

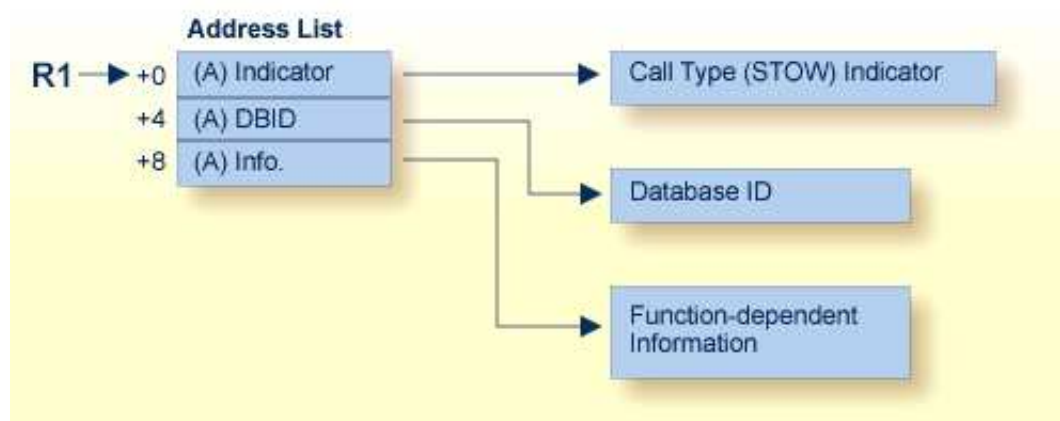
# User Exit 8 (Operator Interface)

This user exit receives control from the Adabas nucleus whenever the nucleus starts or stops, or whenever the nucleus or an Adabas utility receives a message from or sends a message to the operator. User exit 8 can be used to provide specific instructions to the operator

- when the nucleus starts and (normally) stops operation;
- as added information when Adabas sends console messages to the operator;
- to confirm commands entered by the operator.

User exit 8 is invoked

- (MODE=MULTI only) after Adabas startup, as soon as the nucleus is able to answer calls from user programs. At this point, the nucleus is now active.
- immediately after the Adabas nucleus or utility issues a console operator message. The user exit call is in addition to the standard message processing; the message itself cannot be changed during the user exit.
- after the Adabas nucleus or utility receives an operator command. The exit is called before the command is actually processed, and can reject or replace the command. The command cannot be modified in its original area.
- before a normal Adabas nucleus stop. At this point, the nucleus is no longer active; any more nucleus calls result in response code 148. This exit is not called if the nucleus ends abnormally.



## Operator Interface User Exit (8) Parameters

This chapter covers the following topics:

- Input Parameters

## Input Parameters

Parameter	Usage
0 (R1)	<p>Address of the byte containing the call type ("STOW") indicator:</p> <p>S        called at nucleus start</p> <p>T        called at normal nucleus termination</p> <p>O        called with an operator message to the nucleus/utility</p> <p>W        called with a nucleus/utility message to the operator</p>
4 (R1)	Address of the fullword containing the database ID.
8 (R1)	<p>Address of variable-length message-related information for "O" and "W" type calls. The information at this address has the following format: Call format:</p> <p>O        The one-byte message length, followed by the alphanumeric message. The length excludes the length byte itself. If the message is to be changed, location 8(R1) must point to the new message on return. This message is structured as described above. If the message is to be suppressed, location 8(R1) must point to a two-byte field containing X'0140'.</p> <p>W        8(R1) points to the message, which has the following structure:</p> <pre>DC  H'message-length+4' DC  H'0' DC  C'message-text'</pre>