**Purpose of this Session**
The implementation of Aeries in a district requires that working knowledge of Microsoft SQL and SQL query language be possessed by the district staff that is acting as the database administrator (DBA). While certification is not required, the working knowledge is invaluable and used by the database administrator on a daily basis. This session is mainly focused on coverage of the most used processes and interfaces of Microsoft SQL. It is not intended to be a training session for usage of Microsoft SQL.

**Table Basics**
A relational database system contains one or more objects called tables. The data or information for the database is stored in these tables. Tables are uniquely identified by their names and are comprised of columns and rows. Columns contain the column name, data type, and other attributes. Rows contain the records or data for the columns. Some of the tables are considered “Parent” tables and have subordinate or “Child” tables that are related (connected) through a common field, known as the “key” field. The parent table contains the main information (student demographics and course information), and the child tables carry supporting information (medical information, grades, and attendance data).

**SQL Server Management Studio**
Management Studio is the graphical interface used to manage SQL servers and databases. The core of your administrative tasks will be performed in Management Studio attaching to the Database Engine. The skill set required of the DBA includes registering and connecting to SQL servers, backing up, restoring, detaching and attaching SQL databases, importing/exporting data, plus setting up and maintaining SQL user security. Upon launching Management Studio, a dialog box for connecting to Server will display on top of the Management Studio interface.

- **Server type** will be Database Engine.
- **Server name** will be local host or the SQL Server you are attaching to.
- **Authentication** should start out as SQL Server Authentication. You may use Windows Authentication after setting up SQL Authentication logins.
- **Login** – use a login setup as a server administrator (such as ‘SA’)
- **Password** – may be blank for the ‘sa’ account upon first login. CHANGE the password from blank to a secure password. **Remember password** is an option and is not required but for security purposes, it is recommended that a strong password be set for all administrator logins.
Once connected to Microsoft SQL Server Management Studio, you may add additional SQL Server instances by clicking on Connect to start the Connect to Server dialog box. You may register additional SQL servers to startup by using Registered Servers (Ctrl+Alt+G).

Create an Aeries Database

Right click on Database, select New Database...

All Aeries databases are created with similar prefix as follows: DSTXX000 where XX is the two-digit school year. The suffix will be the name of your database. Example: DST16000Eagle, DST16000District.
After typing in the name of your database, click on OK. Now that your database is created Aeries tables need to be created in your new database. Click on New Query to open a query tab.

Navigate to the c:\eagle directory and select SQLMODEL.SQL. (FILE | OPEN) SQLMODEL.SQL is the script used to create the Aeries tables. Ensure you are running the script against the new database. Click on Execute or F5 to run the script against the new database.

Verify your tables have been created.
Setting Up Security for Aeries Client Users

The security for Aeries Client utilizes Microsoft SQL Server security. It is required that you know how to setup user accounts and permissions through Management Studio on the SQL server.

Prior to setting up SQL Server Security it is important to determine the different responsibilities of the users. This document will outline some basic set up procedures, and give insight into what tables are required for certain job functions in a typical school district.

One of the benefits of using the SQL server security is that it gives freedom in creating your own “roles”, “security groups” or “job duty tasks”. This will also allow you to customize permissions to a specific job position.

For example, you can create an “Elementary Secretary” role. They will then be given permissions to different roles for the different tables that they will need to access (see below for more details).

PERMISSIONS

The following is a listing of permissions that are used by Aeries when setting up your groups.

- **SELECT** – allows the user /group to select and only display data
- **UPDATE** – allows user /group to update data
- **INSERT** – allows user /group to insert (add) data
- **DELETE** – allows user /group to delete data

Any combination of permissions can be used to fulfill specific requirements. To view what the table is used for, please refer to Table Definitions.

**NOTE:** This document is intended for Database Administrators only

Setting Up User Groups or Roles

The following are general examples of possible job functions and the tables required for the user groups to perform certain tasks. The group list begins with **ALL Aeries Users**. This is a starting point for setting up the user groups or roles and these tables are the minimal tables required to log into Aeries. Also, if the LOG Database Configuration is implemented, required permissions MUST be given to either each group or the group common to all groups.

The View Only group would be determined by the district and would contain various tables that allow users to only display data.

The Admin User group would also be determined by the district and would give the user full control over tables that are considered secure, such as free and reduced.

**ALL AERIES USERS**

- **Must have FULL Control:** LTL, OPT, PRT, USO
- **Must at least have SELECT ONLY:** ABS, BEL, CHI, COD, CRS, DAY, DSD, ENR, IDN, IMM, LOC, LTR, SEC, SSD, STR, STU, SUP, TCH, TFR, TRM, TXT, USR, VAC
- **SELECT, INSERT:** ATL, LOG
- **SELECT, UPDATE, INSERT:** QRY

**ADMIN USERS**

- **FULL Control:** Any Aeries Tables that requires security such as FRE, USR etc.

Adding Roles
Roles, groups or tasks can now be added to your database. Click the mouse on **New Database Role**. The first roles to be created are the **All Aeries Users**, **View Only** and **Admin Users**.

On the **General** page type in the new **Role name** and dbo as the **Owner**.

Click on **Securables**, click on **Search** then select **All** objects of the types. Click on **OK**.
Select **Tables** and click on **OK** to view all tables within the database.

Highlight a table by clicking on it within the **Securables** window. The **Explicit** permissions frame will show you the permissions associated with the table you selected. Aeries uses **SELECT, INSERT, UPDATE** and **DELETE**.

Check mark the appropriate permission under **GRANT**. **DENY** permissions will be discussed later in this documentation.

After the permissions have been set up, the additional **Role Members** can be added. For example, Aeries Users, which are assigned to **ALL** roles. The Aeries 2nd Level Support team can supply sample SQL Scripts to create your Roles, groups or job duty tasks.

**Adding Role Members**

Setup SQL Accounts for Aeries administrators and **CHANGE** the password for the ‘sa’ account. All users will be added to Security whether they access using the preferred method of SQL Authentication or by Windows Authentication. SQL Authentication adds a second layer of security. Windows Authentication is more complex and requires district Network Administrative assistance.
Add a user by navigating to Security, Logins. Left click on Logins to highlight. Right click on Logins or anywhere on the Summary tab located on the right side of the Management Studio window. Left click on New Login.

On the Login – New, select the General page. Type the SQL administrator login name. Select SQL Server authentication. Give the SQL administrator account a secure password. Be aware that SQL passwords are case sensitive. Uncheck Enforce password policy.

Click on the Server Roles page. Give your administrative account permissions to all server roles. Regular users will not belong to any of these server role functions.
Click on the **User Mapping** page. Highlight the Master database. Select **db_owner**. In the example below a database has been created with roles. ALL SQL administrator accounts will belong to **db_owner**, and Aeries Admin roles including the roles of public and Aeries User which **ALL** users will belong. Click **OK** to complete the creation of administrative user account.

![User Mapping Image]

**Field Level Permissions**

Administrators of Aeries SQL Databases have full discretion to implement security of the database. But sometimes to achieve the desired permissions in Aeries for groups of users, the obvious table level permissions are not enough.

Sometimes field/column level permissions are needed. The Administrator **must** have full understanding on how various permissions will affect the users and their access to the Aeries programs. Below are detailed steps in adding **Field Level Security** to a **Role** that has already been setup.

**Adding Field Level Permission**

The example used below in adding **Field Level Security** is for a Counselor that must be allowed to modify a student’s class schedule but not overload the class. Field level security must be added to certain fields in the MST table to give the Counselor certain field level permissions. From the **Management Studio** click the mouse on **Roles** and all roles setup will display on the right-hand side. Double-click the mouse on the **Role** to be updated, such as, Counselor.
Locate and click on the MST table under Securables where Field Level Permissions will be added. Under Explicit permissions for dbo MST: click UPDATE permission to highlight. Click the mouse on Columns Permissions.

The Column Permissions will display. Locate the fields or Columns that additional permissions will be added to. Click the mouse on UPDATE for the Column selected. Ensure you include permissions to the Primary Key fields and the DTS field. For example, adding update to the TB, TG, and TS fields will now give the Counselor Role the ability to modify a student’s class schedule but not overload the class.
Field Level Permission Examples

The following are examples for adding **Field Level Security** and what they will accomplish.

**MODIFY STUDENT CLASS SCHEDULES**

<table>
<thead>
<tr>
<th>Table Permissions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEC – UPDATE, INSERT, DELETE</td>
</tr>
<tr>
<td>CAR – UPDATE, INSERT, DELETE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field/Column – Level Permissions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>MST.TS – UPDATE</td>
</tr>
<tr>
<td>MST.TB – UPDATE</td>
</tr>
<tr>
<td>MST.TG – UPDATE</td>
</tr>
</tbody>
</table>

**NOTE:** Full rights given to the MST table will allow a user, such as a counselor to overload sections. To allow the user to still make schedule changes but not overload sections, the above minimum permissions should be given.

**MODIFY STUDENT FIELDS (ONLY READ PERMISSION TO STU)**

The following examples are for various users that only have READ permission to the STU table but have UPDATE permission to other tables, such as, the MED table. A field may display on a form that they need to update but the data is stored in the STU table.

<table>
<thead>
<tr>
<th>Table Permissions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>STU – READ Only</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field/Column – Level Permissions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>STU.CCG - UPDATE</td>
</tr>
<tr>
<td>*Citizenship GPA – Re-compute HIS (CCG) button</td>
</tr>
<tr>
<td>STU.GCG - UPDATE</td>
</tr>
<tr>
<td>*Citizenship GPA – Re-compute GRD (GCG) button</td>
</tr>
<tr>
<td>STU.HP - UPDATE</td>
</tr>
<tr>
<td>STU.BCY - UPDATE</td>
</tr>
<tr>
<td>STU.BST - UPDATE</td>
</tr>
<tr>
<td>STU.BCU - UPDATE</td>
</tr>
<tr>
<td>*Medical form - Health Problems (HP), Birth City (BCY), State (BST), Country (BCU)</td>
</tr>
<tr>
<td>STU.CU - UPDATE</td>
</tr>
<tr>
<td>*Elementary Classes - “Input” and Move(All/Grade/Tagged) buttons</td>
</tr>
</tbody>
</table>
**STU.NT - UPDATE**  
*Elem... for Next Year - "Input" and Move(All/Grade/Tagged) buttons*

**STU.SEM - UPDATE**  
**STU.PEM - UPDATE**  
*Emergency Contacts - Student Email (SEM), Parent Email (PEM)*

**STU.BM - UPDATE**  
*Photograph Setup - "Update" button*

**STU.HC - UPDATE**  
**STU.EY - UPDATE**  
**STU.HT - UPDATE**  
**STU.WT - UPDATE**  
*ID Card Setup&Print - Photo (BM), Hair (HC), Eye (EY), Height (HT), Weight (WT)*

**STU.FK - UPDATE**  
*Siblings - "Renumber” button*

**STU.SQ – UPDATE**  
*Running Scheduling Course Request/Selection Scanner Sheets Student numbers exceed 9999

**Table Permissions:**

**STU – READ Only**

**STU.PT – UPDATE**  
*Master/Scheduling Master Schedule “Move” option*  
*Student Class Schedules – Upon Add/Delete/Change of Class (to print change locator)*  
*Student Enrollment Form – Add/Delete/Change of Class or Reschedule (to print change locator)*

**STU.RN – UPDATE**  
*Schedule All Students into Classes (From Scheduling Cycle) – option to schedule random*

Field level permissions can be added using a SQL script like the following sample for STU records related to the Medical Form:

**GRANT UPDATE (SC,SN,DTS,HP,BCY,BST,BCU) ON STU TO [ROLE NAME]**

**Deny Permissions (Negative Field Level Security)**

Aeries Client will support negative field-level permissions to be established for individuals or groups of individuals. Previously only affirmative field-level permissions on specific fields were supported. This continues to be the case for affirmative field-level permissions and these fields are listed under Field Level Permissions. Support for negative field-level permissions is only for UPDATE permission. Specific negative READ permissions are NOT supported.

To understand how to properly secure Aeries you must understand how the Aeries Client security model works. This entire process assumes that the SQL System Administrator has already established security permissions for various tables inside the Aeries SQL database. Every time Aeries Client is opened, Aeries reads those permissions through a series of T-SQL queries specifically addressing each table inside the Aeries database.
As permissions for each table are retrieved from the SQL Server the equivalent set of permissions are established on the local Aeries Access Cache Database. This allows the forms inside of Aeries Client to automatically detect the local Access permissions set on the cache database and can lock the tables and forms accordingly. This process works well until we get into security features of SQL Server that MS Access does not support like field-level permissions. In order to support field-level permissions, Aeries makes a few assumptions with the security permissions.

When a SQL user is granted full UPDATE permissions to a table, SQL reports the current SQL user’s permissions as “UPDATE ALL”. But, if a single field cannot be updated because of the system administrator establishing field level restrictions, SQL Server reports the current user’s permissions as “UPDATE SOME”.

In a parallel situation, when a user does NOT have UPDATE permissions to a table but specific field-level UPDATE permissions have been established for a select set of fields, SQL reports the permissions as “UPDATE SOME”.

This makes it difficult for Aeries to establish whether the SQL Administrator intended the user to be able to change the entire record EXCEPT a few fields, or able to change ONLY a select set of fields. The correct answer is important because it translates into specific table-level permissions being established in the Aeries Access cache database.

Aeries determines the difference between the 2 types of “UPDATE SOME” permissions by using a secondary check on the fields that make up the Primary Key in the table. If the current SQL user has permission to change EVERY field that makes up the Primary Key, it is assumed that the user has permissions to the table and only a few fields are restricted. UPDATE permissions are then established on the local cache database. If the user does not, it is assumed they have been granted affirmative permissions and will lock the table in the local cache and rely on specific code in Aeries Client to override those permissions.

Support for negative field-level permissions is done when each form opens by dynamically locking the particular fields that a user does not have permission to change. The record is updatable but the fields the user does not have change permissions to are locked and cannot be changed.

This does have some quirks. If a particular form that allows updating of a table and field that has had specific negative UPDATE permissions set for a field has not been modified to support field level security, changing that particular field will result in errors being generated by the system. In that case, it will be unclear whether the data that was changed was actually saved onto the SQL database.

This situation is most likely to occur using the Aeries Query CHANGE command or using the Aeries Query Change button. It is not anticipated that these features will ever support field-level security, so proper training of your end-users is necessary to avoid any problems or confusion.
**Negative Field Level Security Example**

Create the following Database role called **DenySTU**. Add the **Test** user as a member of the role **DenySTU** group by clicking on **ADD** then browse to the **Test user** and click **OK**.

![Database Role - New](image)

Specific **DENY** permissions are set on the **Test user** to the **STU.LF** and **STU.HL** fields.

![Column Permissions](image)

You may create a SQL Script to add DENY permissions to any existing role.

**DENY UPDATE (LF,HL) ON STU TO [EXISTING ROLE OR GROUP]**

The **Student Data** form in Aeries will “Grey-Out” the specific fields that the user cannot update. Notice the “HmLng” and “LepFep” fields below. These fields cannot be updated through Aeries Client for the Test user.

![Student Data](image)

**Recommended SQL Database Maintenance Processes**

There are many different processes available to keep SQL databases in an “optimum performance” status. These processes are part of the Microsoft SQL Server program and complete documentation is available on the Microsoft SQL Server website and Microsoft Press. Listed below are a few that keep Aeries databases in peak performance status.
**Maintenance Plans**

Maintenance plans should be the INITIAL level of backup and routine maintenance expended on Aeries databases. Maintenance plans give the opportunity to schedule routine Shrinking, Re-indexing, and Performing Incremental and Full Database Back Ups on a scheduled basis.

The suggested plans for you to consider are:

- Maintenance plan that backs up the current year’s database on an hourly basis during the regular school operating time
- Daily maintenance plan that will completely backup the current year’s database and create a “Set” of backup files that can be used to restore data to a specified “Point in Time” status. These files can be then offloaded to a location that can be backed up to a removable media drive, such as a tape drive/library or a SAN setup.
- Maintenance plan that performs a periodic “full” backup of non-current year databases should also be setup. These periodic and incremental backup sets will prove invaluable not only in the event of a catastrophic failure, but also for the common deletion mistakes that are made by district staff, such as removal of a current schedule or replacement of student data such as Free and Reduced status and Program participation records.

**Downloading an Application of Microsoft SQL Server Updates and Service Packs**

Microsoft releases updates and service packs on a periodic basis. These updates and service packs enhance the server performance, as well as apply repaired code for anomalies that are produced by Microsoft SQL Server. The scheduled checking, evaluation and application of updates and service packs from Microsoft will ensure that the Microsoft SQL Server is running at peak performance, which will help to ensure peak performance for Aeries.

**Index Management**

Index management on the Aeries’ SQL databases is very important to maximize performance and to aid in “dropped connection” related issues. The SQL Query script below will assist in the Index Management process by identifying potential candidates for Index Management:

```
DBCC SHOWCONTIG WITH ALL_INDEXES.
```

Any table with a “Pages Scanned” greater than 8 should adhere to the following guidelines:

1. **Scan Density** should be as close to 100% as possible
2. **Logical Scan Fragmentation** should be within the range of 0% and 10%
3. **Extent Scan Fragmentation** should have a value as close to 0% as possible
4. **Average Page Density** should be as close to 100% as possible

Any values conflicting with the 4 guidelines mentioned above will most likely cause “connection timeout” issues. A good guide to resolutions for correcting index fragmentation is available via the internet at the following website:

[http://www.sql-server-performance.com/?s=performance&x=0&y=0](http://www.sql-server-performance.com/?s=performance&x=0&y=0)

The DBCC script process above should be done regularly (once a month for most large districts, 4-5 times per year for small districts) to maintain optimum system performance.

Aeries tables that you should pay special attention to are:

```
IDN, STU, SUP, ENR, ATT, CAR, SEC, CRS, MST, TCH, TST, HIS, GRD, GBK, GBU, GBA, GBS, GBE, GBO, GTG, GBT, LOG
```
Import/Export Data

A useful tool provided with **Microsoft SQL Server – Import/Export Data** can be utilized to export, import, restore, and transfer data from within Microsoft SQL Management Studio. Transactions are performed by creating a “package” that is a script created.

The **SQL Server Import/Export Wizard** box will display. Click the mouse on **Next**.

You will be prompted to **Choose a Source Data**. The source can be several file types: SQL database, MS Access database, MS Excel spreadsheet, or Text file. Additional parameters will be required if an Aeries data file is used (system.mdw credentials).
Select the server to copy data to, use your SQL Server Authentication administrative user account and select the source from which to copy data. Click on **Next**.

The next dialog prompts to **Choose a Destination** for the “packaged” data. The normal for importing will be SQL database.

Select the appropriate SQL Server name, use SQL Server Authentication administrative user account, and the database you are copying data to. Click on **Next**.
The next dialog box prompts to **Specify the Table Copy or Query**. Select the **Copy Tables and View** option and review the graphic that depicts the data flow. Click the mouse on **Next**.

The following dialog box displays all of tables in the source and destination files. It is important to rename any destination differently than the actual table to preserve data integrity.

Since the occurrence of duplicate data is common between source data and data already present in the SQL database table, it is strongly suggested that the temporary table creation process be utilized. A common naming convention technique is to add additional characters to an existing name (e.g. CODxxx for COD data imported for eventual insertion into the COD table). It is also important to choose a temporary naming convention that does not affect Aeries.

It is suggested that the temporary table name is at least 6 characters in length so that Aeries will ignore the presence of the temporary table. When finished, click on **Next**.
The next dialog box allows the user to save the Import/Export package for running later, or allows the package to be run immediately. If the package is saved, it can be recalled and run again at any time.

Clicking on **Finish** on the Completion dialog box to either run or save the Import/Export package. After running the Import/Export package, a dialog box will display showing the progress of the package execution. This will show the creation of the new table and the population of the data. At this point, the data can be imported into an existing Aeries table to complete the import process.

**Import/Export** is most commonly used to import data into Aeries from third party sources, such as Library and School Nutrition programs, test scores, and demographics data. Import/Export process is an excellent tool to restore data that has been corrupted or deleted without a complete restoration of the database. It is also useful for the creation of listings of Aeries data such as spreadsheets, mailing lists, etc. For more complete information as to the capabilities and functions of Import/Export, consult the Microsoft SQL server documentation.

**Adding Fields in SQL Server Management Studio**
From the **SQL Server Management Studio** screen double click the mouse on **Databases**.

![SQL Server Management Studio](image)

A list of all available databases will display. Double click the mouse on the **Database** to be updated.

![Database List](image)

The database name selected will display. Double click the mouse on **Tables**. Scroll Down until the **SUP** table is located and right click on the table name. Click the mouse on **Design**.

![Table Design](image)

**Insert or Delete a field**

To highlight a row, click on the gray box to the left of the field name. After highlighting a row, you can use the **Insert** or **Delete** keys on the keyboard or **Right** click and select to insert or delete a row.

![Table Design](image)

**Remember:** When deleting a field the data in the field will also be deleted.
To add a field

Start by placing your cursor in the last blank **Column Name**. Select the **Data Type**. Deselect **Allow Nulls** for all **Data Types** except for **datetime Fields**.

![Column Name and Data Type Selection](image)

Type your **Column Name** in **upper case letters**. This is the two or three letter name also used in Query.

- Tab to the **Data Type** column. The **Field Properties** box will display below. Use the drop-down arrow to select which type the field will be (remember to select a type supported by Aeries!).
- Tab to the **Allow Nulls** column. De-select "Allow Nulls" for all data types except datetime.

The **Column Properties** box displays at the lower half of the screen. This will look a little different for each **Data Type**. Below are **samples** of the Column Properties for each Data Type.

**To move fields:**

While in the Design form you can move fields by first highlighting the row(s). Next with the mouse pointer pointing to the gray box to the left of the field (which will now have an arrow), hold down the left mouse button and drag the row(s) to the new position.
**varchar or nvarchar**

Allow Nulls should be set to No.

Length – Change to the size needed for the data that will be entered.

Default Value or Binding – Enter two single quotes ("")

Description – Whatever is typed here displays as the field heading on the Supplemental form. Try to fit the heading to the size of the field if possible.

---

**text or ntext**

Allow Nulls should be set to No.

Default Value or Binding Enter two single quotes ("")

Description – Whatever it typed here Displays as the field heading on the SUP form. It will go all the way across the screen and Display two lines of data at a time.

---

**smallint, int, bit, or money**

Allow Nulls should be set to No.

Default Value or Binding Enter ((0))

Note – Bit fields can also default to True ((1))

Description – Whatever is typed here Displays as the field heading on the SUP form. Try to fit the heading to the size of the field.

---

**datetime**

Allow Nulls should be set to Yes.

Default Value or Binding leave blank.

Description – Whatever is typed here will appear as the field heading on the SUP form.
Cut, copy and pasting fields: USE WITH CAUTION!

You can also use the Cut or Copy and Paste method to move fields from place to place after highlighting the rows. When using Cut you will get the message asking if you want to delete the field(s). If you say yes and don’t Paste them in other location, they will be deleted! There is an Undo, but it only restores a certain number of actions.

The Hidden field:

You can define Text fields (varchar, nvarchar, text or ntext) with the Description "Hidden", which will not show on the form, but will allow you to space fields at certain lengths from each other or force a field(s) to go down to the next line. You define this field just like any other text field. You may need to "play" with this field to get the spaces right. You can have more than one Hidden field, but the Field Name must be different (suggestion: H1, H2, etc.).

Drop-Down Menus:

If you want your new fields (or any existing fields this may apply to) to have the drop-down lists, then you need to go to Miscellaneous, then N. Update Code Table. After entering data, fields in the Supplemental form will automatically have the drop down arrows the next time you get into the Supplemental form (remember it is created “on the fly”). Note: Adding drop down lists takes up screen space so your fields will be shifted over to the left and you may need to go back into the Design form to make adjustments.

NOTE: Date and Memo fields do not allow drop-down menus.

When done adding a field or fields:

• Check all entries carefully!

• Click on the Save icon on the toolbar to save changes.

• Log into Aeries.

• Go to the Supplemental form and check the new field or fields you just added (you may need to scroll down). Enter some data to be sure it will accept it. Correcting any errors is easier now while everyone is still out of AERIES. If you need to make corrections or adjustments, follow the directions above to get back to the SUP Design form.

• The macro Clear Null Values From a Table should be run against the SUP table. This will insert default values into any fields that have been added.

Upon completion, all Cache databases MUST be recreated. The easiest way to accomplish this is to click on Force New Cache from the Aeries Control Panel in AdminCS.