

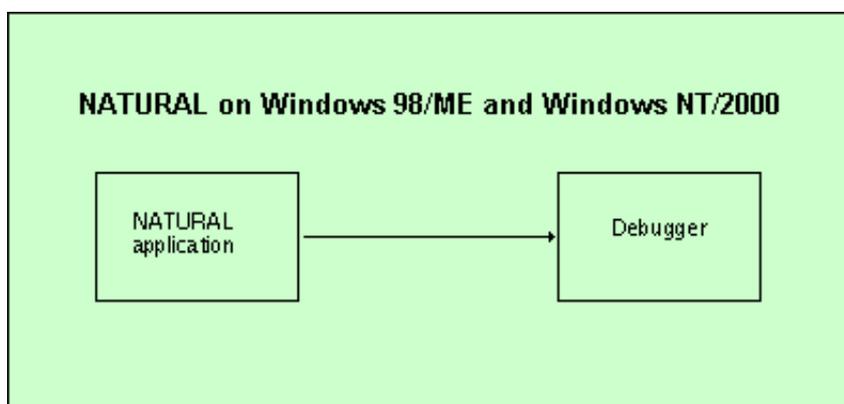
# Local and Remote Debugging

The following topics are covered below:

- Local Debugging
  - Remote Debugging
  - Setting Up Your Environment
  - Running a Natural Remote Debugging Session
- 

## Local Debugging

Local debugging, as opposed to remote debugging, is done with a debugger integrated in Windows. This debugger can be used to debug Natural applications running on the same computer.

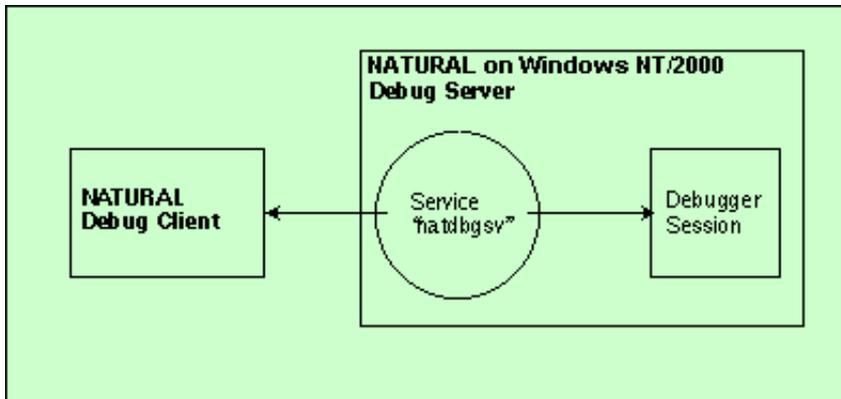


## Remote Debugging

There are several scenarios of how you can use remote debugging: a single Natural client runs under the control of one remote debugging session or a distributed Natural application runs under the control of several remote debugging sessions. Such a distributed application may include both Natural RPC and DCOM servers or even components not written in Natural, such as Visual Basic clients.

- Scenario 1: Debugging a Single Natural Application
- Scenario 2: Debugging a Distributed Natural Application
- Scenario 3: Debugging The Natural Part of a Heterogeneous Application

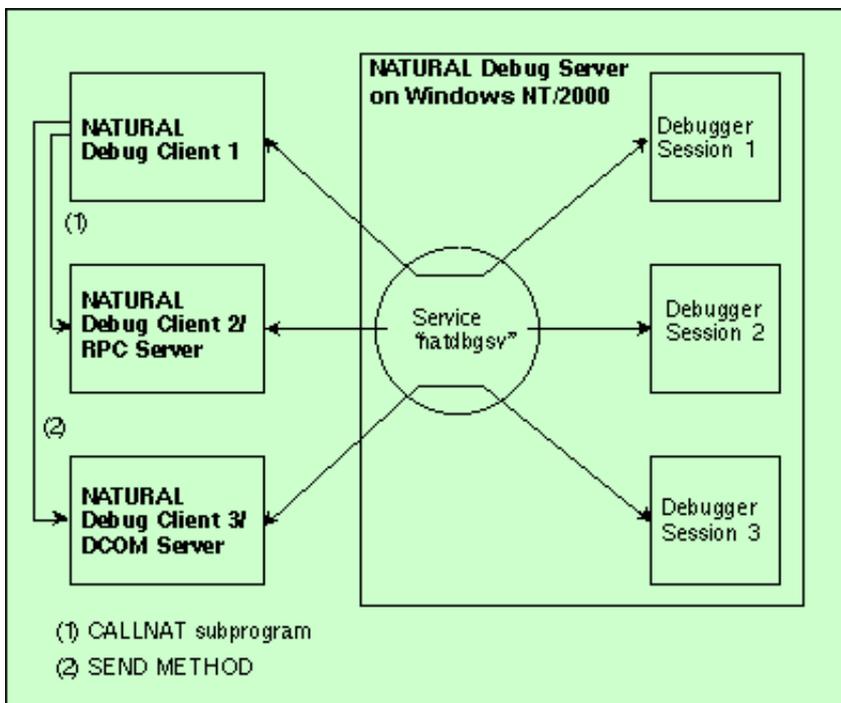
### Scenario 1: Debugging a Single Natural Application



### Scenario 2: Debugging a Distributed Natural Application

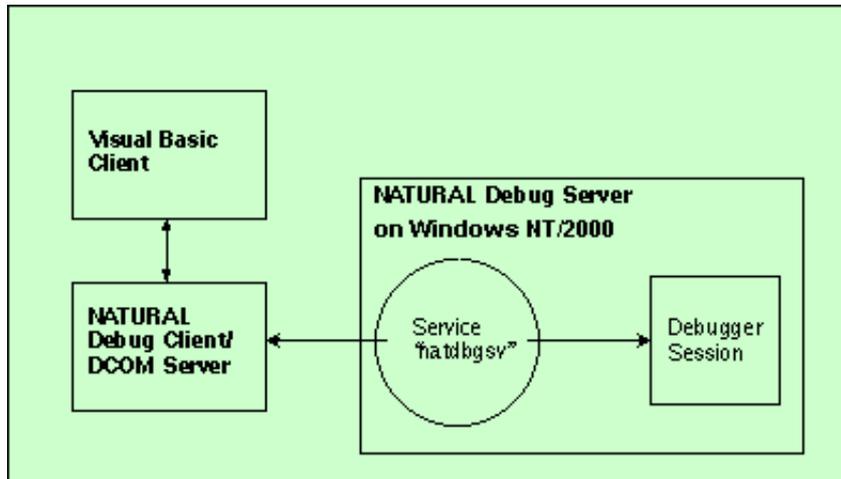
To debug each component of the following distributed Natural application, you enter "DEBUG *objectname*" in the command line of Natural Debug Client 1. The first time the Natural Debug Client calls a subprogram on a Natural RPC server, for example, a new debug session is opened for the RPC server. Then, the RPC server's processing is debugged. The debug session is closed as soon as the RPC server is terminated.

The same applies to a Natural DCOM server.



### Scenario 3: Debugging The Natural Part of a Heterogeneous Application

As in Scenario 2, the first time a method on the DCOM server is called, a new debug session is opened for the DCOM server, the DCOM server's processing is debugged, and the debugger session is closed as soon as the DCOM server is terminated:



## Setting Up Your Environment

### Windows NT/2000 Side

Install the Natural remote debugger or install Natural for Windows NT 4 PL 6 or Windows 2000. This also installs the Natural Remote Debugging Service.

To uninstall the Natural Remote Debugging Service, enter "natdbgsv -u" in the command line. To view the current service's port name and version, enter "natdbgsv -s". To re-install the service on a different port, uninstall it first and then enter "natdbgsv -i *portnumber*", where *portnumber* is the value of the RDPORT profile parameter.

#### Note:

If you install the Natural Remote Debugging Service on a port other than 2500, you have to change the value of the RDPORT profile parameter accordingly on the UNIX client computer.

### Natural Side

Start Natural with the following three profile parameter settings:

- RDACTIVE=ON;
- RDNODE=*nodename*, where *nodename* is the name of the Windows NT/2000 server;
- RDPORT=2500 (or another port number, depending on which port you have installed the Natural Remote Debugging Service on the Windows NT computer).

## Running a Natural Remote Debugging Session

To run a Natural Remote Debugging Session:

1. Ensure that the Windows NT/2000 server has been started and that the service "natdbgsv" is active.
2. Enter "DEBUG *objectname*" in the Natural command line, where *objectname* is the name of the Natural object you wish to debug.

The Natural Debugger becomes active on the Windows NT/2000 computer, with the following window caption: "Debugging remote Natural client (\\*nodename*::*username*::*process-id*)", where *nodename* is the name of the computer where Natural is running, *username* is the name of the Natural user and *process-id* is the Natural process ID.