

# ATM Client Runtime Controls

- Usage of Client Runtime Controls
  - Descriptions of Client Runtime Controls
- 

## Usage of Client Runtime Controls

Correct operation of the ATM transaction manager and the Adabas Transaction Manager client proxy depends on the settings of client runtime controls. These are defined and maintained using Online Services. Their settings determine the logic to be used by the proxy during transaction processing. These settings should be determined according to production system requirements, and should be tested before being used in production.

At execution time, the Adabas Transaction Manager client proxy checks the Adabas System Coordinator system file for client runtime controls that match the name of the currently executing job. If no exact match is found, a wild-card match is attempted. Failing this, the control values for job name DEFAULT will be returned. If there is no definition under the name DEFAULT, hard-coded default values in the Adabas Transaction Manager kernel will be used.

It is therefore important to set appropriate values for each job that could load and execute the Adabas Transaction Manager client proxy.

### Note:

The `NumberOfLogRecordEntries` control setting requires particular attention. This setting determines the amount of memory, per client session, that is allocated for logging diagnostic information. In a TP system that services many users, a setting of 100, for example, could quickly consume a large amount of memory. In such systems, choose a small value for this setting. See the section Client Sessions for more information.

## Descriptions of Client Runtime Controls

This section describes each ATM runtime control:

- ATM ON/OFF: Activate ATM Processing
- System Coordinator Group Name
- Maximum Number of Databases
- Number of Log Record Entries
- Transaction Control
- Emergency Serial ET Commands
- Allow Serial Remote RM Coordination

- Generate OP Commands
- Syncpoint Processing Mode
- Transaction Model
- Generate External Syncpoint on BT Command
- Generate External Syncpoint on CL Command
- Generate External Syncpoint on ET Command
- Use Client-Side Transaction Manager
- Use Host System Transaction Manager
- Extended Hold

### ATM ON/OFF: Activate ATM Processing

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not ATM is to provide transaction coordination for this client session.	ON   OFF	OFF

Possible values:

Value	Description
ON	Users will perform distributed transaction processing, and ATM is to provide transaction coordination.
OFF	ATM is not to be used for coordinating transactions for users in this client session.

### System Coordinator Group Name

Parameter Type	Use	Possible Values	Default
Client runtime	Identifies the Adabas System Coordinator group in which the client session or TP system will execute.	Any valid Adabas System Coordinator group name	A valid group name must be provided

### Maximum Number of Databases

Parameter Type	Use	Possible Values	Default
Client runtime	Sets the maximum number of databases with which a client session can have an active session at one time.	number	4

The setting should include all databases without regard to their DTP parameter settings.

## Number of Log Record Entries

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not internal logging is to be performed, and if so, how much space, per client session, will be used for this purpose.	0-4096	0

The ATM client proxy is capable of logging internal events for diagnostic purposes. The log buffer area is reused cyclically.

Possible values:

Value	Description
0	The client proxy will not acquire log buffer areas and will not log internal events.
1-4096	The number of entries to be reserved for internal event logging. Each entry requires 96 bytes.

### Note:

Do not specify a high value for a TP system that services a large number of users since this may result in memory becoming exhausted.

The `NumberLogRecordEntries` client control has no effect on the logging that takes place in the transaction manager's address space. Logging by the transaction manager is controlled by the `ADARUN` parameter `TMLOG`.

## Transaction Control

Parameter Type	Use	Possible Values	Default
Client runtime	Determines the action to be taken whenever an unsolicited syncpoint request is received.	GLOBAL   LOCAL	LOCAL

The setting of this control determines what action is to be taken whenever an unsolicited syncpoint request is received, that is, whether a decision to commit or back out a global transaction is to be made by the application program itself using its local ATM client proxy, or by a software component other than ATM

or ATM client proxy.

Unsolicited syncpoint requests can occur if:

- The option `Client-sideTransactionManager` is in effect and ATM could receive a request to commit or back out a global transaction other than through an ET or BT command; or
- the ATM transaction manager executes with the parameter setting `TMSYNCMGR=RRMS`; or
- an application could cause a participating trigger to execute, causing an ET command to be issued, all without the knowledge of the user's local ATM client proxy.

Possible values are:

Value	Description
LOCAL:	<p>Decision to commit or back out a global transaction is to be made by the application program itself using its local ATM client proxy.</p> <p>If unsolicited syncpoints are not valid in your environment, specify LOCAL so that any unsolicited syncpoint request is reported as an error. Specifying <code>TRNCTL=LOCAL</code> does not prevent unsolicited syncpoints from being processed; rather, it results in the user's next Adabas command being rejected with a response code 243 indicating that the syncpoint has occurred.</p>
GLOBAL	<p>Decision to commit or back out a global transaction can be made by a software component other than ATM or ATM client proxy. GLOBAL is required if unsolicited syncpoint requests are acceptable.</p> <p><b>Note:</b> Running with GLOBAL in any environment other than CICS/RMI can cause the ATM client proxy to make additional calls to the ATM transaction manager to check transaction status. It is recommended to set <code>Transaction Control</code> to LOCAL wherever possible to eliminate these extra calls.</p>

## CICS/RMI Considerations

When running with the `Client-side Transaction Manager` option set to YES in a CICS/RMI environment, a decision to commit or back out a global transaction can occur in several ways:

- the application program issues ET or BT;
- the application program issues `EXEC CICS SYNCPOINT` (or `SYNCPOINT ROLLBACK`);
- the CICS task terminates, normally or abnormally, when updates are pending.

In the first case, the ATM client proxy is aware of the syncpoint decision.

In all other cases, from the client proxy's point of view, the syncpoint is unsolicited. If the client session is at global transaction status, an unsolicited syncpoint has no effect on Adabas databases, other than to cause held ISNs to be released. On the other hand, if an unsolicited syncpoint occurs when Adabas changes are pending, ATM is instructed to commit the changes; in this case also, held ISNs are released.

If such unsolicited syncpoints are acceptable in your CICS system, specify GLOBAL for the TransactionControl client control - otherwise, specify LOCAL to treat unsolicited syncpoint requests as errors.

Most CICS applications run in pseudo-conversational mode; that is, the current CICS task terminates when a screen I/Os occurs. If ATM CICS/RMI is installed and active, and client runtime control TransactionModel is set to MESSAGE, an application that runs in pseudo-conversational mode cannot keep an Adabas transaction open across screen I/Os because an implied (unsolicited) commit syncpoint occurs at every screen I/O. In this mode, it is not possible for a transaction to remain open across screen I/Os. See the TransactionModel control for details of a different way of responding to unsolicited syncpoints.

If the client control TransactionControl is set to LOCAL and the syncpoint resulted in Adabas changes being committed, the next Adabas call following a screen I/O returns response code 243.

Various Natural features including some SYSSEC functions maintain open Adabas transactions across screen I/Os. In a CICS/RMI environment operating with the client runtime control TransactionModel set to MESSAGE, such functions are likely to

- fail with a NAT3243 error when TransactionControl is set to LOCAL; or
- be unable to back out in case of errors when TransactionControl is set to GLOBAL; or
- fail with response code 144.

You should therefore execute Natural system functions in one of the following ways:

- execute Natural system functions in a system other than CICS, or in a CICS system using a link module in which the RMI is not active; or
- execute Natural system functions in a client session that operates with the client runtime control TransactionModel set to DYNAMIC; or
- execute Natural system functions in conversational mode when using the CICS Resource Manager Interface.

## RRMS Considerations

When ATM's RRMS interface is active, a decision to commit or back out a global transaction can occur in several ways:

- the application program issues ET or BT;
- the application program issues SRRCMIT or SRRBACK;
- the application or some other agent issues ATRCMIT or ATRBACK.

In the first case, the ATM's client proxy is aware of the syncpoint decision. In other cases, from ATM's point of view, the syncpoint is unsolicited. If the client session is at global transaction status, an unsolicited syncpoint has no effect on Adabas databases, other than to cause held ISNs to be released. On the other hand, if an unsolicited syncpoint occurs when Adabas changes are pending, ATM is instructed to commit the changes; in this case, also, held ISNs will be released.

If unsolicited syncpoints are acceptable in your system, set the client control `TransactionControl` to `GLOBAL` - otherwise, set `TransactionControl` to `LOCAL` to treat unsolicited syncpoint requests as errors. If `TransactionControl` is set to `LOCAL` and an unsolicited syncpoint causes a client sessions's Adabas changes to be committed, the next Adabas call issued by the user returns response code 243.

If ATM runs with the `HostSystemTransactionManager` option is specified for IMS/TM systems whose transactions are coordinated by RRMS, `TransactionControl` must be set to `GLOBAL` for these IMS systems.

## Emergency Serial ET Commands

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not the ATM client proxy is to switch a client session from DTP to serial ET/BT mode if the local ATM transaction manager becomes unavailable.	YES   NO   FORCE	FORCE

Possible values:

Value	Description
YES	The ATM client proxy is to switch a client session from DTP to serial ET/BT mode if the local transaction manager becomes unavailable. A response code will be returned to indicate that the local TM is unavailable.
NO	The ATM client proxy is not to switch a client session from DTP to serial ET/BT mode if the local TM becomes unavailable.
FORCE	A client session is to be switched to serial mode if necessary. No notification is provided.

For more information regarding DTP and Serial mode processing, refer to the section `Processing Modes`.

### Note:

If ATM automatically switches a client session to serial ET/BT mode when the session has a transaction open, the client might receive unexpected response codes, even if the runtime control value for serial processing is set to `FORCE`. For example, the transaction might be backed out, and response 9 returned to the client. Other response codes, such as response 240 subcode 88, might be given, depending on how far the transaction had progressed, and there is a possibility that the transaction will remain unresolved until the ATM transaction manager is once again able to carry out its responsibilities. Even so, the setting of value `FORCE` will ensure minimum disruption when a switch to serial ET/BT mode is necessary, and will probably allow most clients to continue processing without noticing the switch.

## Allow Serial Remote RM Coordination

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether the client's local Transaction Manager should use serial ET commands to remote RMs which are not signed on for Distributed Transaction Processing in the client's System Coordinator group.	YES   NO	YES

Possible values:

Value	Description
YES	If the client session changes one or more remote RMs which are currently not signed on for Distributed Transaction Processing, to any Transaction Manager within the scope of the client's System Coordinator Group, the client's local Transaction Manager will treat these RMs as if they were running with DTP=NO. That is, such a transaction will be committed in the normal way, using the two-phase commit protocol with respect to any signed-on RMs, and then ATM will issue ET commands to any changed remote RMs that are not signed on for DTP.
NO	If the client session attempts a transaction that changes an RM which is currently not signed on for DTP to any Transaction Manager within the scope of the client's COR Group, the transaction will usually be rejected with a suitable response code. In some circumstances the transaction may be allowed to complete normally, but only if it can be completed safely without the need of a two-phase commit operation.

**Note:**

If YES is specified, and a transaction changes one or more remote RMs which are not signed on for DTP, and a system or component failure occurs during the commit process, ATM cannot guarantee integrity of the transaction.

**Note:**

If the `EmergencySerialETCommands` control is set to YES or FORCE, and a client session is switched into serial ET/BT mode, serial remote RM coordination will take effect as long as the session is in serial ET/BT mode, regardless of the setting of the `AllowSerialRemoteRMCoordination` control.

## Generate OP Commands

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not the ATM client proxy is to generate an OP command when a session uses a database without having issued an OP.	NO   YES	NO

Possible values:

Value	Description
NO	The ATM client proxy will not generate OP commands on behalf of clients.
YES	Whenever a client session uses a new database for which no OP command has been issued, the ATM client proxy will issue an OP command on behalf of the client.

## Syncpoint Processing Mode

This parameter has been replaced by the `TransactionModel` runtime control.

## Transaction Model

Parameter Type	Use	Possible Values	Default
Client runtime	Transaction model	MESSAGE   DYNAMIC	MESSAGE

Possible values:

Value	Description
MESSAGE	The message-based transaction model will be used. According to this model, a syncpoint always takes place when processing of a message is complete (normally, this means that screen I/O causes a syncpoint). Syncpoints caused by ET, BT, OP and CL commands are handled in the normal way. Unsolicited syncpoints cause ATM to commit (or back out) changes made to Adabas databases, and release held ISNs.
DYNAMIC	The dynamic transaction model will be used. According to this model, transactions are delimited by Adabas commands. Unsolicited commit syncpoints are ignored by ATM; ET and CL commands cause ATM to commit pending Adabas changes. Unsolicited rollback syncpoints cause ATM to back out changes from Adabas databases and release held ISNs.

### Note:

Setting `TransactionModel` to MESSAGE is equivalent to `SYNCPMODE=FULL` with ATM Version 1.2.



**Note:**

Setting `TransactionModel` to `DYNAMIC` is equivalent to `SYNCMODE=ADABAS` with ATM Version 1.2.

**Note:**

There is no `TransactionModel` setting equivalent to `SYNCMODE=ALL` with ATM Version 1.2. The reason is that there is no distinction between the settings `SYNCMODE=ALL` and `SYNCMODE=FULL` with more recent versions of ATM, since ATM will always cause held ISNs to be released when a syncpoint takes place, unless directed to do otherwise by command options and the `ExtendedHold` client control.

For more information, see also the section `Syncpoint Processing Options`.

**Generate External Syncpoint on BT Command**

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not a BT command should cause ATM to request a rollback syncpoint from the external transaction coordinator.	YES   NO	YES

Possible values:

Value	Description
YES	Any BT command will cause ATM to request a rollback syncpoint from the external transaction coordinator that controls the current transaction. If there is no external transaction coordinator in use, this setting is ignored.
NO	ATM will not request a rollback syncpoint from any external coordinator when it processes a BT command. Furthermore, if ATM itself decides that a backout must be performed, it will not request a rollback from any external coordinator.

Regardless of the setting for this control, a BT command causes all Adabas changes to be backed out.

**Generate External Syncpoint on CL Command**

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not a CL command should cause ATM to request a commit syncpoint from the external transaction coordinator.	YES   NO	YES

Possible values:

Value	Description
YES	Any CL command will cause ATM to request a commit syncpoint from the external transaction coordinator that controls the current transaction. If there is no external transaction coordinator in use, this setting is ignored.
NO	ATM will not request a commit syncpoint from any external coordinator when it processes a CL command.

For IMS/TM systems for which the `HostSystemTransactionManager` option is specified, a setting of YES for the client control `GenerateExternalSyncpointOnCL` will be ignored, and the value NO will take effect.

## Generate External Syncpoint on ET Command

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not an ET command should cause ATM to request a commit syncpoint from the external transaction coordinator.	YES   NO	YES

Possible values:

Value	Description
YES	Any ET command will cause ATM to request a commit syncpoint from the external transaction coordinator that controls the current transaction. If there is no external transaction coordinator in use, this setting is ignored.
NO	ATM will not request a commit syncpoint from any external coordinator when it processes an ET command.

Regardless of the setting for this control, ET command causes any pending Adabas changes to be committed.

For IMS/TM systems for which the `HostSystemTransactionManager` option is specified, a setting of YES for the client control `GenerateExternalSyncpointOnET` will be ignored, and the value NO will take effect.

## Use Client-Side Transaction Manager

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not transactions in this client environment are to be controlled by the environment's native syncpoint manager.	YES   NO	NO

Possible values:

Value	Description
YES	Transactions in this client environment are to be controlled by the environment's native syncpoint manager.
NO	Transactions in this client environment are not to be controlled by the environment's native syncpoint manager.

**Note:**

CICS under z/OS or VSE/ESA is the only client environment for which a setting of YES is currently supported for this control.

**Note:**

The `Client-sideTransactionManager` and `HostSystemTransactionManager` options are mutually exclusive. However, setting `Client-sideTransactionManager` to YES does not preclude the use of RRMS as an external transaction coordinator for client sessions in other jobs or TP systems, since RRMS interfaces directly with the ATM client sessions in other jobs or TP systems and not with the client proxy

### CICS/RMI Considerations

When defining client runtime controls for a CICS/RMI environment:

- Setting `Client-sideTransactionManager` to NO means that ATM coordinates transactions in the normal way, but not under the control of the CICS Syncpoint Manager. That is, even when the RMI is enabled, no transaction coordination occurs between Adabas and other RMI-enabled resource managers.
- Setting `Client-sideTransactionManager` to YES activates the RMI so that the CICS Syncpoint Manager coordinates transactions across all resource managers.

### Use Host System Transaction Manager

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether or not transactions in this client environment are to be controlled by the host system's native transaction manager.	YES   NO	NO

Possible values:

Value	Description
YES	Transactions in this client environment are to be controlled by the host system's native transaction manager.
NO	Transactions in this client environment are not to be controlled by the host system's native transaction manager.

**Caution:**

If this control is set to YES for an inappropriate type of job or client environment, or for a job or environment which has not been correctly configured to use the host system's transaction manager, sessions executing in that job will experience response code 240, subcode 444, and will not function correctly.

**Note:**

The setting YES is currently supported only for single-user, single-TCB batch jobs or TSO sessions, and for Com-plete and IMS/TM systems, running under z/OS.

**Note:**

The `HostSystemTransactionManager` and `Client-sideTransactionManager` options are mutually exclusive.

**Extended Hold**

Parameter Type	Use	Possible Values	Default
Client runtime	Indicates whether P and M options on ET and BT commands will be honored when a distributed transaction is terminated by a series of ET or BT commands.	YES   NO	NO

Possible values:

Value	Description
YES	P and M options on ET and BT commands will be honored. For any application environment in which prefetch or multifetch command options can be used, the setting YES should be used.
NO	P and M options on ET and BT commands will not be honored.

If Adabas Vista is present in an application job or TP environment, the value YES will take effect, regardless of the setting specified for the `ExtendedHold` client control.

For more information on extended hold processing, see `Extended Hold Processing` in section `Termination Commands: ET and BT`.