

Dear RETS Customer,

The Saskatoon Region Association of REALTORS (SRAR) is upgrading their MLS® System software from Matrix (legacy) to Matrix 7.0. With this change, we will also be getting a new RETS server.

NOTE; this change affects all of Saskatchewan

Below is information you need regarding the change.

Timeline

The Matrix 7.0 RETS server will be available starting 12th of June (parallel). The legacy and Matrix 7.0 systems will run in parallel from that point forward. The legacy system, including the RETS server, will be turned off on 3rd of July (cutover).

During the parallel, the Matrix 7.0 RETS server will be updated every 15 minutes. Once the legacy system is turned off, the Matrix 7.0 RETS server will be real-time. Once you make the changes, you are welcome to use Matrix RETS before cutover. If you update more than every 15 minutes, note you may not see changes more often until after cutover.

Please arrange for the resources on your end to make the changes necessary for your products and services to work with the Matrix 7.0 RETS server during the parallel window.

IDs and URL

Your ID and password will remain the same. **IDs and passwords are case sensitive with Matrix 7.0 RETS.**

A User Agent is not necessary to use Matrix 7.0 RETS. If your software needs one, just leave what you had before.

The URL for the new Matrix RETS server is: <http://matrixrets.skmls.ca/rets/login.ashx>

Data Changes

There were various changes made to the data in listings. Below is a synopsis of the major changes. It is best to use the metadata from the server to see all the data changes and lookup values.

Unique ID

In SASK Matrix (legacy) the ID fields were:

Office: MUI

Agent: MUI

Listing: RES, COM, FRM, MUL, LEA: MUI

Media: MUI

OpenHouse: ID

In Matrix 7.0, Matrix_Unique_ID (MUI) exists in every class/resource. Do not use MLS® Number as that may not be unique or may change depending upon the MLS®. The MUI is the primary key in the database and will never change.

See the section below called *Downloading the Entire Set of Listings* for the best way to obtain all listings from the Matrix RETS server.

Matrix_Unique_ID (MUI) is the unique id for each record. If there is a relationship between two resources, you will see another ID that would relate to it with MUI in its name. Example:

Open House

- Matrix_Unique_ID = unique id within Open House data
- Listing_MUI = MUI to the listing resource (cross property).

Listing

- Matrix_unique_ID = unique id within listing data

The join needed to find the data would be OpenHouse.Listing_MUI = Listing.Matrix_Unique_ID.

The same type of logic would be used for other relationships.

- Rooms.Listing_MUI = Listing.Matrix_Unique_ID
- Listing.ListAgent_MUI = Agent.Matrix_Unique_ID
- Media.Table_MUI = Listing.Matrix_Unique_ID (media is where photos are stored)

Classes

Previously, there was one Class per Property Type. With Matrix RETS, there is just one Class – Cross Property. To filter the information, use the field called Property Type. More information about Property Type and other fields that help classify data is discussed below.

Property Types

Property Type is now a database field, not a class. Previously, there were 5 property types.

- Residential
- Farm
- Multi-Family
- Industrial/Commercial
- Lease

In Matrix 7.0, there will be different Property Types. A Property Sub Type has been added to use in your searches. Also, the Transaction Type has been added to distinguish between the Lease and Sale listings for Commercial Property Type. Finally, the Ownership Type field has been added to distinguish between the Condominium, Leasehold and Freehold properties. These changes were made to allow more flexibility with searching for listings, especially across property types. Below is the High-Level layout for SK Matrix 7.0:

- Residential
 - Ownership Type
 - Condominium
 - Leasehold
 - Freehold

 - Property Sub Type
 - Acreage
 - Apartment
 - Attached
 - Detached
 - Mobile

- Lot/Land
 - Ownership Type
 - Condominium
 - Leasehold
 - Freehold

- Farm
 - Property Sub Type
 - Beef
 - Cash Crop
 - Dairy
 - Feed Lot
 - Game
 - Grain
 - Hay
 - Hog
 - Poultry

- Commercial
 - Transaction Type
 - For Sale

- For Lease
- Ownership Type
 - Condominium
 - Leasehold
 - Freehold
- Property Sub Type
 - Business
 - Hotel/Motel
 - Industrial
 - Land
 - Mixed
 - Office
 - Other
 - Retail
 - Warehouse
- Multi-Family
 - Ownership Type
 - Condominium
 - Leasehold
 - Freehold
 - Property SubType
 - Residential & Commercial
 - Residential & Offices
 - Residential Only
 - Residential/Commercial & Offices
- Cross Property
 - Property Type
 - Residential
 - Commercial
 - Farm
 - Lot/Land
 - Multi-Family
 - Transaction Type
 - For Lease
 - For Sale

- Ownership Type
 - Condominium
 - Leasehold
 - Freehold

- Property SubType
 - Acreage
 - Apartment
 - Attached
 - Beef
 - Business
 - Cash Crop
 - Dairy
 - Detached
 - Feed Lot
 - Game
 - Grain
 - Hay
 - Hog
 - Hotel/Motel
 - Industrial
 - Land
 - Mixed
 - Mobile
 - Office
 - Other
 - Poultry
 - Residential & Commercial
 - Residential & Office
 - Residential Only
 - Residential/Commercial & Offices
 - Retail
 - Warehouse

Status

Previously, there were the following Statuses for listings:

Name	RValue	Description
Active	AC	Listings for sale with no contract
ConditionalSale	CC	Listings for sale with accepted contract
Sold	SO	Listings sold and closed
Cancelled	CA	Listings that are Cancelled

Expired	EX	Listings with listing period expired
Withdrawn	WI	Listings withdrawn prior to expiration

In SK Matrix 7.0, the following Statuses will be available:

Name	RValue	Description
Incomplete	I	Listings entered but not active
Active	A	Listings for sale with no contract
ConditionalSale	C	Listings for sale with accepted contract
Sold	S	Listings sold and closed
Cancelled	K	Listings that are Cancelled
Expired	X	Listings with listing period expired
Withdrawn	W	Listings withdrawn prior to expiration

Incomplete is a new status

Rooms

Previously, the information about Rooms were flattened out in each listing and limited the system to only track information about a certain number of rooms. Rooms in Matrix are considered a separate data set and are unlimited due to this.

Matrix_Unique_ID (MUI) is the unique id for each record. If there is a relationship between two resources, you will see another ID that would relate to it with MUI in its name. For rooms, the relationship is Rooms.Listing_MUI = Listing.Matrix_Unique_ID

Parcels

Previously, the information about Parcels was flattened out in each listing and limited the system to only track information about certain parcels. Parcels in Matrix are considered a separate data set and are unlimited due to this.

Matrix_Unique_ID (MUI) is the unique id for each record. If there is a relationship between two resources, you will see another ID that would relate to it with MUI in its name. For parcels, the relationship is Parcels.Listing_MUI = Listing.Matrix_Unique_ID

Media

Media contains all images or documents stored in a Matrix system. They can relate to listings or to other resources.

For media and listings, the relationship is Media.Table_MUI = Listing.Matrix_Unique_ID.

There is a field called PhotoCount in the listing tables that indicates the number of photos for that particular listing.

We have added a field to the listings called PhotoModificationTimestamp. This field contains a modification timestamp for the photos for a listing. So if any new pictures are added, removed or replaced the listing's PhotoModificationTimestamp would be changed as well as the MatrixModifiedDT.

You can pull photo updates for listings using a similar method for pulling listing updates, but instead of using MatrixModifiedDT you can PhotoModificationTimestamp. When listings have a newer PhotoModificationTimestamp, you need to download the photos for that listing. Alternatively, you could use a different method whereby when you download updates for a listing, you can compare the old PhotoModificationTimestamp with the new one. If they differ, re-download the photos for that listing. This method works because the MatrixModifiedDT is updated at the same time as the PhotoModificationTimestamp.

For agent photos there is a relationship of Media.Table_MUI = Agent.Matrix_Unique_ID. Agent photos have an Object Type of Agent. If Agent Photos exist, there will be a PhotoModificationTimestamp in the Agent records similar to the one noted above for listings. Only one photo per Agent is stored.

See the section below on Downloading Photos for specifics on how to obtain all media images.

Below are frequently asked questions regarding the Media resource.

- How do I obtain the largest resolution images? Use the Object Type of Large Photo.
- Do I have the ability to directly link to the images on your server opposed to downloading the images to my local server? No, you must download the media to your server.
- How many photos are allowed per listing? The limit is 40 with Matrix.
- What is the best way to download photos? See the section below called Downloading Photos

Media Padding

In SK Matrix 7.0, images are no longer padded with white margins to maintain a uniform size. Instead, they will be served as close to the requested size as possible, without stretching the original image. This will not be especially visible within Matrix because Matrix will still display images with padding, but it will affect RETS users who may be used to standardized images.

For example:

If you request an image at a size of 640x480px, you will get an image that will be no wider than 640 pixels, and no taller than 480 pixels, but it will not be exactly 640x480 (unless the original image meets those exact dimensions).

If the original image was only 100x100, then that's the largest size you'll receive.

Downloading the Entire Set of Listings

There is a limit of 5,000 records that can be downloaded at one time with Matrix RETS. You can use the Offset feature if you have the need to obtain more than 5,000 records. Below is what is recommended for you to do to populate the initial load:

1. Take note of the time that you start this process - it will be used later.
2. Use a query of (matrix_unique_id = 0+)&Limit=1000
3. Retrieve all those listings
4. Find the highest matrix_unique_id from those listings
5. Create a new query with (matrix_unique_id=<HighestID from step 4>+)&Limit=1000
6. Repeat from Start step 3 until you receive all listings.
7. The general download of listings using the dt_mod equal to the date you got in step 1. This makes sure to get any changes you may have missed while getting the entire set of listings.

This process works because all of our output is ordered by the primary key, the Matrix_unique_Id field. The RETS spec does not allow a client to specify an order.

Matrix RETS servers will have only one property resource called Cross Property. Typically, fields such as PropertyType, PropertySubType and TransactionType will indicate if a listing is Residential, Commercial or a Rental. Check with your MLS® as to what their data fields are that identify these key fields.

Downloading Photos

Matrix does not provide URLs to the photos, they must be downloaded. Downloading photos in RETS is performed through the GetObject request. Here is an example of the GetObject request to get an image for a property. It is a sample of the code to retrieve an actual photo.

<http://rets.hgmls.mlsmatrix.com/rets/GetObject.ashx?Type=Photo&Resource=Property&ID=269251:0>

The ID portion of the parameters (ID=269251:0) has two parts.

- The 269251 is the Matrix_Unique_ID of the listing, our primary key, as noted in the METADATA-RESOURCE. It is not the MLS® number. This has caught other users in the past.
- The “:0” of the ID is used to get the primary picture.
 - A “:1” will get you the first picture, which on our system is the primary picture, so in effect, :0 and :1 are equivalent.
 - Using “:2” will give you the second picture.
 - If you use a “:*” you will receive a multipart response which contains all of the images for that property.

There is no request in RETS to simply ask for ALL of the images.

There is a field called PhotoCount in the listing tables that indicates the number of photos for that particular listing.

We have added a field to the listings called PhotoModificationTimestamp. This field contains a modification timestamp for the photos for a listing. So if any new pictures are added, removed or replaced the listing's PhotoModificationTimestamp would be changed as well as the MatrixModifiedDT.

You can pull photo updates for listings using a similar method for pulling listing updates, but instead of using MatrixModifiedDT you can PhotoModificationTimestamp. When listings have a newer PhotoModificationTimestamp, you need to download the photos for that listing. Alternatively, you could use a different method whereby when you download updates for a listing, you can compare the old PhotoModificationTimestamp with the new one. If they differ, re-download the photos for that listing. This method works because the MatrixModifiedDT is updated at the same time as the PhotoModificationTimestamp.