Using Alert and Timeout Settings

In a cluster environment, the Adabas nuclei working on the same cluster database must collaborate to keep the database physically and logically consistent while processing user commands. To do this, they issue intracluster commands to one another. If one cluster member does not respond to an intracluster command from another cluster member within a specified time period, the sending member cancels the unresponsive member.

Adabas Cluster Services provides alert and timeout settings that are designed to help you prevent or handle critical situations where system problems might cause the prolonged unavailability of one cluster member, thus endangering the ability of the cluster member (or in severe cases, the entire cluster) to provide service. These settings include:

- A cancel alert that generates an operator message can be invoked when a cluster member is unresponsive to an intracluster command for a specified period of time. If the cluster member does not respond before the message times out, it is canceled.

- Self-termination alerts that generate operator messages can be invoked when a canceled cluster member does not terminate as requested (in a specified period of time) and the other cluster members prepare to self-terminate.

- An operator query that prints an operator console message and requests a response from the operator. This can be invoked by cluster members that are preparing to self-terminate because a canceled peer member remains active.

- XCF system- and member-level status monitoring can be used to determine if a cluster member is unable to respond to internal intracluster requests. This monitoring process prints operator messages that provide early warning information about the cluster member.

- Messaging statistics provide information about the performance of message transmission events. These statistics can be used to determine the impact of messaging on system performance and to determine how to set the ADARUN parameters related to the alert and timeout settings.

This chapter covers the following topics:

- Using Cancel Alerts

- Using Self-Termination Alerts

- Using the Self-Termination Operator Query

- Using XCF Status Monitoring

- Using Messaging Performance Statistics

Using Cancel Alerts

In cluster environments, the cluster members issue intracluster commands to one another to ensure that the database is kept physically and logically consistent. If one cluster member does not respond to an intracluster command from another cluster member within the time specified by the ADARUN MXMSG parameter, the sending member cancels the unresponsive member.

You can invoke a cancel alert before the unresponsive peer member is canceled. This alert generates an operator message that provides early warning information before the unresponsive cluster member is canceled.

The cancel alert is governed by the setting of the ADARUN MXMSGWARN parameter. This optional parameter specifies the number of seconds after which a cluster nucleus should generate an operator message warning about an outstanding intracluster response. If the cluster member does not respond within the time specified by ADARUN MXMSGWARN, message ADAX9C is issued. This warning message can be used to notify you sometime before the unresponsive cluster member is canceled.

As complements to the ADARUN MXMSG and MXMSGWARN parameters, two operator commands, `MXMSG` and `MXMSGWARN`, are provided that allow you to change the corresponding ADARUN settings dynamically, while the database is running. In addition, the `DPARM` operator command's output information includes the settings of the ADARUN parameters related to cluster alert and timeout enhancements.

For more information about the MXMSG and MXMSGWARN parameters, read *MXMSG: Timeout Threshold for Internucleus Command Processing* and *MXMSGWARN : Timeout Threshold for Internucleus Command Processing Warning* . For information about the `DPARM`, `MXMSG`, and `MXMSGWARN` operator commands, read *Adabas Cluster Nucleus Operator Commands* .

Using Self-Termination Alerts

In cluster environments, if one cluster nucleus has issued a cancellation request for a second unresponsive cluster nucleus, but the canceled peer cluster nucleus does not terminate within the time specified by the ADARUN MXCANCEL parameter, the sending nucleus will either return response code 124, subcode 28 (if the intracluster communication occurred on behalf of an Adabas command) or terminate itself abnormally (if the intracluster communication occurred on behalf of an internal process that must not fail).

You can invoke a self-termination alert before a nucleus terminates itself because a canceled peer nucleus fails to terminate. This alert generates an operator message that provides early warning information regarding the pending self-termination.

Self-termination alerts are governed by the setting of the new ADARUN MXCANCELWARN parameter. This optional parameter specifies the number of seconds after which a requesting cluster nucleus should generate an operator message warning about the inability of a canceled peer nucleus to terminate quickly. If the canceled peer nucleus does not terminate within the time specified by ADARUN MXCANCELWARN, message ADAX9G is issued. This warning message can be used to notify you that the nucleus issuing the message is in danger of terminating itself.

As complements to the ADARUN MXCANCEL and MXCANCELWARN parameters, two operator commands, `MXCANCEL` and `MXCANCELWARN`, are provided that allow you to change the corresponding ADARUN settings dynamically, while the database is running. In addition, the DPARM operator command's output information has been enhanced to include the settings of ADARUN parameters related to Adabas Cluster Services alert and timeout enhancements.

For more information about the MXCANCEL and MXCANCELWARN parameters, read *MXCANCEL: Timeout Threshold for Canceled Peer Nucleus* and *MXCANCELWARN : Timeout Threshold for Canceled Peer Nucleus Warning* . For information about the `DPARM`, `MXCANCEL`, and `MXCANCELWARN` operator commands, read *Adabas Cluster Nucleus Operator Commands* .

Using the Self-Termination Operator Query

You can invoke an operator query when a cluster member is in the process of self-terminating because a canceled peer nucleus fails to terminate. This gives you a chance to terminate the canceled cluster member manually, thus avoid the self-termination of the member that issued the ineffective cancel request.

This operator query prints a console message (message ADAX9J) explaining the situation and requesting instructions, waiting for a specified time for a response. The valid responses to message ADAX9J are:

- R (print the ADAX9J message again and continue to wait for resolution of this issue, but without setting a new wait period for the response)

- T (terminate the querying nucleus with message ADAX99 and user abend 79)

- W (continue to wait for another time period of length MXWTOR)

The amount of time the operator query waits for a response is governed by the setting of the ADARUN MXWTOR parameter. This optional parameter specifies the number of seconds the nucleus should wait for the operator response. If the operator does not respond in this timeframe and if the canceled peer nucleus still has not terminated, the requesting nucleus issues message ADAX99 and terminates itself.

However, if the canceled cluster member terminates after all (whether due to operator intervention or another reason), the cluster nucleus that issued the operator query stays alive; it retracts the query and initiates an online recovery process.

As a complement to the ADARUN MXWTOR parameter, an operator command, `MXWTOR`, is provided that allows you to change the MXWTOR setting dynamically, while the database is running. In addition, the `DPARM` operator command's output information includes the settings of ADARUN parameters related to Adabas Cluster Services alert and timeout enhancements.

For more information about the MXWTOR parameter, read *MXWTOR : Self-Termination Operator Query Interval* . For information about the `DPARM` and `MXWTOR` operator commands, read *Adabas Cluster Nucleus Operator Commands* .

Using XCF Status Monitoring

XCF system- and member-level status monitoring on z/OS systems can be used to determine early if a cluster member may be unable to respond to internal intracluster requests. This monitoring process occurs by checking the activity (heartbeat) of each cluster nucleus and printing operator messages which provide early warning information about the cluster nuclei that show no heartbeat.

XCF status monitoring provides a second method by which Adabas Cluster Services can warn you that a cluster nucleus might be unable to respond in a timely way to intracluster commands. The first method is, of course, via the normal intracluster communication that occurs between cluster members. If a nucleus has heartbeat exceptions (as determined by XCF status monitoring), it most likely will be unable to process and respond to an intracluster command; if a nucleus is slow to respond to an intracluster command, it might or might not have a heartbeat monitor exception (a nucleus may appear to be active to XCF but be unable to respond to an intracluster command). If the ADARUN MXMSGWARN parameter for a cluster nucleus is nonzero (read *Using Cancel Alerts*), it produces warning messages (ADAX9B or ADAX9C) when intracluster communication with other nuclei in the cluster is too slow, Likewise, when XCF status monitoring determines that a nucleus is missing its heartbeat updates, it produces warning

messages (ADAX22 and ADAX04). You can use an automated mechanism set up at installation to raise an alert or take other appropriate action based on the existence of these messages, as they identify existing or potential problems in the cluster.

XCF status monitoring uses an ADARUN parameter, MXSTATUS, to activate XCF member-level status monitoring and to specify the monitoring interval (in seconds). In addition, the `DMEMTB` operator command includes a flag in its member state table messages indicating whether a system or message-level status monitoring exception was encountered and whether a message was issued for the exception.

To complement the new ADARUN MXSTATUS parameter, an operator command, `MXSTATUS`, allows you to change the MXSTATUS setting dynamically, while the database is running. In addition, the `DPARM` operator command's output information includes the settings of ADARUN parameters related to Adabas Cluster Services alert and timeout enhancements.

Note:
The MXSTATUS parameter and operator command are only used by Adabas Cluster Services and not by Adabas Parallel Services. Adabas Parallel Services does not use XCF and ignores this parameter and setting.

For more information about the MXSTATUS parameter, read *MXSTATUS : Member-Level XCF Status Monitoring Heartbeat Interval* . For information about the updated `DPARM` and `DMEMTB`, and `MXSTATUS` operator commands, read *Adabas Cluster Nucleus Operator Commands* .

Using Messaging Performance Statistics

Adabas Cluster Services messaging statistics provide information about the performance of message transmission events. These statistics can be used to determine the impact of messaging on system performance and to determine how to set the ADARUN MXMSG and MXMSGWARN parameters related to the other alert and timeout enhancements in Adabas Cluster Services.

The performance statistics are provided in the termination statistics of an Adabas nucleus as well as in response to the `DXMSG` operator command. The performance statistics are split into those that are subject to the ADARUN MXMSG parameter setting and those that are not; after each is reported separately in the output, a combined report is provided containing the summarization of the two for all messages.

For more information about the `DXMSG` operator command, read *Adabas Cluster Nucleus Operator Commands* .