6. So you need to mass change the -6 in the spreadsheet to a grade level that does not normally exist in the school building. You may have to add that grade to the school setup. After the import you can do a mass change to change the grade from (?) to -6. Remember to remove the (?) grade range in the school setup.
- MISD Schools - we chose to use grade 30 as a normal grade in PowerSchool since our schools' use it so heavily. Since the conversion program changes grade 30 to -6, we need to check the Student.txt spreadsheet prior to import in order to change -6 back to 30.

## Next\_School (student.txt)

- For schools with grade level of 30 the conversion program assigns next school of 999999. Need to change this to the same school so student doesn't get put in the graduated school during the EOY process. Or...this can be done using Student Field Value change in PowerSchool.

## State Reporting

- Column width must be less than 200 columns. Need to break the import into 2 imports.

## Student Renumbering

- If student numbers are too large in SASI it will be necessary to assign students new numbers in PowerSchool. There is an option to assign new numbers during the Importing function. This process works well if you are importing all students at once and not doing a mid-year conversion.
- PowerSchool also has a renumbering option named Assign New Student Numbers which replaces student numbers with new ones. This is how L'Anse Creuse was renumbered but Tom had to export and then reimport in order to preserve the previous numbers .

- UTICA RENUMBERING - Because Utica took their schools in phases and wanted to be sure and preserve the students' previous SASI numbers for 3<sup>rd</sup> party software, they chose to assign numbers at the spreadsheet level. See steps below.
  - Rename column heading Student\_Number to prevstudentID
  - Add a column after SchoolID and name the header Student\_Number.
  - Go into PowerSchool District Office, select ALL students and use DDE to get the highest student number.
  - Pad by 2000
  - Key that number into the first cell of the Student\_Number column.
  - In the next cell, key in the next student number. Now that you have two numbers keyed in, you've established a pattern in Excel.
  - Select both cells and copy.
  - Use the handle and drag down for remaining students. Numbers should get incrementally larger.
  - Save the Student.txt spreadsheet and this will be the very first spreadsheet to import.
  - Before you can import other "slave file" spreadsheets (those that are student related), you have to export the students from PowerSchool using DDE. See directions under the Pre Import Prep in PowerSchool below.

## Pre-Import Prep in PowerSchool

### Years and Terms

- There can be no overlapping of dates. Term must begin on a Monday or must be on an in-session day – check the calendar!!!
- In order to import Sections and Student Schedules, you must have the Import File Term # field filled in. Go to School, Years & Terms, Edit Terms, click on each year/semester/etc link, and fill in the Import File Term # field.
- Need to know your calendar
- Years / Semesters / Quarters / Trimesters / Minis?
- In PowerSchool terms are alpha numeric. For example the 2009-2010 has year long, semester, and quarter classes. Therefore the terms are setup as:  
Y1 = 1900, S1 = 1901, S2 = 1902, Q1 = 1903, Q2 = 1904, Q3 = 1905, Q4 = 1906

### Grade Levels

- If you are using negative numbers such as -6 or a higher number such as 30 in your school's grade range you will have to enter in the number in an opposite manner. For example if you key in -6 to 6 and Submit, the numbers will disappear when you go back in the screen. You need to key in 6 to -6 and click Submit. The program will automatically transpose the numbers for you. This seems to be a quirk in the system, but it works!

### Calendar

- Should be setup as complete as possible.
- Mid-year – make sure the calendar matches the SASI calendar.
- School start date - must be first of enrollment

### Bell Schedule

- Must have a bell schedule assigned to the calendar in order for schedules to show in the bell schedule view.
- Elementary schools (also for mid-year). Make sure set to 1 and 2, not 0 and 2 like SASI's bell schedules.

### Periods/Expressions

- Should be setup beginning period 1, since PowerSchool doesn't like period zero. Periods 1 & 2 for AM and PM elementary schools.
- Periods & Days are called expressions in PowerSchool. A 1<sup>st</sup> hour that meets everyday will look like 1(A)

## Special Programs

- Need to start thinking about buildings with special programs – different calendars / different bell schedules / periods.

## Scheduling Considerations

- Rotations
- Days
- Teams
- Houses
- Blocks – what does a block mean to you?

## Attendance Preference Setup

- PowerSchool is setup very differently than SASI. Many reports such as state reports and our custom attendance letters are dependent on how attendance preferences are set. We have detailed attendance setup documentation – please refer to this when setting up attendance.
- All schools are set up as Meeting Attendance – even elementary level buildings.

## Attendance Conversions

- If the elementary school is using periods, but wants to take attendance only AM and PM, this can be accomplished by setting the preferred periods to count for ADA in the bell schedule and removing this option from the rest of the periods.
- It is also important to set up the Attendance Conversion for a full day to be based on a 2 period day. This would mean that if the student is present two periods, it would be worth a full FTE. If the student is present on period, it is worth a 0.5 FTE. And if the student is absent 2 or more periods, then it is worth zero FTE.

## Final Grading Setup

- Has to include all your grade reporting segments in order for progress reports, report cards, and the teacher gradebook to work correctly. Please see our documentation on how to setup.

## Ethnicity

- In PowerSchool have to add 7 Multi-racial

## Test Scores

- Tests need to be entered. Test scores can be imported (see documentation for this process) clean as possible.

## Student Contacts Screens

- Need to initialize all the fields before import screen makes sense when importing parent.txt file.

## Utica's Renumbering Process

- Import the students (Student.txt) spreadsheet ONLY.
- After student import, go to the Start Page and select ALL students.
- Go to DDE to export the students.
- Make sure Current Table: is Students (1)
- Select all records in this table
- Click on Search only in records belonging to (name of school)
- Search within the current ??? records only
- Export Records
- The Student table should automatically be selected
- Key in ID, Student\_Number, prevstudentid (see screenshot)
- Select Record Delimiter CRLF
- Submit

Export the 686 selected records (Table: Students)

ID  
Student\_Number  
prevstudentid

Field Delimiter: Tab

Record Delimiter: CRLF

"Surround fields"  Column titles on 1st row

Optional: Sort Field Name      Direction

> > >

[Back to Direct Database Export Main Screen](#)

Submit

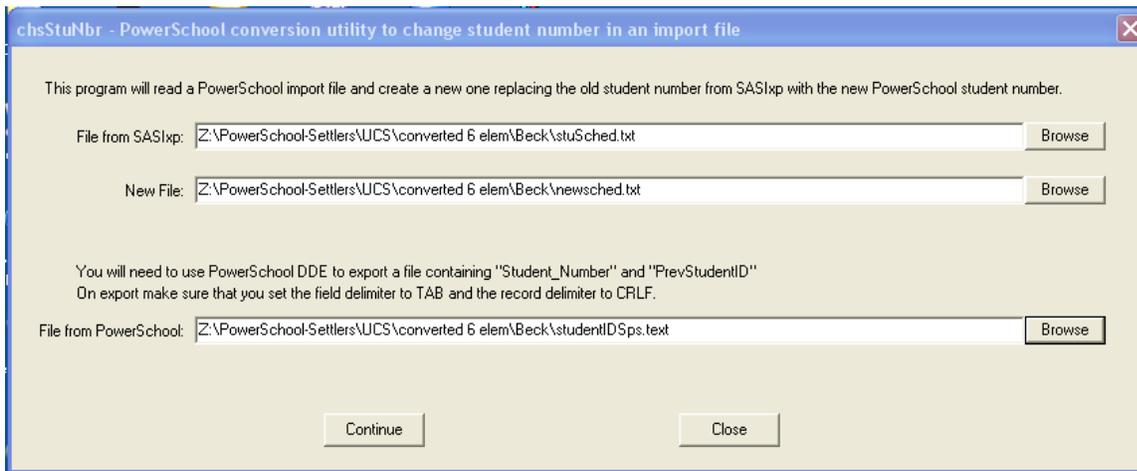
- Save the file to the converted data location and name the file something like newstudentids.txt.

- Run the chsStuNbr program. This program will read a PowerSchool import file and create a new one replacing the old student number from SASIxp with the new PowerSchool student number.

File from SASIxp: stuSched.txt (converted data file)

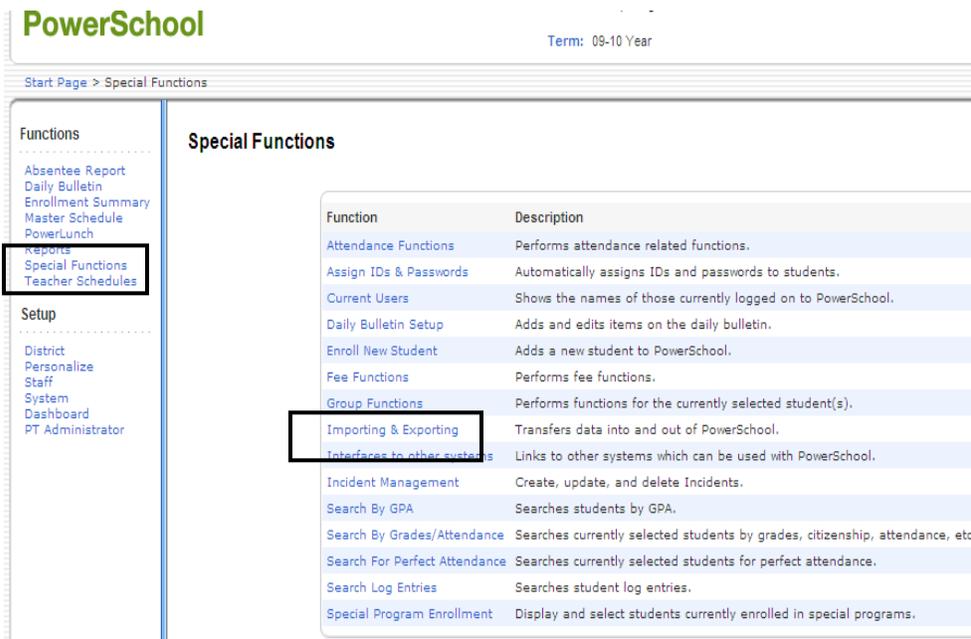
New File: newsched.txt (this will be the file you will import into PowerSchool after running this chsStuNbr program)

File from PowerSchool: newstudentids.text (file created from the Quick Export)



## Importing

- Have your most current SASI data for importing. Although the conversion program allows you to select multiple schools at once it creates a big file with all schools in it. We found we were not able to import from this spreadsheet. Best practice is to select each school's ASTU individually.
  - Using the Update opt Using the Update option when importing into the student or other tables will overwrite the records with the corrected information. It is not necessary to complete clear the table and re-import.
  - Importing Order:
    - ~~School – DON'T DO – THIS PROCESS WAS ONLY DONE BY SCOTT. This is done at District Level. Once the schools imported go into System / System Settings / Reset Server – Submit. Then you go to District Info and change district information.~~
    - Students - done at the school level
    - Teachers - done at the school level
    - Courses - done at the school level
    - Sections - done at the school level
    - Student Schedules - done at the school level
    - Attendance – (mid-year conversions) done at the school level
    - State Reporting – done at the school level
    - Historical Grades - done at the school level
    - The rest of the imports can be done in any order as long as you have followed the above order. Discipline, Emergency Contacts & Medical, Parent/Guardian, Test Scores, Migrant, Staff, Fees, Community Service.
1. Turn Global Sync Off and Set Sync to Non-Atomic Mode
    - System, Special Operations, Turn Global Sync Off – Submit
    - Set Sync to Non-Atomic Mode – Submit
  2. Make sure you are in the correct school and correct year
    - Click on Special Functions
    - Importing & Exporting



3. Click on Quick Import

### Importing & Exporting



4. Once the Quick Import window opens select:

- Table: use the drop down window to choose the correct table to import your data into
- Field delimiter: leave as Tab
- End-of-line marker: choose CRLF
- Character Set: leave as Mac Roman
- File to import: browse to your converted data folder and choose the file you are importing
- Suggest field map: leave checked
- School – you should see the school name that you are importing the data to
- Import

Option	Value
Table:	Students <input type="button" value="v"/>
Field delimiter:	Tab <input type="button" value="v"/> <input type="text"/>
End-of-line marker:	CRLF <input type="button" value="v"/> <input type="text"/>
Character Set:	Mac Roman <input type="button" value="v"/>
File to import:	<input type="text"/> <input type="button" value="Browse..."/>
Suggest field map	<input checked="" type="checkbox"/>
School	Bovenschen MoCI

5. Matching your fields will be the next window to open.

## Students Import (done at the school) (the District Import Checkbox Option Does NOT Work) (see screenshot)

- Select a table: Students
- File to import: student.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is.
- Check to exclude first row - check this
- Do not process this line from the file being imported.
- Allow Update of Enroll Status - only checked the first time you import students
  - The first time you import students check Allow Update of Enroll Status.
  - If you are going to do subsequent imports that have new students, check Allow Update of Enroll Status.
  - If you are importing and updating existing information, check Update the student's record....
- Submit

After importing students, initialize the State Reporting fields:

- Initialize the State Reporting fields after importing students and before importing state reporting data. You only need to do this for one building for the entire district. Do this in the State Reporting tab. Any field that is blank, key in the number 1, Submit, go back in and delete the number 1. If these fields are not initialized they will not show up in the PowerSchool Fields List.

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	Student_Number	----->	Student_Number
3.	First_Name	----->	First_Name
4.	Middle_Name	----->	Middle_Name
5.	Last_Name	----->	Last_Name
6.	Grade_Level	----->	Grade_Level
7.	Gender	----->	Gender
8.	Ethnicity	----->	Ethnicity
9.	DOB	----->	DOB
10.	State_StudentNumber	----->	State_StudentNumber
11.	FTEID	----->	FTEID
12.	EntryDate	----->	EntryDate
13.	ExitDate	----->	ExitDate
14.	Enroll_Status	----->	Enroll_Status
15.	Sched_NextYearGrade	----->	Sched_NextYearGrade
16.	Next_School	----->	Next_School
17.	EntryCode	----->	EntryCode
18.	TransferComment	----->	TransferComment
19.	Districtentrydate	----->	DistrictEntryDate
20.	Schoolentrydate	----->	SchoolEntryDate
21.	Street	----->	Street
22.	City	----->	City
23.	State	----->	State
24.	Zip	----->	Zip
25.	Home_Phone	----->	Home_Phone
26.	Family_Ident	----->	Family_Ident

27.	Mailing_Street	----->	Mailing_Street	▼
28.	Mailing_City	----->	Mailing_City	▼
29.	Mailing_State	----->	Mailing_State	▼
30.	Mailing_Zip	----->	Mailing_Zip	▼
31.	Father	----->	Father	▼
32.	fatherdayphone	----->	fatherdayphone	▼
33.	FatherDayPhone	----->	fatherdayphone	▼
34.	Father_home_phone	----->	Father_home_phone	▼
35.	Mother	----->	Mother	▼
36.	MotherDayPhone	----->	motherdayphone	▼
37.	Mother_home_phone	----->	Mother_home_phone	▼
38.	Guardianship	----->	guardianship	▼
39.	Guardian_Ln	----->	Guardian_LN	▼
40.	Guardian_fn	----->	Guardian_FN	▼
41.	Emerg Contact 1	----->	Emerg_Contact_1	▼
42.	Emerg_1_rel	----->	Emerg_1_Rel	▼
43.	Emerg_1_Ptype	----->	Emerg_1_Ptype	▼
44.	Emerg Phone 1	----->	Emerg_Phone_1	▼
45.	Emerg Contact 2	----->	Emerg_Contact_2	▼
46.	Emerg_2_rel	----->	Emerg_2_Rel	▼
47.	Emerg_2_Ptype	----->	Emerg_2_Ptype	▼
48.	Emerg Phone 2	----->	Emerg_Phone_2	▼
49.	Emerg Contact 3	----->	Emerg_Contact_3	▼
50.	Emerg_3_rel	----->	Emerg_3_Rel	▼
51.	Emerg_3_Ptype	----->	Emerg_3_Ptype	▼
52.	Emerg Phone 3	----->		▼
53.	home_room	----->	Home_Room	▼
54.	graduation_year	----->	graduation_year	▼

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

**Submit**

**PowerSchool Import Options**

Options for importing New Students. Will ignore students that already exist in Powerschool and only create new Enrollment Records for new students.

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

[Submit](#)

Options for updating student files for students already enrolled in Powerschool. Make sure the Allow Update of Enroll Status box is unchecked.

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

[Submit](#)

**Teachers Import (done at the school) (can import same teacher number at another building if import by building)**

- Select a table: Teachers
- File to import: teacher.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is. Make sure SSN to SSN is matched as this will convert the teacher employee number to the PowerSchool SSN field which is necessary for Data Director.
- Check to exclude first row - check this
- Advanced Import Options - leave unchecked
- Submit

Col#	From your file	To Power School
1.	SchoolID	SchoolID
2.	TeacherNumber	TeacherNumber
3.	Last Name	Last_Name
4.	First Name	First_Name
5.	Middle Name	Middle_Name
6.	Title	Title
7.	Gender	gender
8.	DOB	dob
9.	Ethnicity	Ethnicity
10.	Email Addr	Email_Addr
11.	StaffStatus	StaffStatus
12.	Home Phone	Home_Phone
13.	School Phone	School_Phone
14.	SSN	SSN
15.	Street	Street
16.	City	City
17.	State	State
18.	Zip	Zip
19.	Homeroom	Homeroom
20.	LoginID	LoginID
21.	Password	Password
22.	PSAccess	PSAccess
23.	Group	Group
24.	canchangeschool	CanChangeSchool
25.	TeacherLoginid	TeacherLoginID
26.	Teacherloginpw	TeacherLoginPW
27.	GradebookType	Gradebook Type
28.	powergradepw	PowerGradePW
29.	sched_scheduled	Sched_Scheduled

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Advanced Import Options**

Update teacher records if an imported teacher number matches an existing teacher record.

### Courses Import (done at the school or district)

Courses cannot have duplicate course numbers.

You can import at the District level by checking the district import option in the Miscellaneous screen but you will have to manually check courses that belong to each school at each school.

It is recommended that courses be imported at the building level so the import process will populate the courses checkboxes at the building and display the list of courses correctly on the left side of the window.

If you have to import courses again, it will update (see Advanced Import Options and check update). Or, you can delete the file and start fresh.

- Select a table: Courses
- File to import: course.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is.
- Check to exclude first row - check this
- Submit

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	Course Number	----->	Course_Number
3.	Course Name	----->	Course_Name
4.	Credit Hours	----->	Credit_Hours
5.	CreditType	----->	Credit Type
6.	GradeScaleID	----->	GradeScaleID
7.	ExcludeFromClassRank	----->	ExcludeFromClassRank
8.	ExcludeFromGPA	----->	ExcludeFromGPA
9.	ExcludeFromHonorRoll	----->	ExcludeFromHonorRoll
10.	alt_course_number	----->	alt_course_number
11.	MaxClassSize	----->	MaxClassSize

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Advanced Import Options**

Update course records if an imported course number matches an existing course record.

### Sections Import (done at the school)

Sections must be imported at the building level instead of the district level. The advantage of this is that Powerschool also uses teacher number to import sections and if there are multiple teachers with the same number across the district, it will import all sections related to that teacher number, regardless of the school that the section is taught at.

- Select a table: Sections (Master Schedule)
- File to import: sections.txt
- Suggest field map - leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
  - When mapping Course Name - leave blank
  - When mapping Teacher Name - leave blank
  - Have to match Teacher Number to Teacher Number
- Check to exclude first row - check this
- Submit

#### Errors you might see:

- Unknown term ID specified – make sure term ID is filled out at the bottom of the screen.
- Invalid teacher number – due to selecting teacher name in field mapping.

Col#	From your file	To PowerSchool
1.	SchoolID	SchoolID
2.	Course Number	Course_Number
3.	Course Name	
4.	Section Number	Section_Number
5.	TermID	TermID
6.	Teacher Number	Teacher Number
7.	Teacher Name	
8.	Room	Room
9.	Expression	Expression
10.	Attendance_Type_Code	Attendance_Type_Code
11.	Att_Mode_Code	Att_Mode_Code
12.	ExcludeFromClassRank	ExcludeFromClassRank
13.	ExcludeFromGPA	ExcludeFromGPA
14.	ExcludeFromHonorRoll	ExcludeFromHonorRoll
15.	MaxEnrollment	MaxEnrollment
16.	grade_level	Grade_Level

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Advanced Import Options**

Update section records if an imported record matches an existing section. ("Matching" means same course and section numbers in the current school year.)

Synchronize Mode (Warning: This will inactivate all Sections for this school for the current school year that are not found in the file being imported. Do not use this unless you know exactly what you are doing!)

**Submit**

### Student Schedules Import (done at the school)

- Select a table: Student Schedules
- File to import: stuSched.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is.
- Check to exclude first row - check this
- Submit

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	<input type="text"/>
2.	Student_Number	----->	Student_Number
3.	Course_Number	----->	Course_Number
4.	Section_Number	----->	Section_Number
5.	Dateenrolled	----->	DateEnrolled
6.	DateLeft	----->	DateLeft
7.	Term_Number	----->	Term_Number

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Advanced Import Options**

Historical Mode    This will allow records for any term of the active school.

Check Overlap Mode    This will cause records with dates that overlap existing records to be rejected.

Synchronize Mode    This will drop all enrollments for CC records that are not found in the file being imported.

**Attendance Import (done at the school) (This is only done for mid-year conversions. Brian has to convert attendance data)**

- Select a table: Attendance
- File to import: schoolnameatt.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is.
- Check to exclude first row - check this
- Select Meeting attendance mode
- Submit

Col#	From your file	----->	To PowerSchool
1.	Student_Number	----->	Student_Number <input type="button" value="v"/>
2.	Course_Number	----->	Course_Number <input type="button" value="v"/>
3.	Section_Number	----->	Section_Number <input type="button" value="v"/>
4.	Att_Date	----->	Att_Date <input type="button" value="v"/>
5.	Attendance Code	----->	Attendance Code <input type="button" value="v"/>
6.	Period Abbreviation	----->	Period Abbreviation <input type="button" value="v"/>

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="button" value="v"/>
<input type="text"/>	----->	<input type="button" value="v"/>

**Advanced Import Options**

Overwrite existing attendance with the imported attendance. (Applies when attendance is imported for a specific day/class and there is already an attendance code recorded for that day/class.)

Meeting Select an attendance mode

## Health/Emergency Import (done at the school)

- Select a table: Students
- File to import: EmgContAndMed.txt
- Suggest field map: leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
- Check to exclude first row (check this)
- Update the student's record...
- Allow Update of Enroll Status (may not need to check)
- Submit

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	Student_Number	----->	Student_Number
3.	Emerg_Contact_1	----->	Emerg_Contact_1
4.	Emerg_1_Rel	----->	Emerg_1_Rel
5.	Emerg_Phone_1	----->	Emerg_Phone_1
6.	Emerg_1_Ptype	----->	Emerg_1_Ptype
7.	Emerg_Contact_2	----->	Emerg_Contact_2
8.	Emerg_2_Rel	----->	Emerg_2_Rel
9.	Emerg_Phone_2	----->	Emerg_Phone_2
10.	Emerg_2_Ptype	----->	Emerg_2_Ptype
11.	Emerg_Contact_3	----->	Emerg_Contact_3
12.	Emerg_3_Rel	----->	Emerg_3_Rel
13.	Emerg_3_Phone	----->	Emerg_3_Phone
14.	Emerg_3_Ptype	----->	Emerg_3_Ptype
15.	Doctor_Name	----->	Doctor_Name
16.	Doctor_Phone	----->	Doctor_Phone
17.	Dentist_Name	----->	
18.	Dentist_Phone	----->	
19.	Medical_Considerations	----->	Medical_Considerations
20.	allergies	----->	allergies
21.	Immunizator_polio	----->	
22.	Immunizator_mmr	----->	
23.	Immunizator_dpt	----->	
24.	Alert_Medical	----->	Alert_Medical
25.	Alert_MedicalExpires	----->	Alert_MedicalExpires

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

## SRSD Import (done at the school)

Prior to importing, you must initialize all of the fields for all of the tabs to have the ability to map the field. This needs to be done in only one school for the district.

Need to make sure that the cross-reference table from SASIxp is correct. Make sure that the resident LEA and grade or setting fields are correct.

- Select a table: Students
- File to import: StateReporting.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is.
- Check to exclude first row - check this
- Update the student's record for... - check this
- Allow Update of Enroll Status (may not need to check this)
- Submit

Col#	From your file	----->	To Power School
1.	SchoolID	----->	SchoolID
2.	Student_Number	----->	Student_Number
3.	MI_SRSD_StudResLEANum	----->	MI_SRSD_StudResLEANum
4.	MI_SRSD_StudResMembership	----->	MI_SRSD_StudResMembership
5.	MI_SRSD_StudentUIC	----->	MI_SRSD_StudentUIC
6.	MI_SCMOperISDESANum	----->	MI_SCMOperISDESANum
7.	MI_SRSD_AdminUnit	----->	MI_SRSD_AdminUnit
8.	MI_ethnInd	----->	MI_ethnInd
9.	MI_ethnAsi	----->	MI_ethnAsi
10.	MI_ethnAfr	----->	MI_ethnAfr
11.	MI_ethnPac	----->	MI_ethnPac
12.	MI_ethnWhi	----->	MI_ethnWhi
13.	MI_ethnLat	----->	MI_ethnLat
14.	MI_MultipleBirth	----->	MI_MultipleBirth
15.	MI_SRSD_BirthCity	----->	MI_SRSD_BirthCity
16.	meis_attendance	----->	meis_attendance
17.	meis_fte_in_gen_ed	----->	meis_fte_in_gen_ed
18.	MI_Setting	----->	MI_Setting
19.	homeless	----->	homeless
20.	MI_SRSD_PrgmElig_504	----->	MI_SRSD_PrgmElig_504
21.	MI_SRSD_PrgmElig_AltEd	----->	MI_SRSD_PrgmElig_AltEd
22.	MI_SRSD_PrgmElig_DevRetKinder	----->	MI_SRSD_PrgmElig_DevRetKinder
23.	MI_SRSD_PrgmElig_OSStud	----->	MI_SRSD_PrgmElig_OSStud
24.	MI_SRSD_PrgmElig_21CCLC	----->	MI_SRSD_PrgmElig_21CCLC
25.	MI_SRSD_PrgmElig_IntStud	----->	MI_SRSD_PrgmElig_IntStud
26.	MI_SRSD_10_30_DayRule	----->	MI_SRSD_10_30_DayRule
27.	MI_SRSD_Immigrant_Status	----->	MI_SRSD_Immigrant_Status
28.	include_time_share	----->	include_time_share
29.	MI_FiscalEntityTypeCode	----->	MI_FiscalEntityTypeCode
30.	MI_FiscalEntityCode	----->	MI_FiscalEntityCode
31.	MI_StuEnrollEC1	----->	MI_StuEnrollEC1
32.	MI_ECProgramStartDate	----->	MI_ECProgramStartDate

33.	MI_ECProgramEndDate	----->	MI_ECProgramEndDate	▼
34.	MI_StuEnrollECExReas	----->	MI_StuEnrollECExReas	▼
35.	MI_ECDeliveryMethod	----->	MI_ECDeliveryMethod	▼
36.	MI_ECDaysPerWeek	----->	MI_ECDaysPerWeek	▼
37.	MI_ECHoursPerDay	----->	MI_ECHoursPerDay	▼
38.	MI_StuEnrollEC2	----->	MI_StuEnrollEC2	▼
39.	MI_ECProgramStartDate2	----->	MI_ECProgramStartDate2	▼
40.	MI_ECProgramEndDate2	----->	MI_ECProgramEndDate2	▼
41.	MI_StuEnrollECExReas2	----->	MI_StuEnrollECExReas2	▼
42.	MI_ECDeliveryMethod2	----->	MI_ECDeliveryMethod2	▼
43.	MI_ECDaysPerWeek2	----->	MI_ECDaysPerWeek2	▼
44.	MI_ECHoursPerDay2	----->	MI_ECHoursPerDay2	▼
45.	MI_StuEnrollEC3	----->	MI_StuEnrollEC3	▼
46.	MI_ECProgramStartDate3	----->	MI_ECProgramStartDate3	▼
47.	MI_ECProgramEndDate3	----->	MI_ECProgramEndDate3	▼
48.	MI_StuEnrollECExReas3	----->	MI_StuEnrollECExReas3	▼
49.	MI_ECDeliveryMethod3	----->	MI_ECDeliveryMethod3	▼
50.	MI_ECDaysPerWeek3	----->	MI_ECDaysPerWeek3	▼
51.	MI_ECHoursPerDay3	----->	MI_ECHoursPerDay3	▼
52.	MI_StuEnrollEC4	----->	MI_StuEnrollEC4	▼
53.	MI_ECProgramStartDate4	----->	MI_ECProgramStartDate4	▼
54.	MI_ECProgramEndDate4	----->	MI_ECProgramEndDate4	▼
55.	MI_StuEnrollECExReas4	----->	MI_StuEnrollECExReas4	▼
56.	MI_ECDeliveryMethod4	----->	MI_ECDeliveryMethod4	▼
57.	MI_ECDaysPerWeek4	----->	MI_ECDaysPerWeek4	▼
58.	MI_ECHoursPerDay4	----->	MI_ECHoursPerDay4	▼
59.	MI_StuEnrollEC5	----->	MI_StuEnrollEC5	▼
60.	MI_ECProgramStartDate5	----->	MI_ECProgramStartDate5	▼
61.	MI_ECProgramEndDate5	----->	MI_ECProgramEndDate5	▼
62.	MI_StuEnrollECExReas5	----->	MI_StuEnrollECExReas5	▼
63.	MI_ECDeliveryMethod5	----->	MI_ECDeliveryMethod5	▼
64.	MI_ECDaysPerWeek5	----->	MI_ECDaysPerWeek5	▼
65.	MI_ECHoursPerDay5	----->	MI_ECHoursPerDay5	▼

66.	MI_SRSD_PrgmElig_EarlyIntervention	----->	MI_SRSD_PrgmElig_EarlyIntervention	▼
67.	MI_EI_EligibilityCode	----->	MI_EI_EligibilityCode	▼
68.	MI_EI_IFSPDate	----->	MI_EI_IFSPDate	▼
69.	MI_EI_PrimarySetting	----->	MI_EI_PrimarySetting	▼
70.	MI_EI_Service1	----->	MI_EI_Service1	▼
71.	MI_EI_Service2	----->	MI_EI_Service2	▼
72.	MI_EI_Service3	----->	MI_EI_Service3	▼
73.	MI_EI_Service4	----->	MI_EI_Service4	▼
74.	MI_EI_Service5	----->	MI_EI_Service5	▼
75.	MI_EI_Service6	----->	MI_EI_Service6	▼
76.	MI_EI_ExitCode	----->	MI_EI_ExitCode	▼
77.	MI_EI_PartBEligible	----->	MI_EI_PartBEligible	▼
78.	MI_EI_ExitDate	----->	MI_EI_ExitDate	▼
79.	MI_SRSD_PrgmElig_GiftTal	----->	MI_SRSD_PrgmElig_GiftTal	▼
80.	MI_GT_ProgramModel	----->	MI_GT_ProgramModel	▼
81.	MI_GT_SpPrgmOption1	----->	MI_GT_SpPrgmOption1	▼
82.	MI_GT_SpPrgmOption2	----->	MI_GT_SpPrgmOption2	▼
83.	MI_GT_SpPrgmOption3	----->	MI_GT_SpPrgmOption3	▼
84.	MI_SRSD_PrgmElig_LEP	----->	MI_SRSD_PrgmElig_LEP	▼
85.	MI_LEP_Enrollment	----->	MI_LEP_Enrollment	▼
86.	MI_LEP_Title3	----->	MI_LEP_Title3	▼
87.	MI_LEP_Title3_ImmEd	----->	MI_LEP_Title3_ImmEd	▼
88.	MI_LEP_Pupil_LimEng	----->	MI_LEP_Pupil_LimEng	▼
89.	MI_LEP_Refugee_ImpPgm	----->	MI_LEP_Refugee_ImpPgm	▼
90.	MI_LEP_LocFundPgm	----->	MI_LEP_LocFundPgm	▼
91.	MI_LEP_CountryOrigin	----->	MI_LEP_CountryOrigin	▼
92.	MI_LEP_PrimaryLanguage	----->	MI_LEP_PrimaryLanguage	▼
93.	MI_LEP_ReEntryDate	----->	MI_LEP_ReEntryDate	▼
94.	MI_LEP_ExitCode	----->	MI_LEP_ExitCode	▼
95.	MI_LEP_ExitDate	----->	MI_LEP_ExitDate	▼
96.	MI_SRSD_PrgmElig_Title1	----->	MI_SRSD_PrgmElig_Title1	▼
97.	MI_StuInstServTAS1	----->	MI_StuInstServTAS1	▼
98.	MI_StuInstServTAS2	----->	MI_StuInstServTAS2	▼
99.	MI_StuInstServTAS3	----->	MI_StuInstServTAS3	▼

100.	MI_StuInstServTAS4	----->	MI_StuInst Serv TAS4	▼
101.	MI_StuInstServTAS5	----->	MI_StuInst Serv TAS5	▼
102.	MI_StuInstServTAS6	----->	MI_StuInst Serv TAS6	▼
103.	MI_StuInstServTAS7	----->	MI_StuInst Serv TAS7	▼
104.	MI_StuSptServTAS1	----->	MI_StuSpt Serv TAS1	▼
105.	MI_StuSptServTAS2	----->	MI_StuSpt Serv TAS2	▼
106.	MI_StuSptServTAS3	----->	MI_StuSpt Serv TAS3	▼
107.	mi_StuSptServTAS4	----->	MI_StuSpt Serv TAS4	▼
108.	mi_StuSptServTAS5	----->	MI_StuSpt Serv TAS5	▼
109.	mi_StuSptServTAS6	----->	MI_StuSpt Serv TAS6	▼
110.	mi_StuSptServTAS7	----->	MI_StuSpt Serv TAS7	▼
111.	mi_StuSptServTAS8	----->	MI_StuSpt Serv TAS8	▼
112.	mi_SRSD_PrgmElig_SpEd	----->	MI_SRSD_PrgmElig_SpEd	▼
113.	mi_SpEd_PrimaryDisability	----->	MI_SpEd_PrimaryDisability	▼
114.	mi_SpEd_AdditionalDisability	----->	MI_SpEd_AdditionalDisability	▼
115.	mi_SpEd_IEPDate	----->	MI_SpEd_IEPDate	▼
116.	mi_SpEd_ParenConsEval	----->	MI_SpEd_ParenConsEval	▼
117.	mi_SpEd_ResofIIEP	----->	MI_SpEd_ResofIIEP	▼
118.	mi_SpEd_FTE52	----->	MI_SpEd_FTE52	▼
119.	mi_SpEd_FTE53	----->	MI_SpEd_FTE53	▼
120.	mi_SpEd_PrgmServiceCode1	----->	MI_SpEd_PrgmServiceCode 1	▼
121.	mi_SpEd_PrgmServiceCode2	----->	MI_SpEd_PrgmServiceCode2	▼
122.	mi_SpEd_PrgmServiceCode3	----->	MI_SpEd_PrgmServiceCode3	▼
123.	mi_SpEd_SupportServices1	----->	MI_SpEd_SupportServices 1	▼
124.	mi_SpEd_SupportServices2	----->	MI_SpEd_SupportServices2	▼
125.	mi_SpEd_SupportServices3	----->	MI_SpEd_SupportServices3	▼
126.	mi_SpEd_SupportServices4	----->	MI_SpEd_SupportServices4	▼
127.	mi_SpEd_SupportServices5	----->	MI_SpEd_SupportServices5	▼
128.	mi_SpEd_PrimaryEdSetting	----->	MI_SpEd_PrimaryEdSetting	▼
129.	mi_SpEd_ExitCode	----->	MI_SpEd_ExitCode	▼
130.	mi_SpEd_ExitDate	----->	MI_SpEd_ExitDate	▼
131.	mi_SpEd_IEPDays	----->	MI_SpEd_IEPDays	▼
132.	mi_SpEd_IEPAnotherDistrict	----->	MI_SpEd_IEPAnotherDistrict	▼
133.	mi_StdntExplus_IncidentDate	----->	MI_StdntExplus_IncidentDate	▼

101.	MI_StuInstServTAS5	----->	MI_StuInstServTAS5	▼
102.	MI_StuInstServTAS6	----->	MI_StuInstServTAS6	▼
103.	MI_StuInstServTAS7	----->	MI_StuInstServTAS7	▼
104.	MI_StuSptServTAS1	----->	MI_StuSptServTAS1	▼
105.	MI_StuSptServTAS2	----->	MI_StuSptServTAS2	▼
106.	MI_StuSptServTAS3	----->	MI_StuSptServTAS3	▼
107.	m_StuSptServTAS4	----->	MI_StuSptServTAS4	▼
108.	m_StuSptServTAS5	----->	MI_StuSptServTAS5	▼
109.	m_StuSptServTAS6	----->	MI_StuSptServTAS6	▼
110.	m_StuSptServTAS7	----->	MI_StuSptServTAS7	▼
111.	m_StuSptServTAS8	----->	MI_StuSptServTAS8	▼
112.	m_SRSD_PrgmElig_SpEd	----->	MI_SRSD_PrgmElig_SpEd	▼
113.	m_SpEd_PrimaryDisability	----->	MI_SpEd_PrimaryDisability	▼
114.	m_SpEd_AdditionalDisability	----->	MI_SpEd_AdditionalDisability	▼
115.	m_SpEd_IEPDate	----->	MI_SpEd_IEPDate	▼
116.	m_SpEd_ParenConsEval	----->	MI_SpEd_ParenConsEval	▼
117.	m_SpEd_ResofIIEP	----->	MI_SpEd_ResofIIEP	▼
118.	m_SpEd_FTE52	----->	MI_SpEd_FTE52	▼
119.	m_SpEd_FTE53	----->	MI_SpEd_FTE53	▼
120.	m_SpEd_PrgmServiceCode1	----->	MI_SpEd_PrgmServiceCode1	▼
121.	m_SpEd_PrgmServiceCode2	----->	MI_SpEd_PrgmServiceCode2	▼
122.	m_SpEd_PrgmServiceCode3	----->	MI_SpEd_PrgmServiceCode3	▼
123.	m_SpEd_SupportServices1	----->	MI_SpEd_SupportServices1	▼
124.	m_SpEd_SupportServices2	----->	MI_SpEd_SupportServices2	▼
125.	m_SpEd_SupportServices3	----->	MI_SpEd_SupportServices3	▼
126.	m_SpEd_SupportServices4	----->	MI_SpEd_SupportServices4	▼
127.	m_SpEd_SupportServices5	----->	MI_SpEd_SupportServices5	▼
128.	m_SpEd_PrimaryEdSetting	----->	MI_SpEd_PrimaryEdSetting	▼
129.	m_SpEd_ExitCode	----->	MI_SpEd_ExitCode	▼
130.	m_SpEd_ExitDate	----->	MI_SpEd_ExitDate	▼
131.	m_SpEd_IEPDays	----->	MI_SpEd_IEPDays	▼
132.	m_SpEd_IEPAnotherDistrict	----->	MI_SpEd_IEPAnotherDistrict	▼
133.	m_StdntExplus_IncidentDate	----->	MI_StdntExplus_IncidentDate	▼
134.	m_StdntExplus_ExpelDate	----->	MI_StdntExplus_ExpelDate	▼

135.	mi_StdntExplus_ExpelLength	----->	MI_StdntExplus_ExpelLength	▼
136.	mi_StdntExplus_IncidentType	----->	MI_StdntExplus_IncidentType	▼
137.	mi_StdntExplus_IncidentLocation	----->	MI_StdntExplus_IncidentLocation	▼
138.	mi_StdntExplus_IncidentTime	----->	MI_StdntExplus_IncidentTime	▼
139.	mi_StdntExplus_DamageCost	----->	MI_StdntExplus_DamageCost	▼
140.	mi_StdntExplus_IncidentPrimaryVictim	----->	MI_StdntExplus_IncidentPrimaryVictim	▼
141.	mi_StdntExplus_ExpelFollowup	----->	MI_StdntExplus_ExpelFollowup	▼
142.	mi_SRSD_PrgmElig_AdEd	----->	MI_SRSD_PrgmElig_AdEd	▼
143.	mi_ADED_ParticipantFunding	----->	MI_ADED_ParticipantFunding	▼
144.	mi_ADED_July_Program1	----->	MI_ADED_July_Program1	▼
145.	mi_ADED_July_FTE1	----->	MI_ADED_July_FTE1	▼
146.	mi_ADED_July_Program2	----->	MI_ADED_July_Program2	▼
147.	mi_ADED_July_FTE2	----->	MI_ADED_July_FTE2	▼
148.	mi_ADED_July_Program3	----->	MI_ADED_July_Program3	▼
149.	mi_ADED_July_FTE3	----->	MI_ADED_July_FTE3	▼
150.	mi_ADED_September_Program1	----->	MI_ADED_September_Program1	▼
151.	mi_ADED_September_FTE1	----->	MI_ADED_September_FTE1	▼
152.	mi_ADED_September_Program2	----->	MI_ADED_September_Program2	▼
153.	mi_ADED_September_FTE2	----->	MI_ADED_September_FTE2	▼
154.	mi_ADED_September_Program3	----->	MI_ADED_September_Program3	▼
155.	mi_ADED_September_FTE3	----->	MI_ADED_September_FTE3	▼
156.	mi_ADED_February_Program1	----->	MI_ADED_February_Program1	▼
157.	mi_ADED_February_FTE1	----->	MI_ADED_February_FTE1	▼
158.	mi_ADED_February_Program2	----->	MI_ADED_February_Program2	▼
159.	mi_ADED_February_FTE2	----->	MI_ADED_February_FTE2	▼
160.	mi_ADED_February_Program3	----->	MI_ADED_February_Program3	▼
161.	mi_ADED_February_FTE3	----->	MI_ADED_February_FTE3	▼
162.	mi_ADED_April_Program1	----->	MI_ADED_April_Program1	▼
163.	mi_ADED_April_FTE1	----->	MI_ADED_April_FTE1	▼
164.	mi_ADED_April_Program2	----->	MI_ADED_April_Program2	▼
165.	mi_ADED_April_FTE2	----->	MI_ADED_April_FTE2	▼
166.	mi_ADED_April_Program3	----->	MI_ADED_April_Program3	▼
167.	mi_ADED_April_FTE3	----->	MI_ADED_April_FTE3	▼

168.	m_ADED_DiplomaGEDStatus	----->	MI_ADED_DiplomaGEDStatus	▼
169.	m_NinthGradeAttended	----->	MI_NinthGradeAttended	▼
170.	m_NinthGradeEnrolled	----->	MI_NinthGradeEnrolled	▼
171.	m_TenthGradeAttended	----->	MI_TenthGradeAttended	▼
172.	m_TenthGradeEnrolled	----->	MI_TenthGradeEnrolled	▼
173.	m_EleventhGradeAttended	----->	MI_EleventhGradeAttended	▼
174.	m_EleventhGradeEnrolled	----->	MI_EleventhGradeEnrolled	▼
175.	m_TwelfthGradeAttended	----->	MI_TwelfthGradeAttended	▼
176.	m_TwelfthGradeEnrolled	----->	MI_TwelfthGradeEnrolled	▼

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

### State Reporting – UMSD Import (done at the school)

- Select a table: Students
- File to import: UMSDS.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is
- Check to exclude first row - check this
- Update the student’s record for... - check this
- Allow Update of Enroll Status–(may not need to check this)
- Submit

#### Import Records from an ASCII Text File

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	Student_Number	----->	Student_Number
3.	MI_SRSD_Resident_County_Code	----->	
4.	MI_EntryYear	----->	MI_EntryYear
5.	State	----->	State
6.	MI_SRSD_10_30_Day_Rule	----->	
7.	MI_PCCM	----->	
8.	MI_PCT	----->	
9.	homeless	----->	homeless
10.	MI_Unaccompanied_Youth	----->	
11.	MI_LEP_Enrollment	----->	MI_LEP_Enrollment
12.	MI_LEP_PrimaryLanguage	----->	MI_LEP_PrimaryLanguage
13.	MI_GT_ProgramModel	----->	MI_GT_ProgramModel
14.	MI_GT_SpPrgmOption1	----->	MI_GT_SpPrgmOption1
15.	MI_GT_SpPrgmOption2	----->	MI_GT_SpPrgmOption2
16.	MI_GT_SpPrgmOption3	----->	MI_GT_SpPrgmOption3
17.	Multiple Fields	----->	
18.	MI_DateofConsent	----->	MI_DateOfConsent
19.	MI_BirthCountry	----->	MI_BirthCountry
20.	MI_AsOfDate	----->	MI_AsOfDate

## Parent/Guardian Import (done at the school)

- Select a table: Students
- File to import: Parent.txt
- Suggest field map - leave checked
- Import
- Map Fields – leave as is
- Check to exclude first row - check this
- Update the student's record for... - check this
- Allow Update of Enroll Status–(may not need to check this)
- Submit

Co#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID <input type="button" value="v"/>
2.	Student_Number	----->	Student_Number <input type="button" value="v"/>
3.	contact1_last	----->	contact1_last <input type="button" value="v"/>
4.	contact1_first	----->	contact1_first <input type="button" value="v"/>
5.	contact1_rel	----->	contact1_rel <input type="button" value="v"/>
6.	contact1_ecpriority	----->	contact1_ecpriority <input type="button" value="v"/>
7.	contact1_street	----->	contact1_street <input type="button" value="v"/>
8.	contact1_city	----->	contact1_city <input type="button" value="v"/>
9.	contact1_state	----->	contact1_state <input type="button" value="v"/>
10.	contact1_zip	----->	contact1_zip <input type="button" value="v"/>
11.	contact1_homephone	----->	contact1_homephone <input type="button" value="v"/>
12.	contact1_cellphone	----->	contact1_cellphone <input type="button" value="v"/>
13.	contact1_employer	----->	contact1_employer <input type="button" value="v"/>
14.	contact1_workphone	----->	contact1_workphone <input type="button" value="v"/>
15.	contact1_workextension	----->	contact1_workextension <input type="button" value="v"/>
16.	contact1_liveswith	----->	contact1_liveswith <input type="button" value="v"/>
17.	contact1_recvmail	----->	contact1_recvmail <input type="button" value="v"/>
18.	contact1_email	----->	contact1_email <input type="button" value="v"/>
19.	contact2_last	----->	contact2_last <input type="button" value="v"/>
20.	contact2_first	----->	contact2_first <input type="button" value="v"/>
21.	contact2_rel	----->	contact2_rel <input type="button" value="v"/>
22.	contact2_ecpriority	----->	contact2_ecpriority <input type="button" value="v"/>
23.	contact2_street	----->	contact2_street <input type="button" value="v"/>
24.	contact2_city	----->	contact2_city <input type="button" value="v"/>
25.	contact2_state	----->	contact2_state <input type="button" value="v"/>
26.	contact2_zip	----->	contact2_zip <input type="button" value="v"/>
27.	contact2_homephone	----->	contact2_homephone <input type="button" value="v"/>
28.	contact2_cellphone	----->	contact2_cellphone <input type="button" value="v"/>
29.	contact2_employer	----->	contact2_employer <input type="button" value="v"/>
30.	contact2_workphone	----->	contact2_workphone <input type="button" value="v"/>
31.	contact2_workextension	----->	contact2_workextension <input type="button" value="v"/>
32.	contact2_liveswith	----->	contact2_liveswith <input type="button" value="v"/>

33.	contact2_recvmail	----->	contact2_recvmail	▼
34.	contact2_email	----->	contact2_email	▼
35.	contact3_last	----->	contact3_last	▼
36.	contact3_first	----->	contact3_first	▼
37.	contact3_rel	----->	contact3_rel	▼
38.	contact3_ecpriority	----->	contact3_ecpriority	▼
39.	contact3_street	----->	contact3_street	▼
40.	contact3_city	----->	contact3_city	▼
41.	contact3_state	----->	contact3_state	▼
42.	contact3_zip	----->	contact3_zip	▼
43.	contact3_homephone	----->	contact3_homephone	▼
44.	contact3_cellphone	----->	contact3_cellphone	▼
45.	contact3_employer	----->		▼
46.	contact3_workphone	----->	contact3_workphone	▼
47.	contact3_workextension	----->	contact3_workextension	▼
48.	contact3_liveswith	----->	contact3_liveswith	▼
49.	contact3_recvmail	----->	contact3_recvmail	▼
50.	contact3_email	----->		▼
51.	contact4_last	----->	contact4_last	▼
52.	contact4_first	----->	contact4_first	▼
53.	contact4_rel	----->	contact4_rel	▼
54.	contact4_ecpriority	----->	contact4_ecpriority	▼
55.	contact4_street	----->	contact4_street	▼
56.	contact4_city	----->	contact4_city	▼
57.	contact4_state	----->	contact4_state	▼
58.	contact4_zip	----->	contact4_zip	▼
59.	contact4_homephone	----->	contact4_homephone	▼
60.	contact4_cellphone	----->	contact4_cellphone	▼
61.	contact4_employer	----->	contact4_employer	▼
62.	contact4_workphone	----->	contact4_workphone	▼
63.	contact4_workextension	----->		▼
64.	contact4_liveswith	----->	contact4_liveswith	▼
65.	contact4_recvmail	----->	contact4_recvmail	▼
66.	contact4_email	----->	contact4_email	▼
67.	contact5_last	----->	contact5_last	▼
68.	contact5_first	----->	contact5_first	▼

69.	contact5_rel	---->	contact5_rel	▼
70.	contact5_ecpriority	---->	contact5_ecpriority	▼
71.	contact5_street	---->	contact5_street	▼
72.	contact5_city	---->	contact5_city	▼
73.	contact5_state	---->	contact5_state	▼
74.	contact5_zip	---->	contact5_zip	▼
75.	contact5_homephone	---->	contact5_homephone	▼
76.	contact5_cellphone	---->	contact5_cellphone	▼
77.	contact5_employer	---->	contact5_employer	▼
78.	contact5_workphone	---->	contact5_workphone	▼
79.	contact5_workextension	---->		▼
80.	contact5_liveswith	---->	contact5_liveswith	▼
81.	contact5_recvmail	---->	contact5_recvmail	▼
82.	contact5_email	---->		▼
83.	contact6_last	---->	contact6_last	▼
84.	contact6_first	---->	contact6_first	▼
85.	contact6_rel	---->	contact6_rel	▼
86.	contact6_ecpriority	---->	contact6_ecpriority	▼
87.	contact6_street	---->	contact6_street	▼
88.	contact6_city	---->	contact6_city	▼
89.	contact6_state	---->	contact6_state	▼
90.	contact6_zip	---->	contact6_zip	▼
91.	contact6_homephone	---->	contact6_homephone	▼
92.	contact6_cellphone	---->	contact6_cellphone	▼
93.	contact6_employer	---->		▼
94.	contact6_workphone	---->		▼
95.	contact6_workextension	---->		▼
96.	contact6_liveswith	---->	contact6_liveswith	▼
97.	contact6_recvmail	---->		▼
98.	contact6_email	---->		▼

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

	---->		▼
	---->		▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range: 10000016 - 214748364
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

### Migrant (done at the school)

- Select a table: Students
- File to import: migrant.txt
- Suggest field map: leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
- Check to exclude first row (check this)
- Update the student’s record for... - check this
- Allow Update of Enroll Status (may not need to check this)
- Submit

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	Student_Number	----->	Student_Number
3.	MI_SRSD_PrgmElig_MigrantEd	----->	MI_SRSD_PrgmElig_MigrantEd
4.	MI_MigrantEd_EntryDate	----->	MI_MigrantEd_EntryDate
5.	MI_MigrantEd_Instruct_LangArts	----->	MI_MigrantEd_Instruct_LangArts
6.	MI_MigrantEd_Instruct_LEPESL	----->	MI_MigrantEd_Instruct_LEPESL
7.	MI_MigrantEd_Instruct_Math	----->	MI_MigrantEd_Instruct_Math
8.	MI_MigrantEd_Instruct_Science	----->	MI_MigrantEd_Instruct_Science
9.	MI_MigrantEd_Instruct_Social	----->	MI_MigrantEd_Instruct_Social
10.	MI_MigrantEd_Instruct_Voc	----->	MI_MigrantEd_Instruct_Voc
11.	MI_MigrantEd_Instruct_Other	----->	MI_MigrantEd_Instruct_Other
12.	MI_MigrantEd_Support_GuidCounsel	----->	MI_MigrantEd_Support_GuidCounsel
13.	MI_MigrantEd_Support_ScoWrk	----->	MI_MigrantEd_Support_ScoWrk
14.	MI_MigrantEd_Support_DOPrevention	----->	MI_MigrantEd_Support_DOPrevention
15.	MI_MigrantEd_Support_Health	----->	MI_MigrantEd_Support_Health
16.	MI_MigrantEd_Support_Dental	----->	MI_MigrantEd_Support_Dental
17.	MI_MigrantEd_Support_EyeCare	----->	MI_MigrantEd_Support_EyeCare
18.	MI_MigrantEd_Support_PupilTransport	----->	MI_MigrantEd_Support_PupilTransport
19.	MI_MigrantEd_Support_Other	----->	MI_MigrantEd_Support_Other
20.	MI_MigrantEd_SumInstruct_LangArts	----->	MI_MigrantEd_SumInstruct_LangArts
21.	MI_MigrantEd_SumInstruct_LEPESL	----->	MI_MigrantEd_SumInstruct_LEPESL
22.	MI_MigrantEd_SumInstruct_Math	----->	MI_MigrantEd_SumInstruct_Math
23.	MI_MigrantEd_SumInstruct_Science	----->	MI_MigrantEd_SumInstruct_Science
24.	MI_MigrantEd_SumInstruct_Social	----->	MI_MigrantEd_SumInstruct_Social
25.	MI_MigrantEd_SumInstruct_Voc	----->	MI_MigrantEd_SumInstruct_Voc

26.	MI_MigrantEd_SumInstruct_Other	----->	MI_MigrantEd_SumInstruct_Other	▼
27.	MI_MigrantEd_SumSupport_GuidCounsel	----->	MI_MigrantEd_SumSupport_GuidCounsel	▼
28.	MI_MigrantEd_SumSupport_SocWrk	----->	MI_MigrantEd_SumSupport_SocWrk	▼
29.	MI_MigrantEd_SumSupport_DOPrevent	----->	MI_MigrantEd_SumSupport_DOPrevent	▼
30.	MI_MigrantEd_SumSupport_Health	----->	MI_MigrantEd_SumSupport_Health	▼
31.	MI_MigrantEd_SumSupport_Dental	----->	MI_MigrantEd_SumSupport_Dental	▼
32.	MI_MigrantEd_SumSupport_Nutrition	----->	MI_MigrantEd_SumSupport_Nutrition	▼
33.	MI_MigrantEd_SumSupport_PupilTransport	----->	MI_MigrantEd_SumSupport_PupilTransport	▼
34.	MI_MigrantEd_SumSupport_Other	----->	MI_MigrantEd_SumSupport_Other	▼
35.	MI_MigrantEd_PrgmFunding	----->	MI_MigrantEd_PrgmFunding	▼

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Advanced Import Options

Note: The student number is required in all import files. All information is keyed off of the student number field.

If the file being imported contains a student number that matches a student number already in the PowerSchool system, how would you like it handled?

- Do not process that line from the file being imported.
- Update the student's record with the information from the file being imported. (Note: even if you have this option selected, if the student is in a different school, the information will not be updated).
- Generate a new, unique student number for the student (from the range specified below) and import the data from the import file.
  - New student numbers are generated at random from within this range:  -
- Synchronize Mode (Warning: This will transfer out all Students who are not found in the file being imported. Do not use this unless you know exactly what you are doing!)
- Allow Update of Enroll Status (Warning: Don't use this unless you know exactly what you are doing!)

Submit

### Discipline (done at the school)

- Select a table: Log Entries
- File to import: btsDiscipline.txt or xpDiscipline
- Suggest field map: leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
- Check to exclude first row (check this)
- Allow Update of Enroll Status (may not need to check this)
- Submit

Col#	From your file	----->	To PowerSchool
1.	SchoolID	----->	SchoolID
2.	First_Name	----->	
3.	Last_Name	----->	
4.	Student_Number	----->	Student_Number
5.	Entry	----->	Entry
6.	Entry_Author	----->	Entry_Author
7.	Entry_Date	----->	Entry_Date
8.	Entry_Time	----->	Entry_Time
9.	LogtypeId	----->	LogTypeID
10.	Subtype	----->	Subtype
11.	Subject	----->	Subject
12.	Consequence	----->	Consequence
13.	Discipline_IncidentType	----->	Discipline_IncidentType
14.	Discipline_IncidentTypeCategory	----->	Discipline_IncidentTypeCategory
15.	Category	----->	Category
16.	Discipline_IncidentTypeDetail	----->	Discipline_IncidentTypeDetail
17.	Discipline_IncidentDate	----->	Discipline_IncidentDate
18.	Discipline_IncidentContext	----->	Discipline_IncidentContext
19.	Discipline_IncidentLocation	----->	Discipline_IncidentLocation
20.	Discipline_IncidentLocDetail	----->	Discipline_IncidentLocDetail
21.	Discipline_Offender	----->	Discipline_Offender
22.	Discipline_Reporter	----->	Discipline_Reporter
23.	Discipline_ReporterID	----->	Discipline_ReporterID
24.	Discipline_VictimType	----->	Discipline_VictimType
25.	Discipline_FelonyFlag	----->	Discipline_FelonyFlag
26.	Discipline_LikelyInjuryFlag	----->	Discipline_LikelyInjuryFlag
27.	Discipline_SchoolRulesVioFlag	----->	Discipline_SchoolRulesVioFlag
28.	Discipline_PoliceInvolvedFlag	----->	Discipline_PoliceInvolvedFlag

29.	Discipline_HearingOfficerFlag	----->	Discipline_HearingOfficerFlag	▼
30.	Discipline_GangRelatedFlag	----->	Discipline_GangRelatedFlag	▼
31.	Discipline_HateCrimeRelatedFlag	----->	Discipline_HateCrimeRelatedFlag	▼
32.	Discipline_AlcoholRelatedFlag	----->	Discipline_AlcoholRelatedFlag	▼
33.	Discipline_DrugRelatedFlag	----->	Discipline_DrugRelatedFlag	▼
34.	Discipline_DrugTypeDetail	----->	Discipline_DrugTypeDetail	▼
35.	Discipline_WeaponRelatedFlag	----->	Discipline_WeaponRelatedFlag	▼
36.	Discipline_WeaponType	----->	Discipline_WeaponType	▼
37.	Discipline_WeaponTypeNotes	----->	Discipline_WeaponTypeNotes	▼
38.	Discipline_MoneyLossValue	----->	Discipline_MoneyLossValue	▼
39.	Discipline_ActionDate	----->	Discipline_ActionDate	▼
40.	Discipline_ActionTaken	----->	Discipline_ActionTaken	▼
41.	Discipline_ActionTakenDetail	----->	Discipline_ActionTakenDetail	▼
42.	Discipline_ActionTakenEndDate	----->	Discipline_ActionTakenEndDate	▼
43.	Discipline_DurationAssigned	----->	Discipline_DurationAssigned	▼
44.	Discipline_DurationActual	----->	Discipline_DurationActual	▼
45.	Discipline_DurationChangeSource	----->	Discipline_DurationChangeSource	▼
46.	Discipline_DurationNotes	----->	Discipline_DurationNotes	▼
47.	Discipline_Sequence	----->	Discipline_Sequence	▼
48.	Custom	----->	Custom	▼

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>	▼
<input type="text"/>	----->	<input type="text"/>	▼

Submit

**Staff (done at the school) (Note: check spreadsheet info carefully before doing this)**

- Select a table: Teachers
- File to import: staff.txt
- Suggest field map: leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
- Check to exclude first row (check this)
- Allow Update of Enroll Status
- Submit

Col#	From your file	To Power School
1.	SchoolID	SchoolID
2.	TeacherNumber	TeacherNumber
3.	Last Name	Last_Name
4.	First Name	First_Name
5.	Middle_name	Middle_Name
6.	Title	Title
7.	Gender	gender
8.	DOB	dob
9.	Ethnicity	Ethnicity
10.	Email_Addr	Email_Addr
11.	StaffStatus	StaffStatus
12.	Home Phone	Home_Phone
13.	School Phone	School_Phone
14.	SSN	SSN
15.	Street	Street
16.	City	City
17.	State	State
18.	Zip	Zip
19.	Homeroom	Homeroom
20.	LoginID	LoginID
21.	Password	Password
22.	PSAccess	PSAccess
23.	Group	Group
24.	canchangeschool	CanChangeSchool
25.	TeacherLoginid	TeacherLoginID
26.	Teacherloginpw	TeacherLoginPW
27.	GradebookType	Gradebook Type
28.	powergradepw	PowerGradePW
29.	sched_scheduled	Sched_Scheduled

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Advanced Import Options**

Update teacher records if an imported teacher number matches an existing teacher record.

### Historical Grades (done at the school)

- Select a table: Historical Grades
- File to import: history.txt
- Suggest field map: leave checked
- Import
- Map Fields – need to map each field where you can find a matching field.
- Check to exclude first row (check this)
- Allow Update of Enroll Status
- Submit

Col#	From your file	To PowerSchool
1.	SchoolID	SchoolID
2.	Student Number	
3.	Course Name	Course_Name
4.	Course Number	Course_Number
5.	EarnedCrHrs	EarnedCrHrs
6.	Grade	Grade
7.	PotentialCrHrs	PotentialCrHrs
8.	Storecode	StoreCode
9.	TermId	TermID
10.	GPA points	GPA_Points
11.	Percent	Percent
12.	SchoolName	SchoolName
13.	Grade_Level	Grade_Level
14.	Credit Type	Credit_Type
15.	Teacher Name	Teacher_Name
16.	Schoolid	SchoolID
17.	ExcludeFromGPA	ExcludeFromGPA
18.	ExcludeFromClassRank	ExcludeFromClassRank
19.	ExcludeFromHonorRoll	ExcludeFromHonorRoll

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

\_\_\_\_\_

\_\_\_\_\_

**Advanced Import Options**

Add imported rows to database.

Allow multiple grades for a student to be stored when identical term and storecode are detected.

Update existing database records using imported rows.

Different import files may specify school years differently. For example, in one file, 1997 may refer to the 96-97 school year while in another 1997 refers to the 97-98 school year. In this file being imported now, when a grade is imported and the year (as an example) specified is 1997, it should be interpreted as meaning the 96-97 school year.

If a course name is imported that is not recognized, create a new, unique course number for it. Draw this course number from this range: \_\_\_\_\_ - \_\_\_\_\_

Allow grades to be stored which contain both (a) no letter grade and (b) no earned credit. (Such grades will not be stored unless you select this option.)

**Submit**

History.txt import:

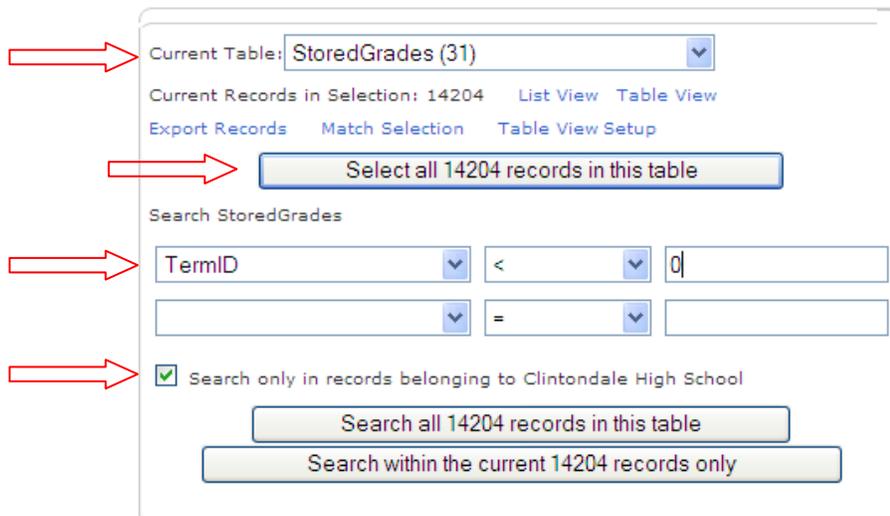
Change Year/Term from “None” to 09/10S1 or 09/10S2 after importing high school course history.

After you import history.txt into your high school in the correct year, you may find that the year/term says “NONE” – that is a “known issue”. The year imports as a negative number behind the scenes, so “NONE” is entered instead.

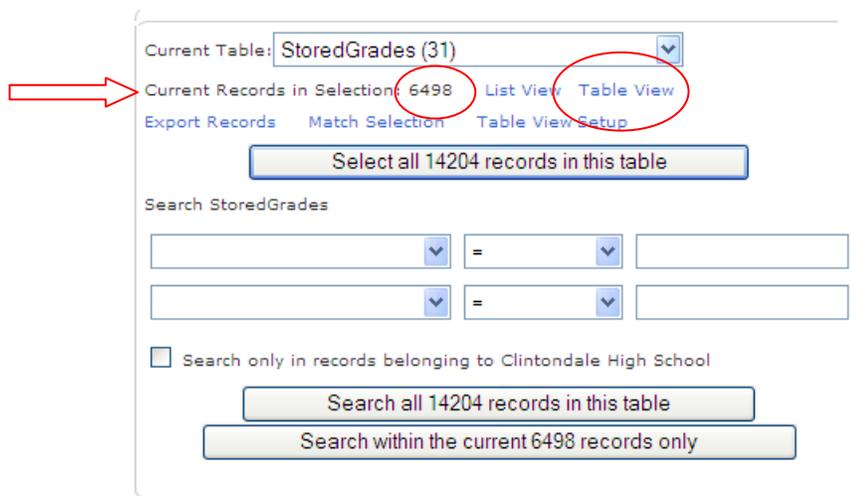
Year/Term	Grd Lvl	Course	Earned Credit	S1	S2
None	11	11th Engl LA CP I	0.5	C	.
None	11	1stYrTheatreArts I	0.5	A-	.
None	11	Beg Mous I	0.5	B	.
None	11	CHEMISTRY CP I	0.5	C	.
None	11	Civics	0.5	C-	.
None	11	PRE CALCULUS I	0.5	D+	.
None	11	11th Eng LA CP II	0.5	.	B-
None	11	1stY TheatreArt II	0.5	.	A
None	11	Beg Mous II	0.5	.	B
None	11	CHEMISTRY CP II	0.5	.	C+
None	11	ECONOMICS	0.5	.	B+
None	11	PRE CALCULUS II	0.5	.	C-
08-09 S1	10	10th Eng LA CP I	0.5	C+	.
08-09 S2	10	10th Eng LA CP II	0.5	.	D
08-09 S1	10	2nd YR SPANISH I	0.5	B	.
08-09 S2	10	2nd YR SPANISH II	0.5	.	B
08-09 S1	10	ADV ALGEBRA I	0.5	C-	.

To correct these negative numbers behind the scene you will go to System, DDE and select the Stored Grades table

Select all the records, then Search StoredGrades for TermID < 0 and check the box to search records belonging to your school. Search all records.



Once all the records are searched and the “Current Records in Selection: xxxx displays (see below), you can look at them by selecting Table View



By looking at your list, you can determine which negative number corresponds to which semester. In this data -9000 corresponds to S1 and -8900 corresponds to S2.

List Records: StoredGrades

A	SchoolID	TermID	SectionID	StudentID	Course Number	Course Name	Grade
1	0	-9000	0	5	HS000016	LL 11th Eng LA I	C-
2	0	-9000	0	5	HS000077	LL Chemistry I	C-
3	0	-9000	0	5		Beg Mous I	F
4	0	-9000	0	5	HS000081	LL U.S. HIST I	A
5	0	-9000	0	5		Ftns/Hlth I (BB)	A
6	0	-9000	0	5	HS000052	LL Geometry I	C+
7	0	-8900	0	5	HS000017	LL 11th Eng LA II	B-
8	0	-8900	0	5	HS000078	LL Chemistry II	D
9	0	-8900	0	5		Ftns/Hlth II (BB)	A
10	0	-8900	0	5	HS000102	Intro to Art II	F
11	0	-8900	0	5	HS000086	LL Economics	A+

Now we can search for the negative numbers and change them. Search for TermID = -9000 which is our S1 term, check the box to search only in this school then select Search within the current xxx records only.

Current Table: StoredGrades (31)  
 Current Records in Selection: 6498  
 Search StoredGrades  
 TermID = -9000  
 Search only in records belonging to Clintondale High School  
 Search within the current 6498 records only

I now have 3233 records that list as -9000 and must be corrected in order for the Year/Term to display in Historical Grades. In this example the number we must enter to correct the problem is 1901 – which is the name for the term as you have set-up in Years and Terms.

Current Table: StoredGrades (31)  
 Current Records in Selection: 3233  
 Select all 14204 records in this table

You can check your List View again to confirm that these are the records that you wish to change.

Now go to DDA and make the changes. Select Modify Records. Make sure the correct number of records are displayed.

### Modify Records

Current Table: StoredGrades  
Current number of records in selection: 3233  
TermID := 1901  
Note: This function will permanently modify your database. Use only if you know exactly what you are doing, and then only with extreme caution. Changes are permanent. You may use ^ to reference the existing value of the field.  
 Check here to verify a delete command  
[Back to Direct Database Access Main Screen](#)  
Delete Selected Records    Modify Selected Records

Select TermID and the correct value for this field. Now select Modify Selected Records.

You will follow this procedure again to change the other incorrect internal number. In this example it was -8900 for 09/10 S2 and the internal number that you will use in the Modify Records would be your S2 number 1902.

### Test Score (directions are printed and can be found under the Test Scores section in this binder and also in the Conversion & Importing Processes folder)

- Importing Test Scores.doc
- Importing Test Scores with delete.doc

### Fees & Community Service (directions are printed and can be found under Fees & Comm Serv section in this binder and also in the Conversion & Importing Processes folder)

- Fees & Comm Serv Conversion and Import Directions cvtOther.exe program
- Brian assisted the Pioneers with this process

cvtOther.exe - SASlxp to PowerSchool Conversion Ver 1.3.0.0

Select area to export

- Fees
- Community Service

Select All

Clear All

Select the school to convert by selecting the appropriate ASTU files:

Browse

Write export files to the following folder:

Browse

Student Number/Student ID reference for this school (exported from PowerSchool)

Browse ?

PowerSchool School ID: \_\_\_\_\_

Continue Close

**District of Residency Import** ((template name: District Numbers by County Macomb)

- Special Functions
- Import / Quick Import
- Choose **Rooms table** (if this table doesn't work try Courses)

Option	Value
Table:	Rooms
Field delimiter:	Tab
End-of-line marker:	CR
Character Set:	Mac Roman
File to import:	<input type="text"/> Browse...
Suggest field map	<input checked="" type="checkbox"/>
School	Example High School

Import

- Browse out to find the template
- Import

Once the Import record from an ASCII table – go to the URL and type in after `html?filenumber=600`

<http://ps-pioneer/admin/importexport/quickimport/quickimport2.html?filenumber=600>

- Enter

The Import record from an ASCII table you should see:

Col#	From your file	----->	To PowerSchool
1.	Value	----->	<input type="text"/>
2.	District Name	----->	<input type="text"/>
3.	cat	----->	<input type="text"/>
4.	Sort Order	----->	<input type="text"/>

Check to exclude first row

The lines below can be used to assign a constant value to field(s) in all of the imported records.

<input type="text"/>	----->	<input type="text"/>
<input type="text"/>	----->	<input type="text"/>

**Match your fields:**

- 1. Value / Value
- 2. District Name / Name
- 3. Cat / Cat
- 4. Sort Order / Sort Order
- Exclude first row
- Submit

## Cleanup In PowerSchool After Importing

- Always check the number of records imported so it matches the spreadsheet. It will be off by one record due to the header record. Also compare these numbers to your SASI export.
- System > System Settings > Reset Server > Submit
- System > Special Operations > Turn Global Sync On > Submit
- System > Special Operations > Set Sync to Atomic Mode > Submit
- System > Security > Users by Groups > select user account and check the district's schools > Submit.
- Setup of Groups first, then restrict pages by going to Access to Page Permissions – make sure you turn it off after using!
- Go to the State/Province-MI, MI General screen and change the Educational Setting for early childhood if you are using a -6. This way your early childhood students will export for the state reporting submission
- Student counts (SASI, PS, SS) - get a student count in SASI, from your spreadsheet and from PowerSchool after importing to you can compare numbers of students before and after.

## Historical Grade Import problem for Term later than 1900

- Tammy explained this problem and has documentation with a workaround. We tested so we could all see it firsthand. Hopefully Pearson will have a fix for this soon. If not we'll have to do this fix for all Phase 2 conversions.

## Credit Types

- If you change credit types in PowerSchool they will show in the converted Historical Grades – BUT the course will not show or calculate on the Graduation Progress Screen.

## Subject Codes and Departments Setup

### Grad Progress/Requirements

- When setting up the grad requirements, the sort order is considered in how Powerschool applies the credits to the subject areas. Courses that have multiple credit types, will be assigned to the credit type that has the highest sort order first. Core classes should be given the highest sort orders. Classes should be listed in groups according to subject and given the priority that you want the credits assigned in.

## Elementary Schedules

- Create the anchor sections
- Create the dependent sections
- Schedule the students into the anchor section – this will schedule them into all the dependent sections.

**Walk-In Scheduling - need to show counselors how to see schedules prior to first day.**

## Lockers

- PowerSchool only stores one locker combination
- The rest of the combinations can be stored in an excel file

## District Entry Date (Modify Info) - PowerSchool

- This field was converted from the SASI OrgEntDate. This field also writes to the State Reporting tab.

## School Entry Date (Modify Info) - PowerSchool

- This field was converted from the SASI Enrollment History.

## Exit Dates

- Had to change the exit dates for all students to the ending date of the calendar which was 6/29/09.

Scheduling Setup Screen

**Scheduling Setup** 

Next Year Grade (Sched\_NextYearGrade) – this field was in the student.txt spreadsheet and imported fine

Schedule This Student (Sched\_Scheduled) – can add this field to the end of the student.txt spreadsheet and mass change the column to a value of 1 or after import use the Student Field Value option and mass change all students to a 1.

Year of Graduation (Sched\_YearOfGraduation) – add this field to the custCvt1 program and use the SASixp field ORIGYRGRAD.

Next School Indicator (next\_school) – this field was in the student.txt spreadsheet and imported fine

**Required Settings**

Next Year Grade

Priority

Schedule This Student

Year of Graduation

Summer School Indicator

Note for Summer School Admin

Next School Indicator

**Optional Settings**

Next Year Campus/Building  Associate

Next Year House  Associate

Next Year Team

## **FTEID Field – need to mass change for all students**

- This field gets populated for the student upon enrollment when a Full-Time Equivalency is selected. The FTEID number comes from the FTE code(s) you created.
- Most likely the school should only have one FTE created.
- Imported students do not have this field populated and this field must be filled in so that ADA/ADM reports work accurately.
- FIND THE FTEID NUMBER
- Go into DDE and select the FTE (159).
- Query YearID = 19 and Search only records in this building.
- Go to List View and open the link and write down the ID number
- Go to the Start Page and key in: FTEID#the number you wrote down. For example FTEID#145
- The students displayed in the list do not have the correct FTEID.
- You can use Student Field Value function to change this field for this group of students.
- Anytime a change is made to attendance preferences or to the FTEID field, you should run the Refresh Attendance Views by going to:  
Special Functions>>Attendance Functions>>Refresh Premier Attendance Views Data.
- Now run the ADA/ADM by Date and/or ADA/ADM by Student to make sure that totals under membership and attendance are accurate.

## **Room File**

- Must be set up once converted to PowerSchool

## **Data Validation (see Data Validation section in binder)**

- **Run Validation Reports in PowerSchool** - these reports should be run daily the first year, by running these reports you will save yourself a lot of headaches.
- Report menu / Section Enrollment Audit Report and School Enrollment Audit Report