

Способы реализации COM серверов

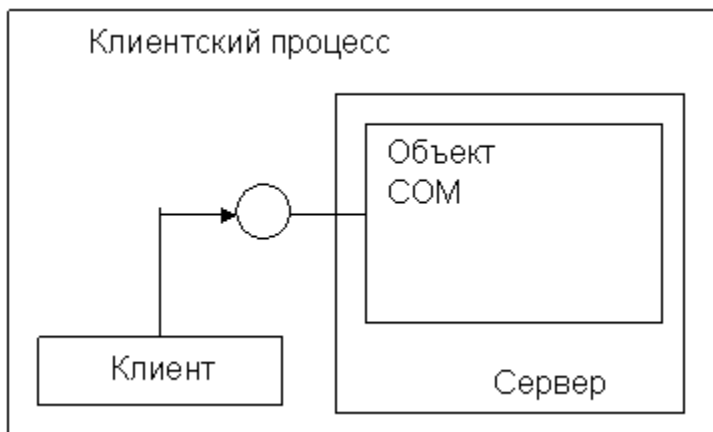
При работе с объектами COM клиент не знает, где находится объект. Он просто обращается к объекту посредством его интерфейса. Далее библиотека COM выполняет все необходимые шаги для удовлетворения вызова клиента.

Эти шаги зависят от конкретного местонахождения объекта COM: в том же процессе, что и клиент, в другом процессе на машине клиента или на другой машине в сети. В зависимости от этого различают три вида серверов COM.

Сервер в клиентском процессе (In-process server). Это библиотека DLL, которая выполняется в адресном пространстве процесса клиента. Например, элемент ActiveX, внедренный в Web страницу, выполняется в Internet Explorer или другом браузере. Это значит, что объект ActiveX загружается на машину клиента и выполняется в том же процессе, что и Web браузер. Клиент обращается к объекту COM путем прямого вызова интерфейса COM.

Клиентский процесс
Сервер
Объект

COM
Клиент



посредством COM библиотек.

Клиентский процесс
Локальный сервер
Клиент
Заместитель

(проху)
Заглушка

(stub)
Объект

COM
RPC
COM

Рис. Сервер в клиентском процессе

Локальный сервер (local server). Он представляет собой другое приложение (файл *.exe), которое выполняется в другом адресном пространстве, но на том же компьютере, что и клиентское приложение. Например, обработка таблицы Excell, внедренной в документ Word, выполняется приложением Excell. Локальный сервер связывается с клиентом

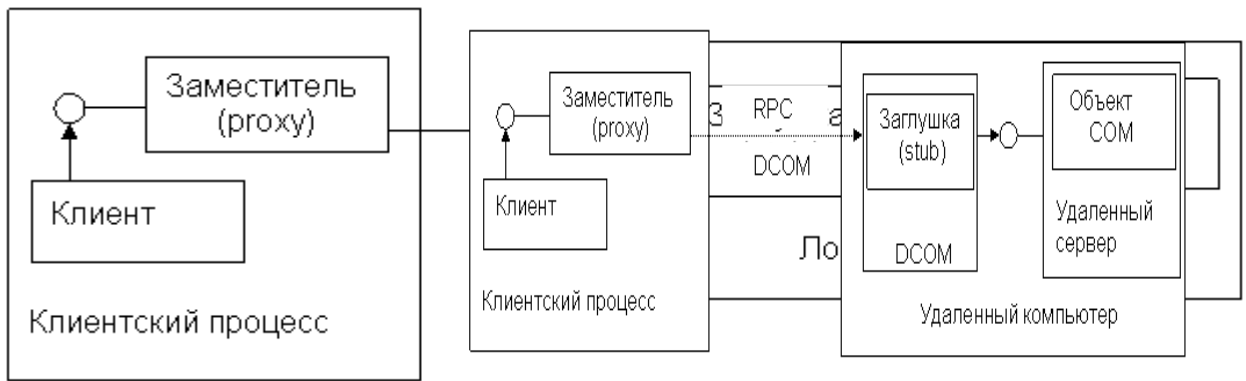


Рис. Локальный сервер

Когда объект COM принадлежит другому процессу на том же компьютере, что и клиентское приложение, или вовсе на другом компьютере (в сети), COM использует так называемого "заместителя" (проху) для того, чтобы инициировать удаленный вызов процедуры (remote procedure call – RPC). Так как заместитель находится в клиентском процессе, то с точки зрения клиента обращения к интерфейсам выглядят так же, как и для случая размещения сервера в клиентском процессе. Заместитель перехватывает вызовы клиента и отправляет их туда, где находится сервер. Механизм, который позволяет клиенту получить доступ к объекту, находящемуся в другом процессе или на другом компьютере (невидимо для клиента), называется маршалингом или маршализацией.

Клиентский процесс
 Удаленный компьютер
 Клиент
 Заместитель

(проху)
 RPC
 DCOM
 DCOM
 Удаленный сервер
 Заглушка

(stub)
 Объект

COM

Удаленный сервер (remote server). Он представляет собой библиотеку (DLL или OCX) или другое приложение, которые выполняются на другом компьютере, а не на машине клиента. Например, клиентское приложение, использующее базу данных, связывается с приложением, выполняемым на другом компьютере в сети. В этом случае удаленный сервер использует интерфейсы DCOM.

Рис. Удаленный сервер

Различие между локальным и удаленным сервером состоит в применяемом способе (и инструментальных средствах) взаимодействия клиента и сервера: в первом случае используется COM, а во втором – DCOM.

Если сервер реализован как библиотека (а библиотека не может выполняться как самостоятельное приложение), то COM создает специальное приложение-суррогат (surrogate), в рамках которого и выполняется библиотека.

Маршалинг. Механизм маршалинга позволяет клиентскому приложению делать вызовы методов интерфейса для объекта COM, который находится в другом процессе или на другом компьютере.

При любом вызове функций посредством интерфейса клиентское приложение помещает фактические параметры функции в стек и выполняет ее вызов. Если функция не находится в клиентском процессе, то вызов передается заместителю. Заместитель упаковывает параметры в пакет и передает их удаленному объекту. Заглушка COM объекта распаковывает пакет, помещает аргументы в стек и вызывает объект COM. Таким образом объект COM обрабатывает запрос клиента в своем собственном адресном пространстве.

Какой именно маршалинг будет реализован, зависит от реализации COM объекта. Стандартный механизм реализован интерфейсом *IDispatch*. Кроме того, объекты могут самостоятельно выполнять маршалинг, однако это довольно сложно.

Отметим, что технология Microsoft Transaction Server (MTS) обеспечивает дополнительную поддержку для удаленных объектов.