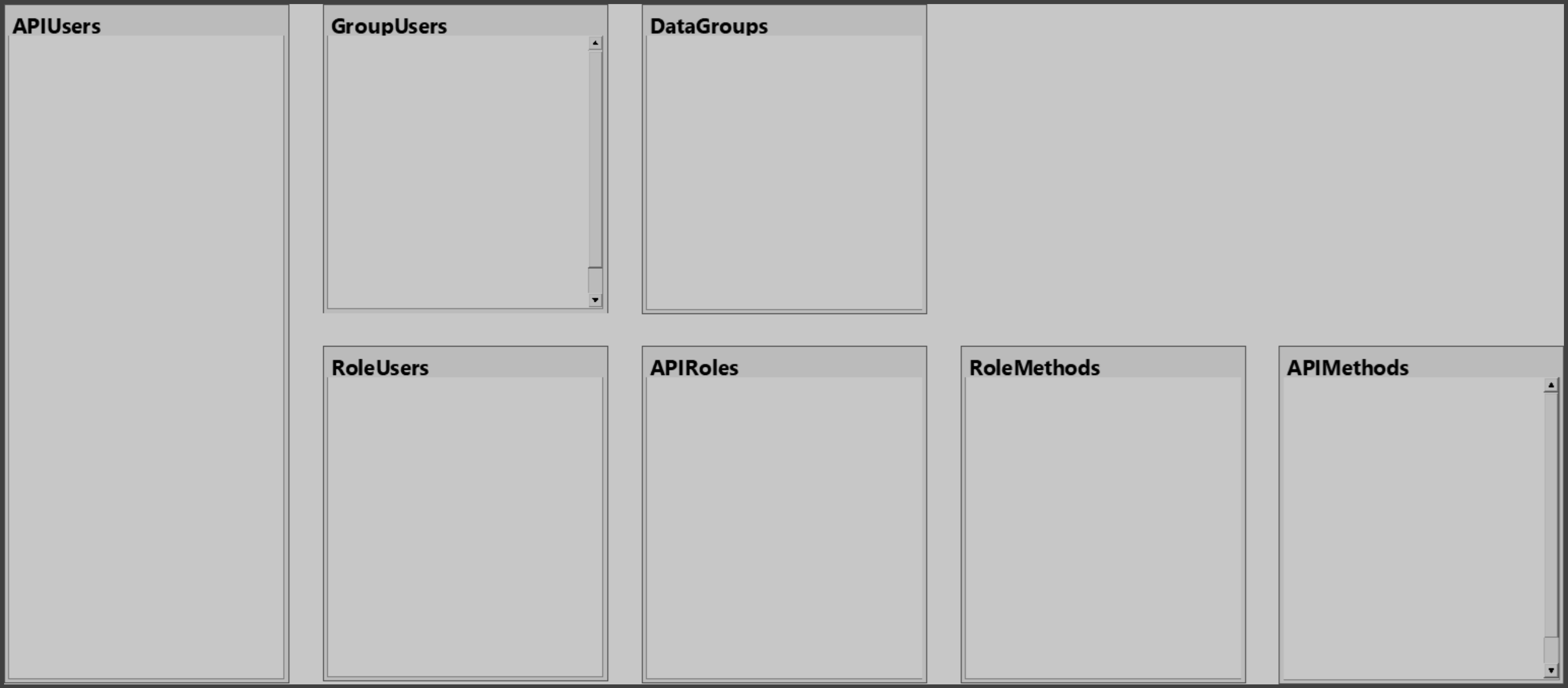
Security Forms Overview

One of the key parts of DGP Lattice security is the data-driven Role Based Access Control (RBAC) subsystem that controls not only user authentication (via HMAC hash) but also which accounts can call which API methods (APIMethod authorization) and which accounts can see which data in Lattice (DataGroup authorization). The heart of that subsystem is the SQL Server SysInfo database that stores all of the security tables and records. The security forms in the Lattice UI are used by system admins to manage and maintain all of the various security records.



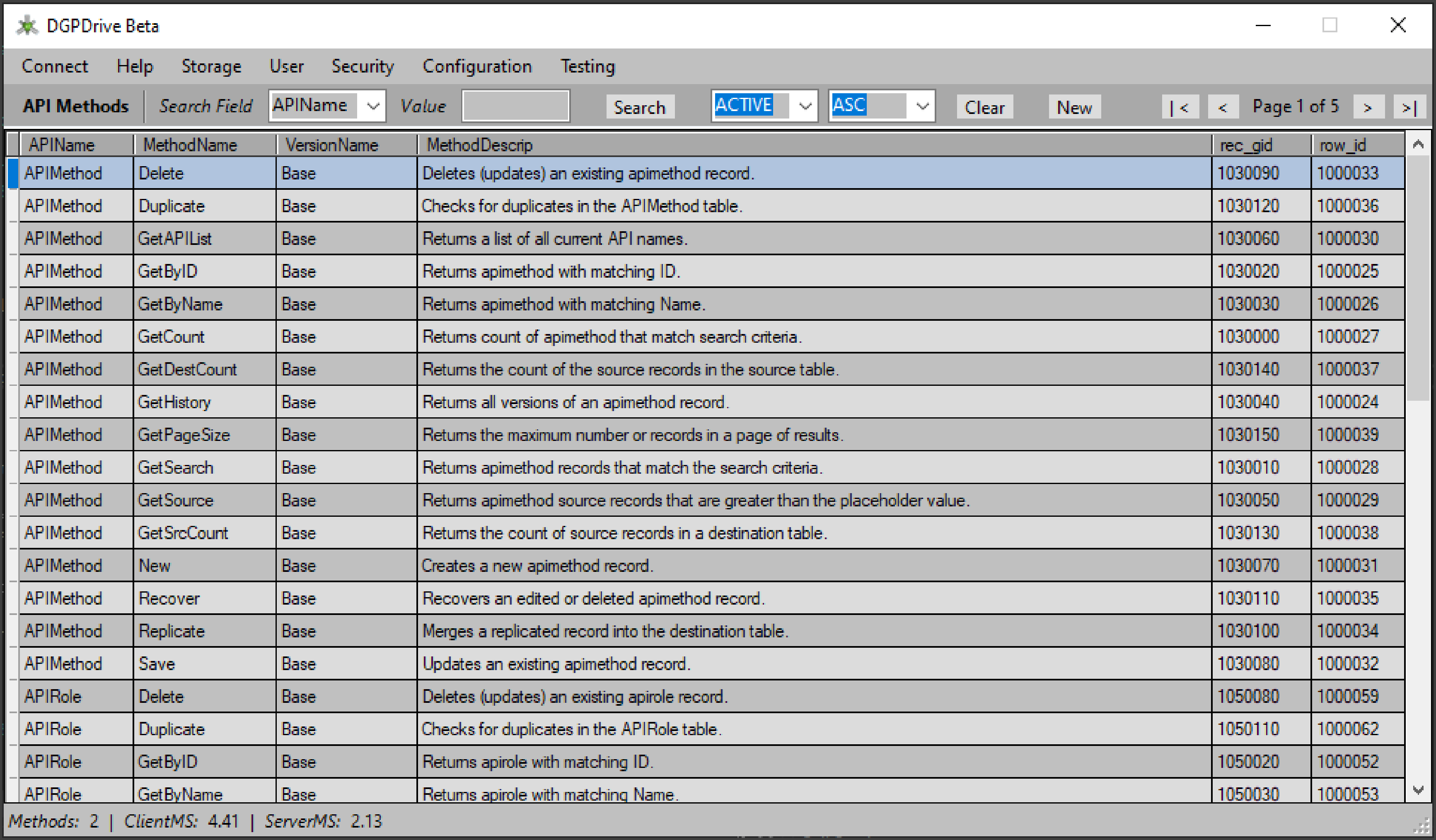
Each of the 4 forms that correspond to the 4 main entity tables in the SysInfo schema all have the same layout and functionality.

A toolstrip across the top of the form contains a dropdown to select a search field, a textbox for the search value, Search and Clear buttons, dropdowns for the number of page rows and the record state to search for, a button to add a new record, and the buttons to navigate pagination.

API Methods Form

Each APIMethod in the system must be documented in the APIMethods table or it cannot be linked to a role, in which case it cannot be authorized for any user to call it. User authentication and API method authorization functionality is built into each DGP web service, and there are no ways to bypass that logic, even for admin users.

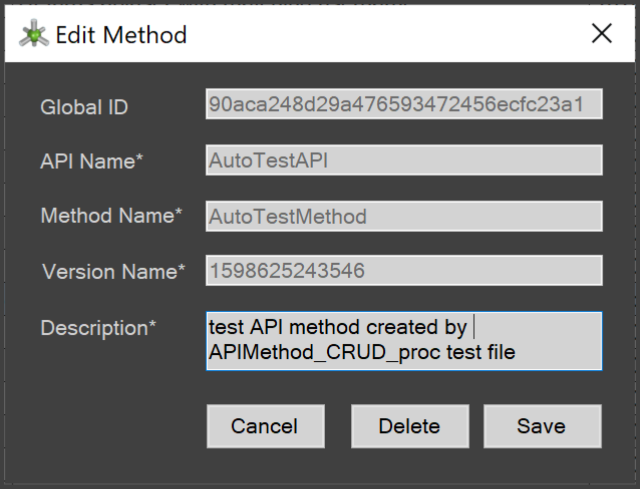
APIMethods Context Menu	Description
Edit…	Only allows the description field of the APIMethod record to be edited. This restriction is due to the “immutable, append-only” convention that strictly enforces backward compatibility. Editing any other APIMethod values would cause breaking changes to applications and systems that depend on the existing state to function correctly.
Method Roles…	This is a simple list of the roles that the selected APIMethod has been assigned to.
View History…	This form shows a grid of all versions of the selected record. If the record state is set to ACTIVE, the Recover Selected Record button is enabled to recover an edited version of the record as the current ACTIVE version. The process to recover a deleted record is different, so the Recover Selected Record button is disabled when the record state is set to DELETED.
Recover Deleted	This option is disabled unless the record state has been set to DELETED. Doing so allows the form to be used to search for deleted records. Selecting this option for a deleted record will recover the deleted record as the new current ACTIVE record.



Search Field APIName Value ACTIVE ASC Page 1 of 5

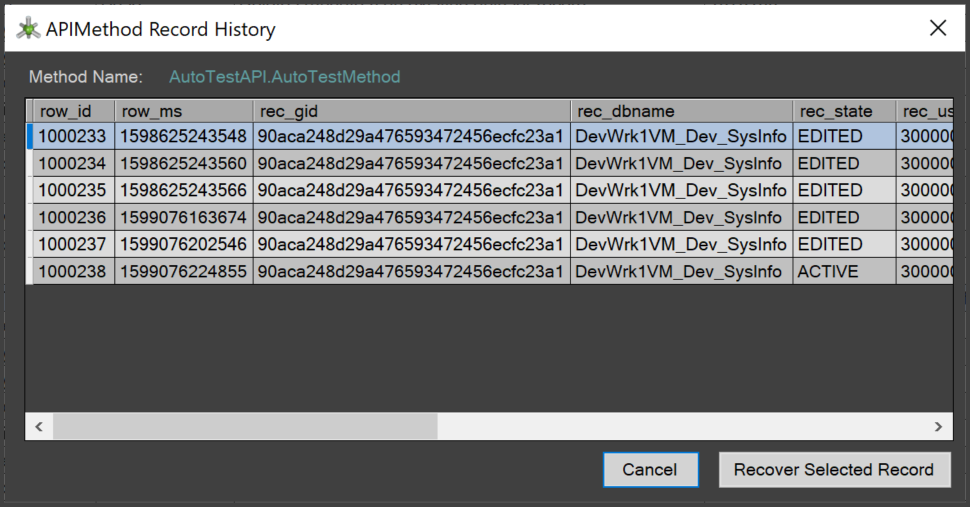
APIName	MethodName	VersionName	MethodDescrip	rec_gid	row_id
APIMethod	Delete	Base	Deletes (updates) an existing apimethod record.	1030090	1000033
APIMethod	Duplicate	Base	Checks for duplicates in the APIMethod table.	1030120	1000036
APIMethod	GetAPIList	Base	Returns a list of all current API names.	1030060	1000030
APIMethod	GetByID	Base	Returns apimethod with matching ID.	1030020	1000025
APIMethod	GetByName	Base	Returns apimethod with matching Name.	1030030	1000026
APIMethod	GetCount	Base	Returns count of apimethod that match search criteria.	1030000	1000027
APIMethod	GetDestCount	Base	Returns the count of the source records in the source table.	1030140	1000037
APIMethod	GetHistory	Base	Returns all versions of an apimethod record.	1030040	1000024
APIMethod	GetPageSize	Base	Returns the maximum number or records in a page of results.	1030150	1000039
APIMethod	GetSearch	Base	Returns apimethod records that match the search criteria.	1030010	1000028
APIMethod	GetSource	Base	Returns apimethod source records that are greater than the placeholder value.	1030050	1000029
APIMethod	GetSrcCount	Base	Returns the count of source records in a destination table.	1030130	1000038
APIMethod	New	Base	Creates a new apimethod record.	1030070	1000031
APIMethod	Recover	Base	Recovers an edited or deleted apimethod record.	1030110	1000035
APIMethod	Replicate	Base	Merges a replicated record into the destination table.	1030100	1000034
APIMethod	Save	Base	Updates an existing apimethod record.	1030080	1000032
APIRole	Delete	Base	Deletes (updates) an existing apirole record.	1050080	1000059
APIRole	Duplicate	Base	Checks for duplicates in the APIRole table.	1050110	1000062
APIRole	GetByID	Base	Returns apirole with matching ID.	1050020	1000052
APIRole	GetByName	Base	Returns apirole with matching Name.	1050030	1000053

Methods: 2 | ClientMS: 4.41 | ServerMS: 2.13



Edit Method

Global ID	90aca248d29a476593472456ecfc23a1
API Name*	AutoTestAPI
Method Name*	AutoTestMethod
Version Name*	1598625243546
Description*	test API method created by APIMethod_CRUD_proc test file



APIMethod Record History

Method Name: AutoTestAPI.AutoTestMethod

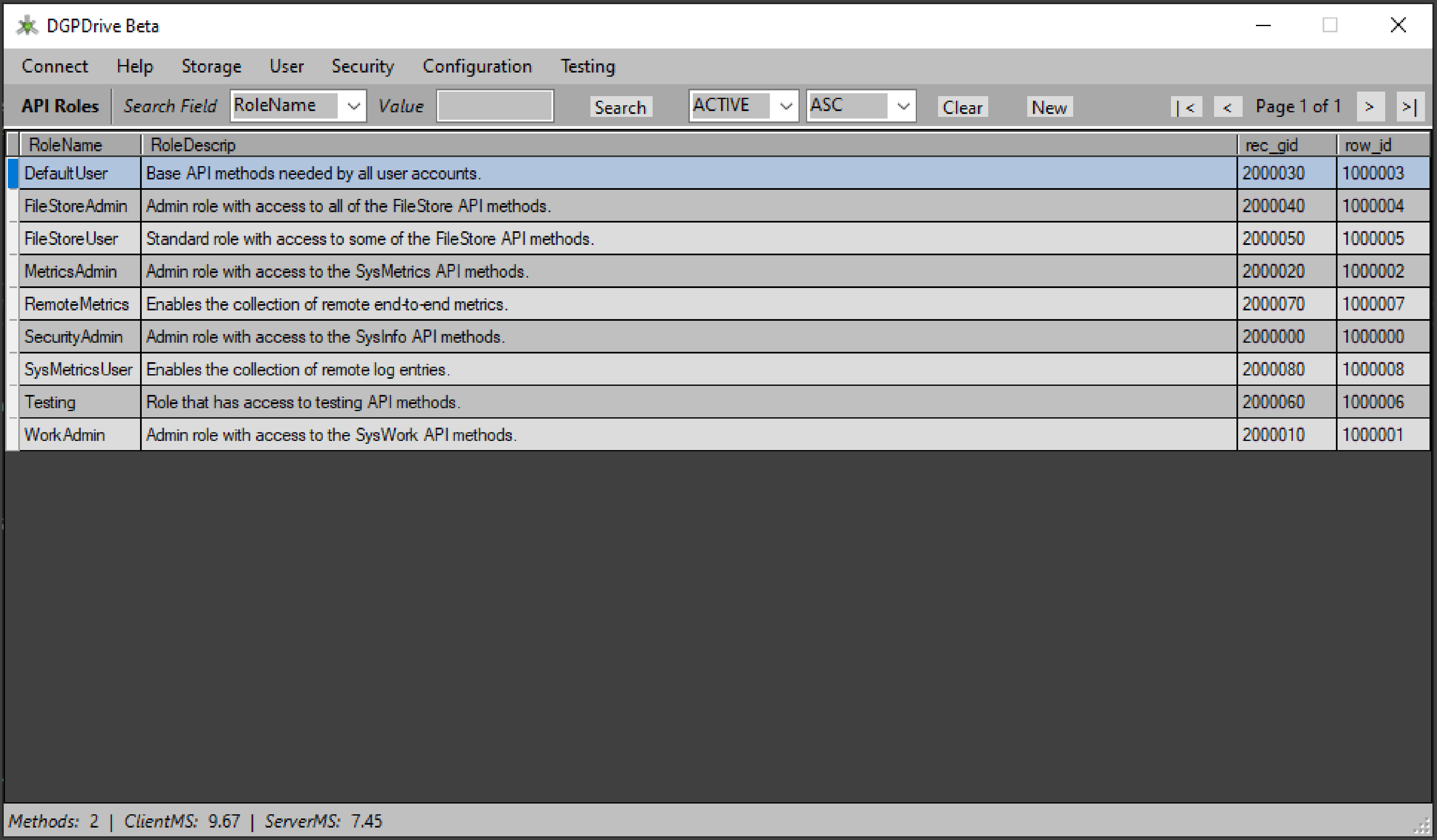
row_id	row_ms	rec_gid	rec_dbname	rec_state	rec_us
1000233	1598625243548	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000234	1598625243560	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000235	1598625243566	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000236	1599076163674	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000237	1599076202546	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000238	1599076224855	90aca248d29a476593472456ecfc23a1	DevWrk1VM_Dev_SysInfo	ACTIVE	30000

API Roles Form

APIRoles are used to create collections of APIMethods that are easier to assign to users than would be the case if each method had to be assigned to each user individually. That is the main point of “Role Based Access Control” (RBAC). There is no limit to the number of roles in a system, and no limit to the number of roles that can be assigned to a user. The user account will end up with a superset of all the methods authorized by each role, without any duplicates.

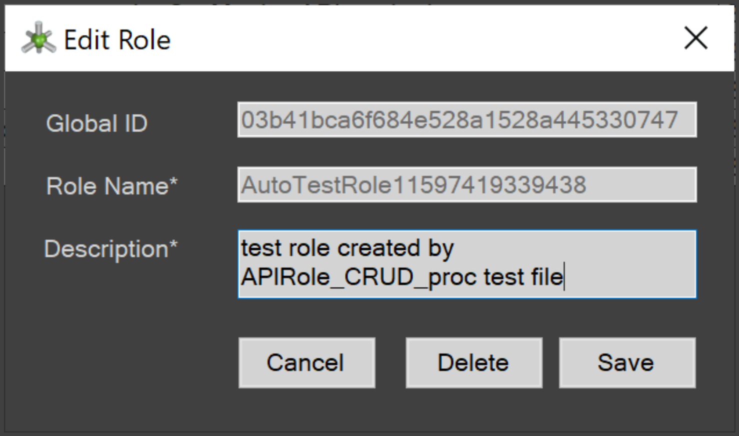
APIRoles Context Menu	Description
Edit…	Only allows the description field of the APIRole record to be edited. This restriction is due to the “immutable, append-only” convention that strictly enforces backward compatibility. Editing any other APIRole values would cause breaking changes to applications and systems that depend on the existing state to function correctly.

Role Methods…	This form creates "intersection" records assigning (linking) APIMethod records to one or more APIRoles. The form shows the methods currently assigned to the role in the left-hand grid. Double-clicking on an assigned method removes (soft deletes) the intersection record. Due to the large number of APIMethod records, the Available Methods section on the right shows a parent/child set of grids. Selecting one of the API's shows all of the API methods that have not yet been assigned to the role (in some cases the list of available methods will be empty if all methods have already been assigned). Double-clicking on an available method will assign it to the role.
View History…	This form shows a grid of all versions of the selected record. If the record state is set to ACTIVE, the Recover Selected Record button is enabled to recover an edited version of the record as the current ACTIVE version. The process to recover a deleted record is different, so the Recover Selected Record button is disabled when the record state is set to DELETED..
Recover Deleted…	This option is disabled unless the record state has been set to DELETED. Doing so allows the form to be used to search for deleted records. Selecting this option for a deleted record will recover the deleted record as the new current ACTIVE record.

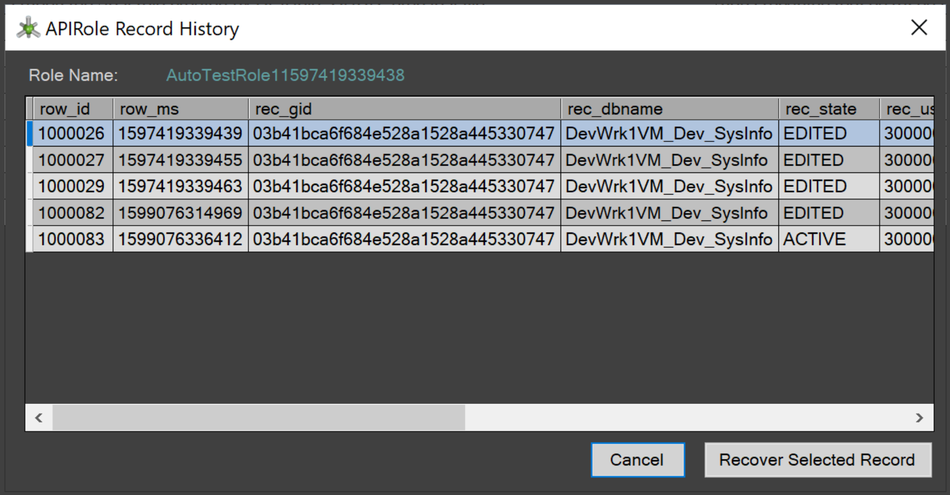


RoleName	RoleDescrip	rec_gid	row_id
DefaultUser	Base API methods needed by all user accounts.	2000030	1000003
FileStoreAdmin	Admin role with access to all of the FileStore API methods.	2000040	1000004
FileStoreUser	Standard role with access to some of the FileStore API methods.	2000050	1000005
MetricsAdmin	Admin role with access to the SysMetrics API methods.	2000020	1000002
RemoteMetrics	Enables the collection of remote end-to-end metrics.	2000070	1000007
SecurityAdmin	Admin role with access to the SysInfo API methods.	2000000	1000000
SysMetricsUser	Enables the collection of remote log entries.	2000080	1000008
Testing	Role that has access to testing API methods.	2000060	1000006
WorkAdmin	Admin role with access to the SysWork API methods.	2000010	1000001

Methods: 2 | *ClientMS:* 9.67 | *ServerMS:* 7.45

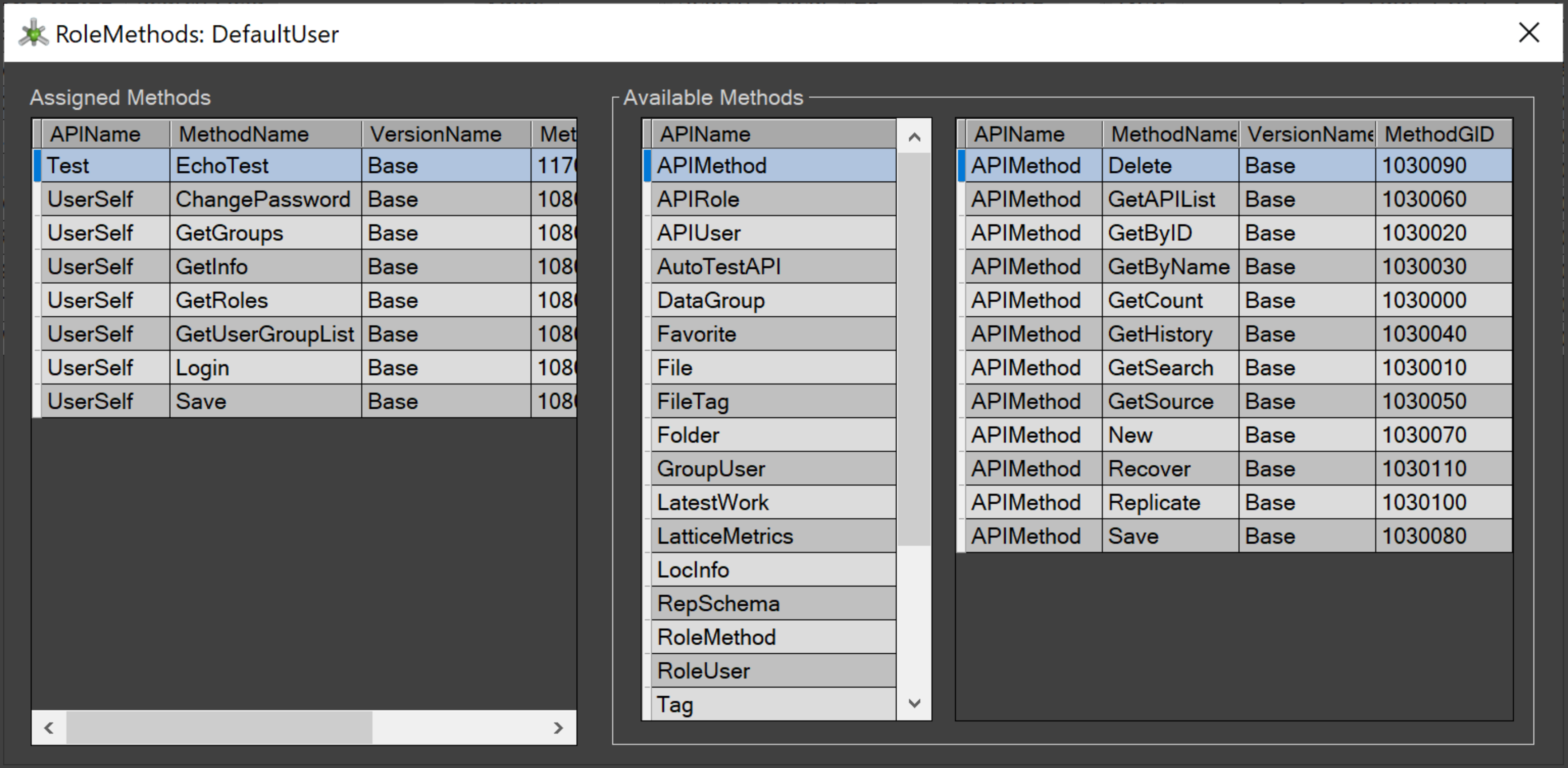


Global ID	03b41bca6f684e528a1528a445330747
Role Name*	AutoTestRole11597419339438
Description*	test role created by APIRole_CRUD_proc test file



Role Name: AutoTestRole11597419339438

row_id	row_ms	rec_gid	rec_dbname	rec_state	rec_us
1000026	1597419339439	03b41bca6f684e528a1528a445330747	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000027	1597419339455	03b41bca6f684e528a1528a445330747	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000029	1597419339463	03b41bca6f684e528a1528a445330747	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000082	1599076314969	03b41bca6f684e528a1528a445330747	DevWrk1VM_Dev_SysInfo	EDITED	30000
1000083	1599076336412	03b41bca6f684e528a1528a445330747	DevWrk1VM_Dev_SysInfo	ACTIVE	30000



RoleMethods: DefaultUser

Assigned Methods

APIName	MethodName	VersionName	Met
Test	EchoTest	Base	117
UserSelf	ChangePassword	Base	108
UserSelf	GetGroups	Base	108
UserSelf	GetInfo	Base	108
UserSelf	GetRoles	Base	108
UserSelf	GetUserGroupList	Base	108
UserSelf	Login	Base	108
UserSelf	Save	Base	108

Available Methods

APIName
APIMethod
APIRole
APIUser
AutoTestAPI
DataGroup
Favorite
File
FileTag
Folder
GroupUser
LatestWork
LatticeMetrics
LocInfo
RepSchema
RoleMethod
RoleUser
Tag

APIName	MethodName	VersionName	MethodGID
APIMethod	Delete	Base	1030090
APIMethod	GetAPIList	Base	1030060
APIMethod	GetByID	Base	1030020
APIMethod	GetByName	Base	1030030
APIMethod	GetCount	Base	1030000
APIMethod	GetHistory	Base	1030040
APIMethod	GetSearch	Base	1030010
APIMethod	GetSource	Base	1030050
APIMethod	New	Base	1030070
APIMethod	Recover	Base	1030110
APIMethod	Replicate	Base	1030100
APIMethod	Save	Base	1030080

API Users Form

User account records are stored in the APIUsers table, and in addition to all the standard fields containing details of the user, it also contains cached authorization data for APIMethods and DataGroups, as well as an API method rate limit value per account.

APIUsers Context Menu	Description
Edit…	The form allows most user values to be edited with the exception of the UserName and Password fields. The password value is not shown in the UI (and in any event is stored in encrypted form in the database record using zero-knowledge encryption).
User Methods…	This form shows a simple grid of the superset of all methods that a user is authorized to call from the combined total of all their role memberships. A list of all the method names in this grid is cached in the APIUser record for API authorization, and updated every time the user calls the Login method.
User Roles…	This form creates "intersection" records assigning (linking) APIUser records to one or more APIRoles. The form shows the roles currently assigned to the user in the left-hand grid. Double-clicking on an assigned role removes (via replica record update) the intersection record. The Available Roles section on the right shows all of the API roles that have not yet been assigned to the user. Double-clicking on an available role will assign it to the user.
Group Access…	This form creates "intersection" records assigning (linking) APIUser records to one or more DataGroups. The form shows the groups currently assigned to the user in the left-hand grid. Double-clicking on an assigned group removes (soft deletes) the intersection record. The Available Groups section on the right shows all of the DataGroups that have not yet been assigned to the user. The appropriate access level (ReadOnly or ReadWrite) must be selected before Double-clicking on an available group to assign it to the user.

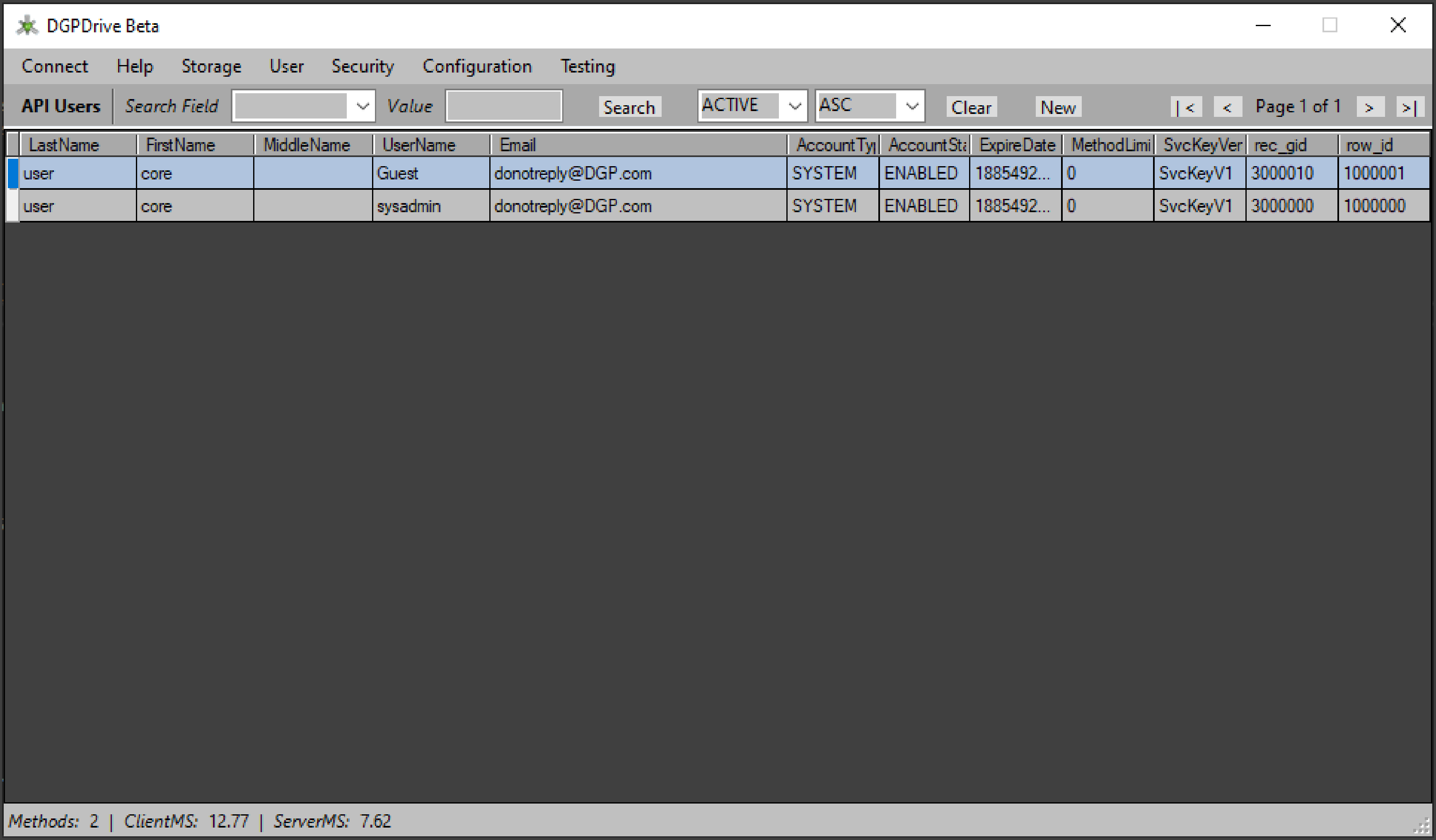
Change Password…	This form calls an API method that runs a process to change a user's password, store it in encrypted form in the database record, and set a new expiration date based on the configured expiration days.
View History…	This form shows a grid of all versions of the selected record. If the record state is set to ACTIVE, the Recover Selected Record button is enabled to recover an edited version of the record as the current ACTIVE version. The process to recover a deleted record is different, so the Recover Selected Record button is disabled when the record state is set to DELETED.
Recover Deleted…	This option is disabled unless the record state has been set to DELETED. Doing so allows the form to be used to search for deleted records. Selecting this option for a deleted record will recover the deleted record as the new current ACTIVE record.

Note: SYSTEM accounts never expire and have no rate limit, since they bypass the expiration check and the rate limit check in the web service (which means any expiration or rate limit values set for system accounts have no effect). Expiration date and rate limit values only affect standard user accounts.

To improve the performance of account authentication and authorization used when processing each API request message, authorization data is cached in each user account record. This means that the authentication query only searches for and retrieves a single record from a single well-indexed table.

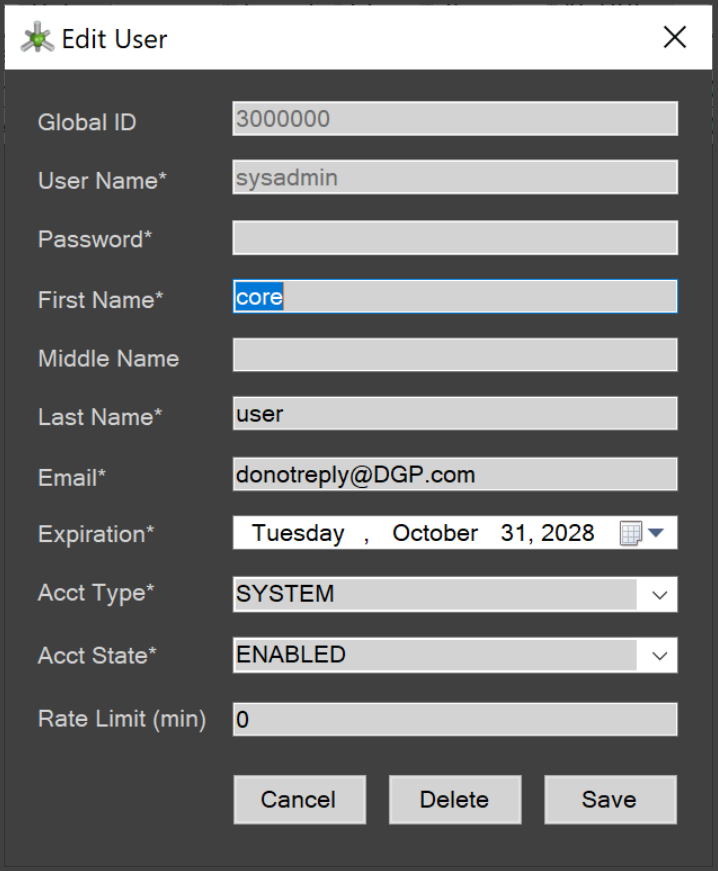
The cached data is updated via a lazy just-in-time process that runs as part of the Login API method for standard accounts. This is to help spread out the workload when very large numbers of user accounts are affected by a change to the data, such as adding a new method linked to a role that already has many users.

The authorization data cached in system accounts is updated whenever the parent tables such as APIRoles, DataGroups, etc. are changed, under the assumption that there will only be a small number of system accounts, and the workload of updating all of them at once will not bog down the system.



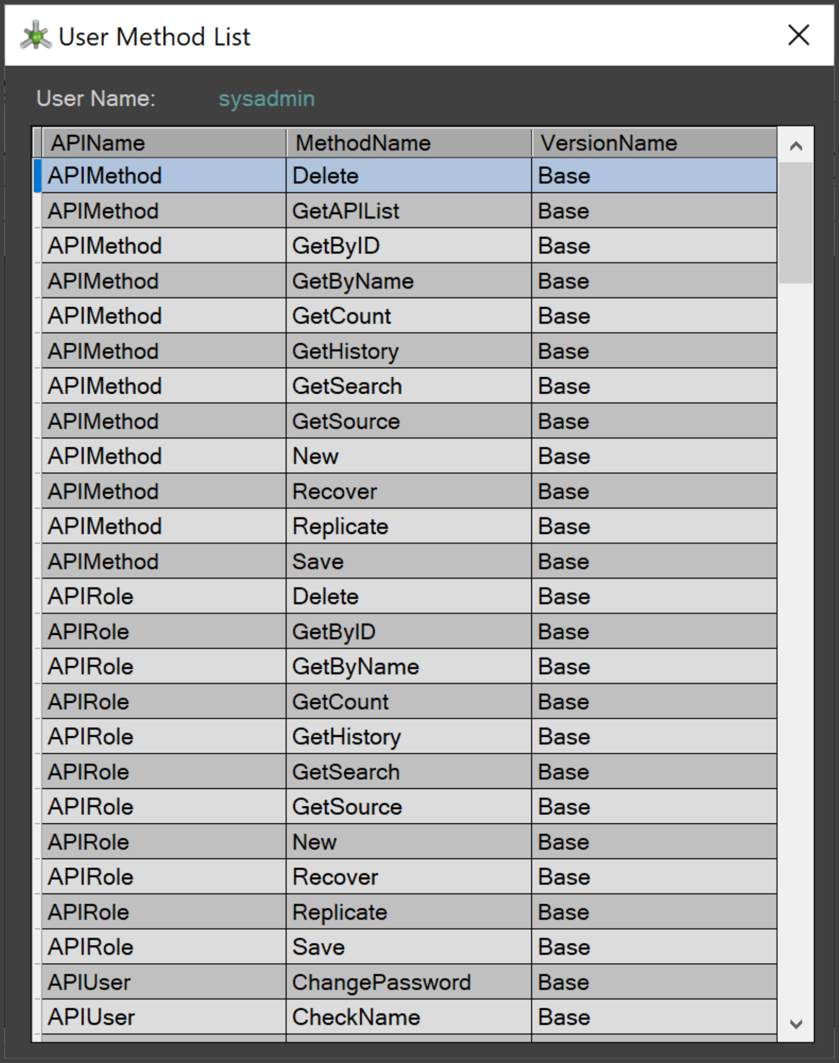
LastName	FirstName	MiddleName	UserName	Email	AccountTy	AccountSt	ExpireDate	MethodLimi	SvcKeyVer	rec_gid	row_id
user	core		Guest	donotreply@DGP.com	SYSTEM	ENABLED	1885492...	0	SvcKeyV1	3000010	1000001
user	core		sysadmin	donotreply@DGP.com	SYSTEM	ENABLED	1885492...	0	SvcKeyV1	3000000	1000000

Methods: 2 | *ClientMS:* 12.77 | *ServerMS:* 7.62



Edit User

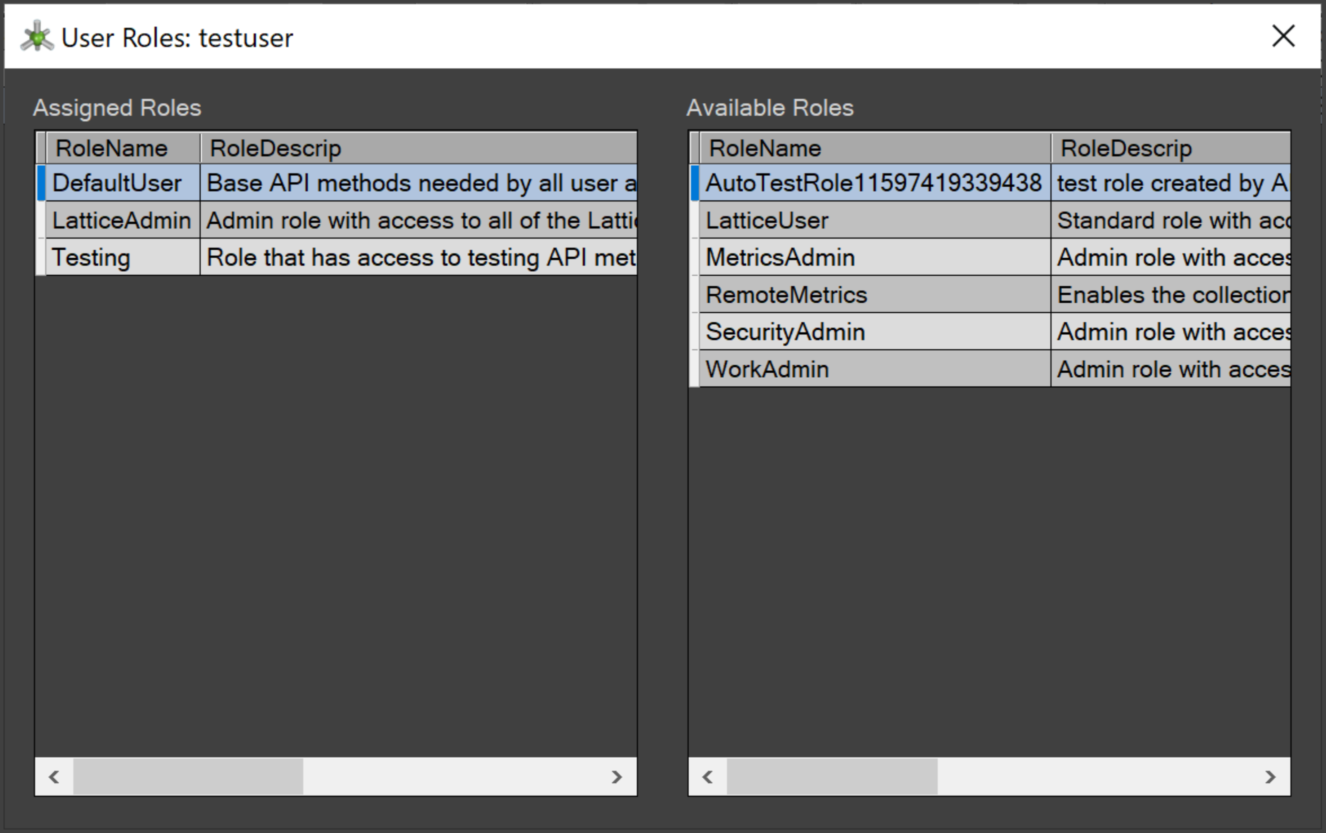
Global ID	3000000
User Name*	sysadmin
Password*	
First Name*	core
Middle Name	
Last Name*	user
Email*	donotreply@DGP.com
Expiration*	Tuesday , October 31, 2028
Acct Type*	SYSTEM
Acct State*	ENABLED
Rate Limit (min)	0



User Method List

User Name: sysadmin

APIName	MethodName	VersionName
APIMethod	Delete	Base
APIMethod	GetAPIList	Base
APIMethod	GetByID	Base
APIMethod	GetByName	Base
APIMethod	GetCount	Base
APIMethod	GetHistory	Base
APIMethod	GetSearch	Base
APIMethod	GetSource	Base
APIMethod	New	Base
APIMethod	Recover	Base
APIMethod	Replicate	Base
APIMethod	Save	Base
APIRole	Delete	Base
APIRole	GetByID	Base
APIRole	GetByName	Base
APIRole	GetCount	Base
APIRole	GetHistory	Base
APIRole	GetSearch	Base
APIRole	GetSource	Base
APIRole	New	Base
APIRole	Recover	Base
APIRole	Replicate	Base
APIRole	Save	Base
APIUser	ChangePassword	Base
APIUser	CheckName	Base



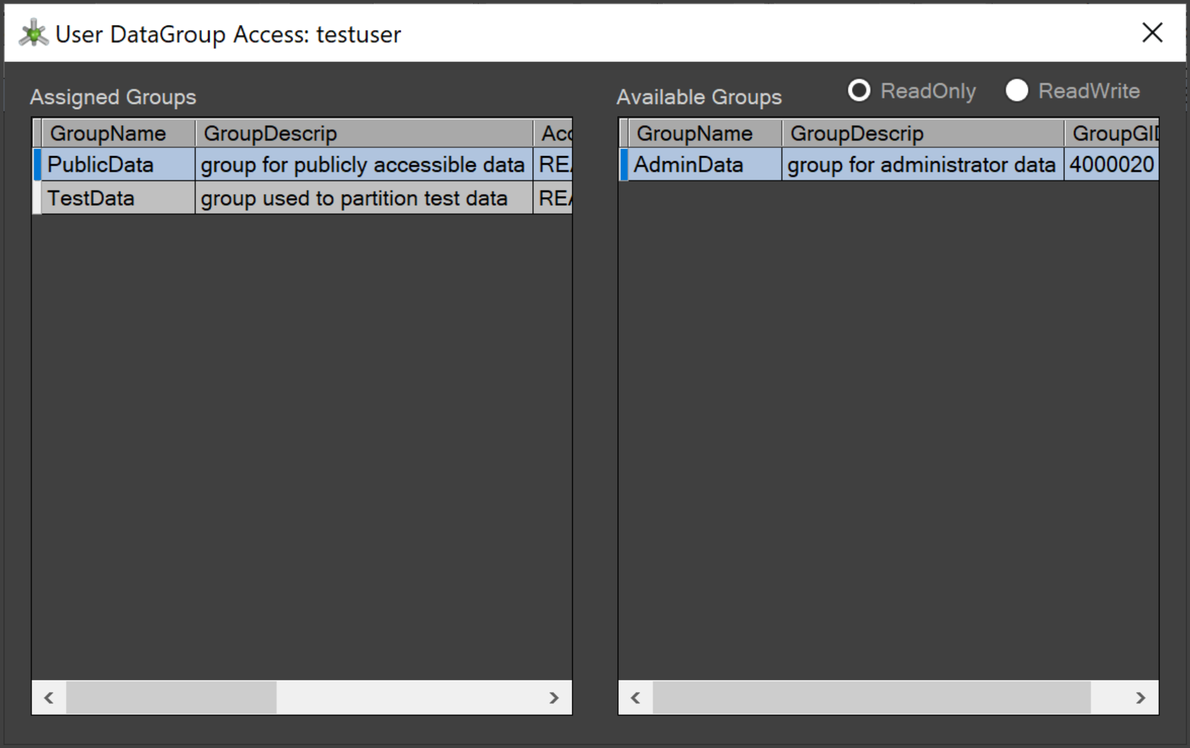
User Roles: testuser

Assigned Roles

RoleName	RoleDescrip
DefaultUser	Base API methods needed by all user a
LatticeAdmin	Admin role with access to all of the Lattic
Testing	Role that has access to testing API met

Available Roles

RoleName	RoleDescrip
AutoTestRole11597419339438	test role created by A
LatticeUser	Standard role with acc
MetricsAdmin	Admin role with acces
RemoteMetrics	Enables the collection
SecurityAdmin	Admin role with acces
WorkAdmin	Admin role with acces



User DataGroup Access: testuser

Assigned Groups

GroupName	GroupDescrip	Acc
PublicData	group for publicly accessible data	RE
TestData	group used to partition test data	RE

Available Groups ○ ReadOnly ● ReadWrite

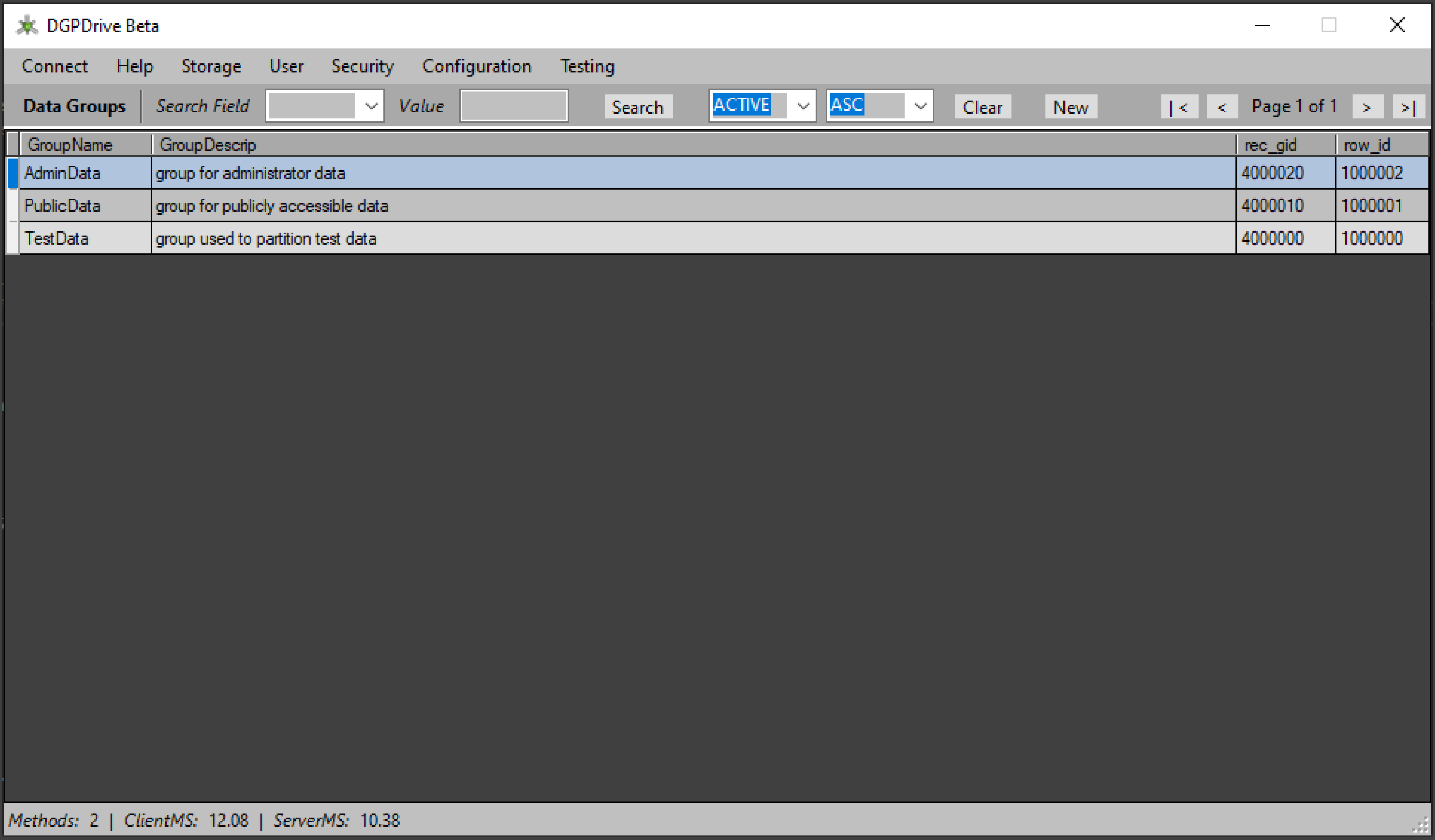
GroupName	GroupDescrip	GroupGI
AdminData	group for administrator data	4000020

DataGroups Form

DataGroups are used as a value embedded in the records of shared tables to horizontally partition the data by the DataGroup global ID value. The SQL syntax of the data access methods automatically include DataGroup membership and access levels of the user account when running all queries on those shared tables. Currently, only the Lattice Folders and Files tables use DataGroups for this type of data partitioning.

The net result is of those queries is that users are only able to see the folders that they are authorized to see, and the same is true for all of the files stored in the system as well. Users are not be able to tell that all other folders and files that they have not been authorized to access even exist.

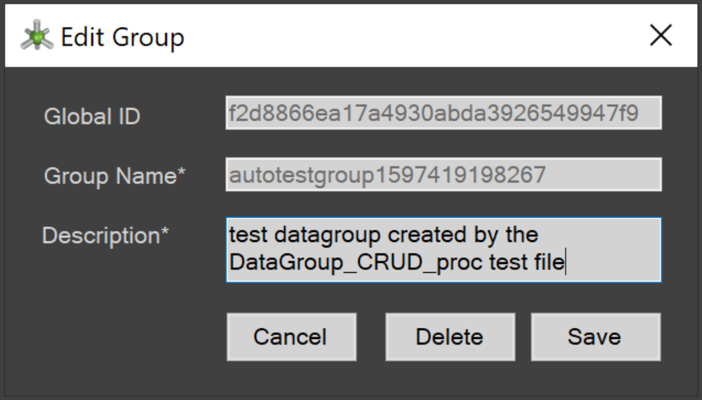
APIMethods Context Menu	Description
Edit…	Only allows the description field of the DataGroup record to be edited. This restriction is due to the "immutable, append-only" convention that strictly enforces backward compatibility. Editing any other DataGroup values would cause breaking changes to applications and systems that depend on the existing state to function correctly.
View History…	This form shows a grid of all versions of the selected record. If the record state is set to ACTIVE, the Recover Selected Record button is enabled to recover an edited version of the record as the current ACTIVE version. The process to recover a deleted record is different, so the Recover Selected Record button is disabled when the record state is set to DELETED.
Recover Deleted	This option is disabled unless the record state has been set to DELETED. Doing so allows the form to be used to search for deleted records. Selecting this option for a deleted record will recover the deleted record as the new current ACTIVE record.



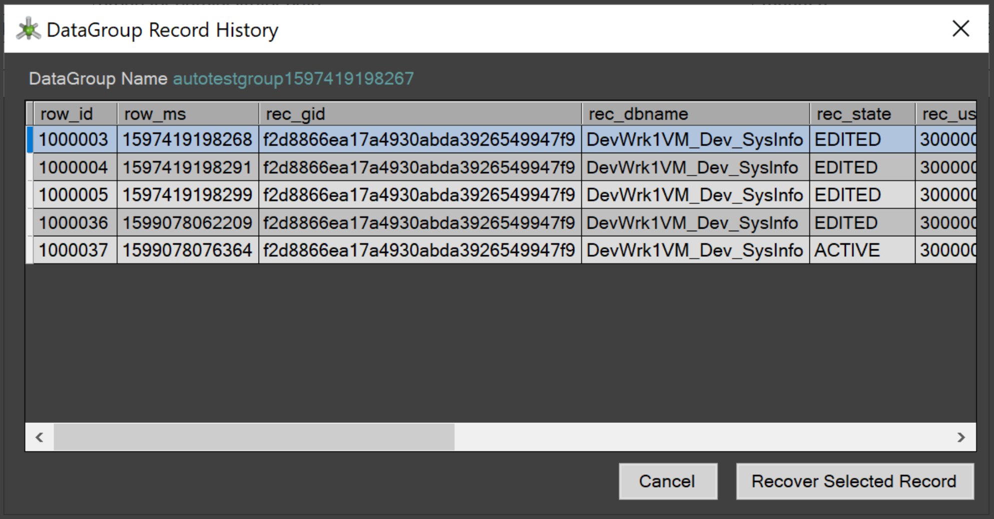
ACTIVE | ASC | Page 1 of 1

GroupName	GroupDescrip	rec_gid	row_id
AdminData	group for administrator data	4000020	1000002
PublicData	group for publicly accessible data	4000010	1000001
TestData	group used to partition test data	4000000	1000000

Methods: 2 | *ClientMS:* 12.08 | *ServerMS:* 10.38



Global ID	f2d8866ea17a4930abda3926549947f9
Group Name*	autotestgroup1597419198267
Description*	test datagroup created by the DataGroup_CRUD_proc test file



DataGroup Name autotestgroup1597419198267

row_id	row_ms	rec_gid	rec_dbname	rec_state	rec_us
1000003	1597419198268	f2d8866ea17a4930abda3926549947f9	DevWrk1VM_Dev_SysInfo	EDITED	300000
1000004	1597419198291	f2d8866ea17a4930abda3926549947f9	DevWrk1VM_Dev_SysInfo	EDITED	300000
1000005	1597419198299	f2d8866ea17a4930abda3926549947f9	DevWrk1VM_Dev_SysInfo	EDITED	300000
1000036	1599078062209	f2d8866ea17a4930abda3926549947f9	DevWrk1VM_Dev_SysInfo	EDITED	300000
1000037	1599078076364	f2d8866ea17a4930abda3926549947f9	DevWrk1VM_Dev_SysInfo	ACTIVE	300000