

Verifying the Installation

At the end of the installation process, you should verify that the installation was successful by performing the steps described in this section.

- Verify Client Communication with the Transaction Manager
 - Verify Two-Phase Commit Operation
-

Verify Client Communication with the Transaction Manager

▶ **To verify client communication with the transaction manager:**

1. If it is currently executing, stop the TP system where you will use Adabas Transaction Manager Online Services.

This must be a TP system in which the Adabas Transaction Manager client has been installed, with client runtime controls `ATM=ON` and System Coordinator Group Name identifying the group within which the transaction manager is operating.

2. Start the transaction manager.
3. Start (or restart) the TP system so that the Adabas Transaction Manager client is loaded and initialized.
4. Log on to the Adabas Online System (AOS). Select Adabas Transaction Manager to invoke the Online Services application `SYSATM`.
5. The transaction manager is identified automatically by the Adabas Transaction Manager client proxy when you enter `SYSATM`.

One of the listed items should be recognizable either as the Terminal ID you noted in step 4 or the User ID of your current session, depending on your TP environment. This verifies the basic operation of the Adabas Transaction Manager client proxy.

6. Select Special Services, then Fix Display.

The default fix display is for the local client, but by choosing PF4 'Other' and marking the Coordinator Daemon option along with the Node ID of the System Coordinator daemon within which the transaction manager is operating, the transaction manager fix display is shown. This verifies that communication with the transaction manager is functioning correctly, even if the list of fixes is empty.

7. Terminate your `SYSATM` session.

Verify Two-Phase Commit Operation

▶ **To verify two-phase commit operation:**

1. Start two databases with the parameter setting ADARUN DTP=RM: one database containing a standard Employees file and the other database containing a standard Vehicles file.
2. In library SYSMT vrs (where vrs is the version, revision, and system maintenance level of Adabas Transaction Manager), modify the test program DEMODTP so that it correctly refers to the Employees and Vehicles files.
3. Execute the DEMODTP program, supplying a non-zero value when prompted for a new personnel number.
4. When the message “About to commit” is displayed
 - start a new session in your TP system
 - log on to the Adabas Online System (AOS)
 - select Adabas Transaction Manager, then Transaction Manager Information, and then Active Transactions.

The display of your pending transaction should indicate that two databases are involved in it.

Note:

This test is not valid in a CICS/RMI environment in which Natural executes in pseudo-conversational mode. In this mode, the INPUT statement that displays the message “About to commit” causes CICS to take a syncpoint at end-of-task and commit the pending Adabas updates. To verify correct operation in a CICS/RMI environment, this test should be done in a Natural session that was started with PSEUDO=OFF, or for which the client runtime control Adabas transaction dynamics has been set to FULL or FORCED (depending on the setting of the client runtime control Open distributed transaction support).

Note:

This test is not valid in an IMS TM environment in which RRMS acts as the superior transaction coordinator. In such an environment, the TM control Open distributed transaction (system) should take the value NO.

5. Display additional details of the transaction by marking it on the screen and pressing Enter.
6. Return to the session where your transaction is waiting to be completed and press Enter.
7. Use SYSATM to check that the transaction no longer appears in the Transaction Manager Information “Active Transactions” list.
8. Execute the program DEMODTP again to ensure that the personnel number was changed consistently in the Employees and Vehicles files.

You can now be assured of global transaction integrity in client environments and databases where the software is installed.