

Broker ACI Functions

Programs written for EntireX Broker contain instructions that specify to the Broker which functions to perform. The function's behavior is controlled by the option value and other ACI fields.

This chapter covers the following topics:

- Overview Table
- Function Descriptions
- Option Descriptions
- ACI Field/Function Reference Table

See also *Broker ACI Fields*.

Overview Table

Function Name	Applicable Participant				Logon Required ⁽¹⁾	User ID Required ⁽²⁾	Minimum API Version
	Client	Server	Publisher	Subscriber			
CONTROL_PUBLICATION			X	X	X	X	8
DEREGISTER		X				X	1
EOC	X	X				X	1
KERNELVERS	X	X	X	X		X	4
LOGOFF	X	X	X	X	X	X	2
LOGON	X	X	X	X		X	2
RECEIVE	X	X			⁽³⁾	X	1
RECEIVE_PUBLICATION				X	X	X	8
REGISTER		X				X	1
REPLY_ERROR	X	X				X	8
SEND	X	X			⁽³⁾	X	1
SEND_PUBLICATION			X		X	X	8
SETSSLPARMS	X	X	X	X			6
SUBSCRIBE				X	X	X	8
SYNCPOINT	X	X			X ⁽³⁾	X	3
UNDO	X	X			X	X	2
UNSUBSCRIBE				X	X	X	8
VERSION	X	X	X	X			2

Key

- (1) Logon is a prerequisite for issuing this command. See LOGON.
- (2) User ID is a prerequisite for issuing this command.
- (3) The following functions require a logon when used with units of work: RECEIVE, SEND, SYNCPOINT.

Function Descriptions

CONTROL_PUBLICATION

This function is used to control a publication.

Option	Description
BACKOUT	<ul style="list-style-type: none"> Used by the publisher, it backs out a publication in <i>received</i> status. Used by the subscriber, it backs out the subscriber's reading of the publication; <code>RECEIVE_PUBLICATION, PUBLICATION-ID=NEW</code> will read it again.
COMMIT	<ul style="list-style-type: none"> Used by the publisher, it completes the publication, places it in <i>accepted</i> status and makes it available to a subscriber. Used by the subscriber, it acknowledges receipt of a publication.
QUERY	Used by the publisher, it returns the status of the publication specified by the <code>PUBLICATION-ID</code> .
SETUSTATUS	Used by the publisher, it updates the optional application-specific user status of the specified publication that is in <i>received</i> status.

DEREGISTER

This function is used by a server application to deregister a service from EntireX Broker. Assigned resources are deallocated. To remove multiple services, specify either `SERVER-CLASS`, `SERVER-NAME` and/or `SERVICE`.

Option	Description
IMMED	To execute a immediate deregistration, use <code>IMMED</code> . The service is removed immediately; an error code informs partners in existing conversations of this removal. Any active UOW is <i>backedout</i> .
QUIESCE	To execute a non-immediate deregistration, use <code>QUIESCE</code> . All active conversations are allowed to continue until an EOC is issued or a conversation timeout occurs. The application that issues the <code>DEREGISTER</code> function must remain active until all existing conversations are ended. No new conversations are accepted.

EOC

This function is used by a client or server and applies to conversational mode only. It is used to terminate one or more conversations. EntireX Broker accepts no additional `SEND` s for the conversation(s). The partner can receive requests and messages that were sent before the EOC was issued.

Although conversations are normally terminated by the client, the EOC function can be issued by either partner in a conversation. If an active UOW has not yet been committed (that is, its current status is *received* or *delivered*) the conversation will not be terminated until the UOW is either *committed*, *backedout*, *cancelled*, or *timedout*. See *Broker UOW Status Transition*.

- To terminate all conversations initiated by the participant, use CONV-ID.
- To terminate all conversations for a particular service, use CONV-ID, SERVER-CLASS, SERVER-NAME and/or SERVICE.

Option	Description
CANCEL	To inform the partner that the EOC is due to an unexpected event, use CANCEL.

KERNELVERS

This function is used by any participant to determine the highest API-VERSION that is supported by the requested Broker. The highest API-VERSION that the Broker supports is returned in the API-VERSION field (see API-TYPE and API-VERSION for client and server | publish and subscribe). Platform and version information is returned in the error text.

Option	Description
CHECKSERVICE	If option is set to CHECKSERVICE, the command will determine whether a specified SERVICE is currently registered to the Broker.

The KERNELSECURITY field returns one of the following values to indicate whether the kernel is running with security. These values are returned only for API version 7 or above.

Value	Description
Y	Software AG-supplied security (SECURITY=YES in the Broker attribute file).
U	User-written security (SECURITY=YES in the Broker attribute file).
N	SECURITY=NO in the Broker attribute file.

LOGOFF

This function is used by all application components before termination when no further Broker functions are to be issued.

LOGOFF should be issued after the application's last SEND, SEND_PUBLICATION, RECEIVE, RECEIVE_PUBLICATION, DEREGISTER or UNSUBSCRIBE has been executed. It releases all resources used by the application immediately rather than waiting until they time out (see *Timeout Parameters* for client and server | publish and subscribe).

LOGON

This function is used by all application components so that the application can establish communication with a particular instance of the Broker kernel.

Allows the client or server application to logon to EntireX Broker, which allocates the necessary structures to handle the new participant. If EntireX Broker is running in a secure environment (with SECURITY=YES in the attribute file), LOGON performs the authentication process.

LOGON is normally the first function.

In addition to the USER-ID, the LOGON optionally transmits the PASSWORD, NEWPASSWORD and SECURITY-TOKEN to authenticate itself, provided SECURITY=YES is set in the broker attribute file.

RECEIVE

This function is used by clients to receive incoming messages and by servers to receive incoming requests.

- You can specify a WAIT time, causing the RECEIVE to wait for the request or message that satisfies the operation.
- The RECEIVE-LENGTH field is required. It specifies the maximum length of data the caller can receive. A receive buffer of at least this length must be provided. The actual length of the message received is returned in the RETURN-LENGTH field.

Option	Description	Note
ANY	Used with the RECEIVE function to indicate that the RECEIVE will be satisfied by any message, whether part of a UOW or not.	
LAST	To retrieve the last (most recent) message in a conversation, use LAST.	With this option, WAIT must be set to "NO" or not specified.
MSG	To indicate that the RECEIVE will be satisfied only by a message that is not part of a UOW, use MSG. See also <i>Broker UOW Status Transition</i> .	
NEXT	To retrieve the next unprocessed request or message in a conversation, use NEXT.	
PREVIEW	To retrieve the next unprocessed request or message in a conversation without deleting the previous message or moving the READ pointer, use PREVIEW, which excludes using units of work.	
SYNC	To receive only messages that are part of a UOW, use SYNC. See also <i>Broker UOW Status Transition</i> .	

RECEIVE_PUBLICATION

This function is used by subscribers to receive publications. If PUBLICATION-ID=NEW is specified, Broker will deliver the next publication in *accepted* status to the subscriber, if a publication is in *accepted* status. In the case of subsequent RECEIVE_PUBLICATION commands, a valid PUBLICATION-ID is used to receive any messages from a multi-message publication. See function CONTROL_PUBLICATION to acknowledge the publication.

ACI Field	Description
RECEIVE-LENGTH	This field is required.
WAIT	You can specify a WAIT time to wait for a new publication.

REGISTER

This function is used by servers to inform EntireX Broker that a service is available. The Broker obtains information about the service from the *Broker Attributes*, creates the appropriate environment, and makes the participant available as the specified `SERVER-CLASS`, `SERVER-NAME` and `SERVICE`.

If REGISTER is the first call by a server when both `AUTOLOGON` and `SECURITY` are set to "YES" in the Broker attribute file, `USER-ID` and `PASSWORD` are required in order to authenticate and authorize the server. This is because an implicit LOGON is being performed.

The services being registered must be defined in the attribute file.

Option	Description
ATTACH	To register an attach service, use ATTACH. An attach service cannot be requested by a client. Its function is to make available a service that cannot otherwise be scheduled.

REPLY_ERROR

This function is used by clients or servers to send an error message to the partner of the conversation. The error number is specified in the error code field. The sent message is delivered as an error text; the specified error number is delivered as an error code.

- The user must be logged on.
- The error number is a numeric 8-byte value and must start with 8 - e.g., 80010001. A zero error number will be rejected. These errors are user-definable and therefore not documented.
- The error message is provided in the send buffer and is limited to 40 bytes.
- Use the `SEND-LENGTH` field to specify the length of the error message.
- `REPLY_ERROR` can be used with a valid `CONV-ID` only.
- Only `WAIT=NO` is allowed.
- The conversation is not allowed to contain units of work.

Option	Description
EOC	To end the conversation after the <code>REPLY_ERROR</code> function, use EOC.

SEND

This function is used by clients to send requests and by servers to send replies (messages). If a corresponding `RECEIVE` function issued by a partner application is outstanding, EntireX Broker forwards the request or message to that partner application. If not, EntireX Broker queues the request or message until a suitable `RECEIVE` is issued by a partner application. If no suitable `RECEIVE` is issued by a partner application, the request will timeout within the specified timeout period.

- You can specify a SEND with either of the following:

- WAIT=YES | Value

This causes an implicit RECEIVE to be generated and the SEND to wait for a reply. If a reply is expected, the SEND must pass the length of the receive buffer, in bytes, as the value of the RECEIVE-LENGTH parameter. The actual - not the specified - length of the reply is returned to the sender as the RETURN-LENGTH value.

- WAIT=NO

Choose WAIT=NO if you are only forwarding a request or message.

- Use the SEND-LENGTH field to specify the length of the request or message being sent. The specified number of bytes is transferred, starting at the beginning of the send buffer.
- The client starts a new conversation, using CONV-ID=NEW.
- The client can specify non-conversational mode, using CONV-ID=NONE.
- Include the SERVER-CLASS, SERVER-NAME and SERVICE if this is a new conversation or a non-conversational request.
- If you add the ENVIRONMENT parameter, its value is passed to the translation routine for the service.
- To transmit conversation-related data to the sending application, use USER-DATA.

Option	Description	Note
COMMIT	Use COMMIT to indicate that the UOW being sent is complete and can now be delivered to the intended receiver, which can be either client or server.	
EOC	To end the conversation after the SEND, use EOC.	WAIT must be "NO" or not specified.
HOLD	To hold SEND data in a queue, use HOLD. The data is released by a SEND without a HOLD.	WAIT must be "NO" or not specified.
SYNC	Client and server can send a message as part of a unit of work (UOW), using SYNC.	WAIT must be "NO" or not specified.

SEND_PUBLICATION

This function is used by publishers to create publications for a specified topic. PUBLICATION-ID=NEW is used to create a new publication. In the case of a subsequent SEND_PUBLICATION, a valid PUBLICATION-ID is used when sending a multi-message publication. The publisher completes the publication either by specifying SEND_PUBLICATION, OPTION=COMMIT or using the function CONTROL_PUBLICATION. If SEND_PUBLICATION, OPTION=COMMIT is used, the option is specified on the final SEND_PUBLICATION in the sequence.

Option/ ACI Field	Description
OPTION=COMMIT	Required to complete a publication if the function CONTROL_PUBLICATION is not used.
PUBLICATION-ID=NEW	Required to start a new publication.
SEND-LENGTH	Required.

SETSSLPARMS

This function is used by clients and servers to set the SSL parameters. It can be called whenever the transport method SSL is available.

The SSL parameters are specified in the send buffer, (second parameter of the Broker ACI call). These SSL parameters are used and communication is performed if the Secure Sockets Layer is configured. See *Running Broker with SSL or TLS Transport* under z/OS | UNIX | Windows.

SUBSCRIBE

This function is used to request EntireX Broker to subscribe to a specific topic. The *Topic-specific Attributes* under *Broker Attributes* contains parameters that specify the behavior of the topic. If Broker runs with the setting SECURITY=YES in the Broker attribute file, the user must be successfully authenticated by EntireX Broker and also authorized to subscribe to the specific topic.

The TOPIC name must be defined in the Broker attribute file.

Option	Description
DURABLE	<p>If durable is specified within the SUBSCRIBE command, users need to subscribe only once to a topic. The subscription is retained after the user issues a logoff command or if the subscriber has timed out. Similarly, the subscription remains if Broker is restarted. All publications necessary to meet subscription requirements are also retained. If a subscriber no longer wishes to subscribe to a topic, it must issue an UNSUBSCRIBE command; otherwise the subscription remains valid until the subscription expiration time has passed.</p> <p>Durable subscription requires the administrator to configure the persistent store. See <i>Concepts of Persistent Messaging</i>. In addition, the topic must be specified as durable within the <i>Topic-specific Attributes</i> under <i>Broker Attributes</i>.</p>

ALLOW-DURABLE is a topic-specific attribute that determines whether a subscriber is allowed to perform a durable subscription to a topic. If users are allowed to durably subscribe to any topic, you must specify a value for the SUBSCRIBER-STORE attribute.

SYNCPOINT

This function allows you to manage units of work (UOWs), both persistent and non-persistent, that have been sent or received. See *Using Persistence and Units of Work*.

SYNCPOINT is used with the OPTION field. The UOWID parameter is required and normally limits the request to a specified UOW. For example:

```
SYNCPOINT OPTION=COMMIT,UOWID=n
SYNCPOINT OPTION=BACKOUT,UOWID=n
```

In cases where a server receives a UOW and sends a different UOW, you can ensure that the two UOWs will be processed together (that is, if one is committed, both are committed) by specifying UOWID=BOTH. For example:

```
SYNCPOINT OPTION=COMMIT,UOWID=BOTH
```

UOWID=BOTH can also be used with BACKOUT. This simply backs out both UOWs in a single call instead of two separate calls:

```
SYNCPOINT OPTION=BACKOUT,UOWID=BOTH
```

Option	Description
BACKOUT	Used by the sender, it causes the UOW to be deleted, with a status of <i>backedout</i> . By the receiver, causes the UOW to be returned to its prior, unreceived state, with a status of <i>accepted</i> . The ADCOUNT field is incremented. See also <i>Broker UOW Status Transition</i> .
CANCEL	Used by the receiver, it causes the UOW to be considered finished, with a status of <i>cancelled</i> . No further processing of the UOW is possible. The sender can cancel the UOW if, and only if, it is in <i>accepted</i> status. The following sequence of commands, issued during recovery processing, allows the sender to remove any created but undelivered UOWs: <ul style="list-style-type: none"> ● SYNCPOINT OPTION=LAST ● SYNCPOINT OPTION=CANCEL ● SYNCPOINT OPTION=DELETE
COMMIT	User by the sender, it indicates that the UOW has been completely built and can be made available for delivery, with a status of <i>accepted</i> . By the receiver, indicates that the UOW has been completely received, with a status of <i>processed</i> . No further processing of the UOW is possible.
EOC	With UOWID= <i>n</i> , commits the UOW being created and ends the conversation.
EOCCANCEL	With UOWID= <i>n</i> , commits the UOW being created and cancels the conversation, that is, terminates the conversation immediately.
DELETE	With UOWID= <i>n</i> , deletes the persistent status of the specified UOW. The UOW must be logically complete (<i>processed, cancelled, timedout, backedout, discarded</i>) and must have been created by the caller.
LAST	Returns the status of the last UOW sent by the caller. In addition, SERVER-CLASS, SERVER-NAME, SERVICE and CONV-ID are also returned.
QUERY	With UOWID= <i>n</i> , returns the status of the specified UOW. In addition, SERVER-CLASS, SERVER-NAME and SERVICE details of the associated server are also returned.
SETUSTATUS	With UOWID= <i>n</i> , updates the user status of the specified UOW.

UNDO

This function is used to remove messages that have been sent but not received. It can only be used with an existing conversation. When a message is undone, the conversation continues.

Note:

UNDO is not used in conjunction with units of work. See *Using Persistence and Units of Work*.

Option	Description
HOLD	To undo messages in HOLD status, use UNDO with HOLD.

UNSUBSCRIBE

This function is used to cancel the subscription to a specific topic. The fully specified topic name is required to execute the UNSUBSCRIBE request.

VERSION

This function is used to return the version of the stub implementation in the receive buffer. This version string is useful to the application in determining the maximum API version supported by the stub and to Software AG support if problems occur.

The string was modified in version 8.2. Example:

```
EntireX Broker Stub XXXXXXXX Version=08.3.0.00, Highest API Supported=09
```

where "XXXXXXXX" is the name of the stub, for example "CICSETB".

Option Descriptions

Number	Option	Description
1	MSG	Used with a RECEIVE function to receive only a message that is not part of a UOW.
2	HOLD	Used in conversational mode only. <ul style="list-style-type: none"> ● On a SEND function, places the messages in a HOLD queue. Messages are released by a SEND without HOLD. ● On an UNDO function, releases all previously held messages. ● On a LOGOFF function, the conversation is not ended although the user is logged off.
3	IMMED	Used with the Deregister function to immediately terminate all conversations for the specified server. All partners are informed with an appropriate error code.

Number	Option	Description
4	QUIESCE	Used with the Deregister function to terminate a server smoothly. Existing conversations are allowed to end normally; no new conversations are accepted. The server is removed from the "active" list. QUIESCE is the default option for the Deregister function.
5	EOC	Used with the SEND function to end the conversation with the current message. It can be issued by either partner. The conversation is not ended if an active UOW has not yet been committed, that is, its status is <i>received</i> or <i>delivered</i> . See <i>Broker UOW Status Transition</i> .
6	CANCEL	Used with the EOC function to abort a conversation rather than terminate normally. The receiver of a UOW can use SYNCPOINT OPTION=CANCEL to interrupt processing and discard the UOW.
7	LAST	<ul style="list-style-type: none"> ● Used in conversational mode with the RECEIVE function to retrieve the last (most recent) message. ● Used with the SYNCPOINT function, it returns the status of the last UOW sent by the caller. ● Used with the CONTROL_PUBLICATION function, it returns the status of the last publication sent by the publisher.
8	NEXT	Used with the RECEIVE function to retrieve the next unprocessed request or message. The request or message is then considered processed and can be accessed only with OPTION=LAST. NEXT is the default option for the RECEIVE function.
9	PREVIEW	Used with the RECEIVE function to retrieve the next unprocessed request message without deleting the previous message or moving the READ pointer. The previewed message will be retrieved again by the next RECEIVE OPTION=NEXT.
10	COMMIT	<ul style="list-style-type: none"> ● COMMIT is used with the SYNCPOINT function to commit the active UOW. For a UOW being sent, it means that the UOW is complete and can now be delivered to the intended receiver. For a UOW being received, it means that that UOW is complete and no further processing of the UOW is allowed. ● COMMIT is used with the SEND function to commit the active UOW. ● COMMIT is used with the CONTROL_PUBLICATION function as follows: <ul style="list-style-type: none"> ● Used by the publisher, it completes the publication and makes it available to subscribers. ● Used by the subscriber, it acknowledges receipt of a read publication. ● COMMIT is used with the SEND_PUBLICATION function to commit the publication and make it available to subscribers.

Number	Option	Description
11	BACKOUT	<ul style="list-style-type: none"> ● The receiver of a UOW can use SYNCPOINT OPTION=BACKOUT to return the UOW to its undelivered state. The UOW can be processed again, in its entirety, by subsequent RECEIVE operations. The sender of a UOW can use SYNCPOINT OPTION=BACKOUT to delete the UOW. No further processing of the UOW is allowed. ● BACKOUT is used with the CONTROL_PUBLICATION function as follows: <ul style="list-style-type: none"> ○ Used by the publisher, it back outs a publication. ○ Used by the subscriber, it back outs the subscriber's reading of the publication; RECEIVE_PUBLICATION will read it again.
12	SYNC	On a SEND function, indicates that the message is part of a UOW. On a RECEIVE function, indicates that the RECEIVE will be satisfied only by a message that is part of a UOW.
13	ATTACH	Used with the REGISTER function to register an attach server.
14	DELETE	Used with the SYNCPOINT function to delete the persistent status information for the specified UOW.
15	EOCCANCEL	Used with the SYNCPOINT function to cancel the conversation after committing a UOW.
16	QUERY	<ul style="list-style-type: none"> ● Used with the SYNCPOINT function only to query the status of a UOW. ● Used with the CONTROL_PUBLICATION function, it returns the status of the publication specified by the PUBLICATION-ID.
17	SETUSTATUS	<ul style="list-style-type: none"> ● Used with the SYNCPOINT function to cancel the conversation after committing a UOW. ● Used with the CONTROL_PUBLICATION function, it is used by the publisher to update the optional application-specific user status of the specified publication.
18	ANY	Used with the RECEIVE function to indicate that the RECEIVE will be satisfied by a message that is or is not part of a UOW.
19		No longer used.
20	DURABLE	Used with the SUBSCRIBE function to establish a durable subscription to a topic with attribute ALLOW-DURABLE=YES.
21	CHECKSERVICE	Use with the KERNELVERS function to check if the specified service is active in EntireX Broker.

ACI Field/Function Reference Table

The following table identifies the ACI fields that apply to each of the Broker functions. For a given function, an ACI field value may be a request field (Rq), and/or a reply field (Rt). Optional fields are marked (O).

ACI Field	Function																	
	SEND	RECEIVE	UNDO	EOC	REGISTER	DEREGISTER	VERSION	LOGON	LOGOFF	SYNCPPOINT	KERNELVERS	SETSSLPARMS	SEND_PUBLICATION	RECEIVE_PUBLICATION	SUBSCRIBE	UNSUBSCRIBE	CONTROL_PUBLICATION	REPLY_ERROR
ADDCOUNT		Rt								Rt			Rt	Rt			Rt	
API-TYPE	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq
API-VERSION	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq Rt	Rq	Rq	Rq	Rq	Rq	Rq	Rq
BROKER-ID	Rq	Rq	Rq	Rq	Rq	Rq		Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq
CLIENT-UID		Rt												Rt				
COMMITTIME	Rt	Rt								Rt			Rt	Rt			Rt	
COMPRESSLEVEL	O Rt	O Rt											O Rt	O Rt				O
CONV-ID	Rq	Rq	Rq	Rq						Rq Rt								Rq
CONV-STAT		Rt												Rt				
DATA-ARCH	O	O											O	O				O
ENCRYPTION-LEVEL	O	O			O					O								
ENVIRONMENT	O	O											O	O				O
ERROR-CODE	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt	Rt
ERRTEXT-LENGTH	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
FORCE-LOGON	O	O	O	O	O	O		O	O	O			O	O	O	O	O	O
FUNCTION	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq	Rq
KERNELSECURITY											Rt							
LOG-COMMAND	Rq	Rq	Rq	Rq	Rq	Rq		Rq	Rq	Rq	Rq		Rq	Rq	Rq	Rq	Rq	Rq
LOCALE-STRING	O	O											O	O				O
MSG-ID																		
NEWPASSWORD	O				O			O										
PARTNER-BROKER-ID (deprecated)	O	Rt	Rt	Rt	Rt	Rt		Rt	Rt	Rt	Rt		Rt	Rt	Rt	Rt	Rt	Rt
OPTION	O	O	O	O	O	O			O	O	O		O	O	O		Rq	O

ACI Field	Function																		
	SEND	RECEIVE	UNDO	EOC	REGISTER	DEREGISTER	VERSION	LOGON	LOGOFF	SYNCPPOINT	KERNELVERS	SETSSLPARMS	SEND_PUBLICATION	RECEIVE_PUBLICATION	SUBSCRIBE	UNSUBSCRIBE	CONTROL_PUBLICATION	REPLY_ERROR	
PASSWORD	O				O			O											O
PUBLICATION-ID													Rq Rt	Rq			Rq Rt		
RECEIVE-LENGTH	O	Rq					Rq							Rq					
RETURN-LENGTH	Rt	Rt					Rt							Rt					
SECURITY-TOKEN	O	O	O	O	O	O	O	O	O	O			O	O	O	O	O	O	O
SEND-LENGTH	Rq											Rq	Rq						Rq
SERVER-CLASS	Rq	O		O	Rq	O				O Rt									
SERVER-NAME	Rq	O		O	Rq	O				O Rt									
SERVICE	Rq	O		O	Rq	O				O Rt									
STORE	O	Rt											Rt	Rt					
TOKEN	O	O	O	O	O	O		O	O				Rq	Rq	Rq	Rq	Rq	Rq	O
TOPIC													Rq	Rq	Rq	Rq	Rq		
UOWID	O Rt	O Rt								Rq									
UOWSTATUS	Rt	Rt		Rt						Rt			Rt	Rt					
UOW-STATUS-PERSIST										O									
USER-DATA	O	Rt								Rt									O
USER-ID	Rq	Rq	Rq	Rq	Rq	Rq		Rq	Rq	Rq			Rq	Rq	Rq	Rq	Rq	Rq	Rq
USTATUS	O Rt	O Rt		O Rt						O Rt			O Rt	O Rt				O Rt	
UWSTAT-LIFETIME										O									
UWTIME	O																		
WAIT	O	O												O					O