Data Warehouse Technical Details

Last Modified on 10/21/2024 8:20 am CDT

Data Storage | Optimization Process | Initial Data Warehouse Configuration | Data Warehouse Backups

> This article is only relevant to districts/states that have enabled Data Warehouse functionality.

This article provides technical details related to the Campus data warehousing process and is intended only for advanced technical users.

Data Storage

Campus uses a normalized entity relationship data model for its transactional/operational database. Data stored in a third normal form (3NF) removes redundant storage of data and optimizes the data structures for efficient read/write transactions so that data elements are only updated in one place to prevent data mismatch.

With the Campus data warehouse, all historical data is kept online and available but is also stored with tables and indexes designed for performance. Pulling data for reporting and analysis involves many joins and complex database operations for value calculations. With the data warehouse, values such as cumulative GPAs or a term's attendance totals are readily available in flat, efficient data structures. Any query or report written for the operational database will run faster and use only a fraction of CPU resources when pulled from the data warehouse.

Optimization Process

Data flattening and creation of calculated fields requires twice the disk space of the operational database. The data warehouse should have at least 20 GB available in excess of this requirement.

> Data warehousing requires a second database server optimized for reporting and analysis. Large districts/states should have a separate server for the data warehouse database. Smaller districts may use a second database on the main server.

The data warehouse refresh process follows these steps in optimizing data for reporting:

1. Optimization starts as an automated backup/restore of the entire operational database.
2. The process cleans and checks data, filtering undesired and inactive records.
3. The process then converts views into tables with indexes for queries.

Initial Data Warehouse Configuration

The initial configuration of the data warehouse is performed by a member of Infinite Campus Hosting staff. A district/state system administrator is then responsible for modifying the details of the views included in the backup process (Data Options tool) and for setting the refresh schedule for data warehouse backups (Refresh Settings tool).

The following steps indicate an example of how a data warehouse is initially configured by Campus personnel.

> This section is for informative purposes only. Campus users SHOULD NOT attempt any of the following steps.

1. Task Scheduler Enabled

The Task Scheduler triggers the scheduled refreshes of the data warehouse database, so it must be enabled.

2. Back Up and Restore of Operational Database

The database backup is restored to the data warehouse server and named appropriately. For example, the Campus2010 database would be named as: *campus2010DW.*

3. Data Warehouse Enabled

The Campus.xml file is modified to include the data warehouse element.

4. Tomcat Restart

Tomcat is restarted and the servers are refreshed. At this point, the data warehouse has been fully enabled behind-the-scenes.

Data Warehouse Backups

The following information provides technical detail on the process used to create a data warehouse backup.

ETL Data Migration

The ETL process converts the operational database to the Campus data warehouse through Extract, Transform and Load (ETL). It looks at the views existing in the database, flattens them and writes them to a new table in the data warehouse.

The ETL process has several places, or "hooks," where it checks for the existence of user-defined, stored procedures. The user can define these hooks in a variety of ways, including:

- Incorporating data from external, third-party applications
- Running clean-up scripts
- Excluding certain tables

Districts/state may implement their own ETL processes for loading external data tabled outside of the Campus application into the data warehouse by using SQL scripts or SQL Server's complete suite of ETL tools.

The ETL hooks available to districts/state include:

- **dw_user_preBackup -** is called on the operational database immediately before it is backed up. If the district/state uses tables in the Campus database to store external data, it can use this to load/refresh that data before backup. The district/state may also perform their own clean/check logic.
- **dw_user_postRestore -** is called on the data warehouse database immediately after the restore finished and before the Campus process cleans data and flattens view to tables. It can be used for the same tasks as preBackup.
- **dw_user_end -** is called after the views have been flattened.

Exceptions/errors generated by these user-created procedures will be logged and the restore process will continue. If a hook does not have a stored procedure defined, it will be skipped.

Excluded Views

Some operational views are not optimized by the data warehouse and, therefore, CANNOT be pulled from the data warehouse for reporting purposes. Tables related to these views will remain virtual, unflattened tables.

- **SIF views** – All SIF views with the prefix of "sif_" are excluded.
- **Legacy views** – Legacy system views with the prefix of "view_" are excluded.
- **Attendance aggregation views** – System-intensive aggregate attendance views are excluded, including:
    - "v_AttDailyDetail"
    - "v_AttDailyExactDetail"
    - "v_AttDailyTermCount"
    - "v_GradebookActivityDetail"
    - "v_AttendanceSectionCount"
    - "v_AttendanceCourseCount"
    - "v_AttendanceSectionTermSummary"
    - "v_SectionRosterAttSummary"
- **Unnecessary state-specific views** – State-specific views that are not applicable to the state are excluded (e.g., a data warehouse in KY would not pull IA state-specific views).
- **Additional State-Edition attendance views** – At the State-Edition level, additional systemintensive aggregate attendance views are excluded, including:
    - "v_MembershipAttendanceDetail"
    - "v_MembershipAttendanceDetailPercent"

- "v_MembershipAttendanceEnrollmentDetail"
- "v_MembershipAttendanceEnrollmentDetailPercent"

Manually Excluding Views

If the district/state will rarely need to analyze certain operational database views, it may exclude them from the data warehouse.

> To exclude a view from the data warehouse, it should be renamed with a prefix of "view_" and it will be skipped.

Views may be excluded from the data warehouse at any time. The data warehouse will recognize excluded views each time it refreshes.

Adding Views to the Data Warehouse

Campus data warehouse logic cycles through existing views and flattens them for use in the data warehouse - unless they fit the exclusion criteria, as described in the previous section.

When additional views are added to the database, they will automatically be available to the data warehouse, unless manually excluded. Views may be created for the data warehouse at any time. The data warehouse will recognize these new views each time it refreshes.

Indexes are automatically created on flattened view tables depending on the fields they contain. If a table has *calendarID* and *personID*, the clustered index will be set to the compound index (*calendarID, personID*). If the table has only one of those fields, the cluster will be a single index.