

Authorized Independent  
Training Provider for IBM Software



**Е. Киселев**

# **Системное администрирование *IBM Lotus Domino R 8.5***

**Учебный курс**

# 1 Предисловие

Эта книга представляет собой учебный материал к курсу по администрированию *IBM Lotus Domino R8.5*, который я читаю в компании *InterTrust*. Он построен на базе трех авторизованных курсов *IBM*:

- *IBM Lotus Domino R8 Operating Fundamentals*
- *Managing the IBM Lotus Domino R8 Servers and Users*
- *Building the IBM Lotus Domino R8 Infrastructure*

При этом структура данного курса, а также последовательность изучения тем существенно переработаны.

Я постарался сместить "центр тяжести" в сторону практических занятий. При этом не ставится задача научить слушателей производить такие-то действия для того, чтобы получился такой-то результат. Гораздо важнее, как мне кажется, понимание внутренней логики всей системы. Такое понимание позволяет не только выполнять перечень типовых процедур, но и находить пути решения в нестандартных ситуациях.

Группировка по темам существует, но материал представлен в таком порядке, в каком он рассматривался во время занятий. Поэтому к некоторым темам мы будем возвращаться неоднократно – на разном уровне и с различными конфигурациями.

С самого начала все строится своими руками – никаких предустановленных серверов и клиентов. Поэтому в начале курса рассказ об основных понятиях и принципах совмещается с первоначальной установкой рабочей схемы.

Во время занятий этот материал не используется. Я вообще уверен, что никакие вспомогательные материалы на занятиях не нужны – они только мешают думать самостоятельно. Назначение этой книги – уже после окончания курса помочь слушателю вспомнить какие-то детали, которые успели вылететь из головы к тому моменту, когда в них возникла реальная необходимость.

Книга также вполне может оказаться полезной при самостоятельном изучении *Domino*. В этом случае важно иметь возможность проведения практических работ.

Существенная часть материала этой книги (те разделы, которые не претерпели изменений после перехода на новую версию) была опубликована в книге "Системное администрирование *IBM Lotus Domino R7*".

В некоторых разделах книги были использованы материалы из книги Н.Н. Ионцева "Почтовая система сервера *Lotus Domino R6* и ее конфигурирование", а также из его курса "Системное администрирование *Lotus Domino R6*". Эти разделы помечены соответствующей сноской.

## 2 Установка и первоначальная настройка серверов и клиентов

### 2.1 Схема класса – один домен, одна организация

Первая схема, с которой нам предстоит работать первую половину курса, представляет собой одну организацию, в которую входят все наши сервера. Кроме того, домен у нас также один. Что такое организация и домен – об этом чуть позже.

Имена и IP-адреса наших компьютеров выглядят так:

172.16.16.61 server1.intrust.ru

172.16.16.62 server2.intrust.ru

...

172.16.16.60 Hub.intrust.ru

Во время занятий мы будем привязаны к этим названиям.

Расположение и нумерация рабочих станций в классе:

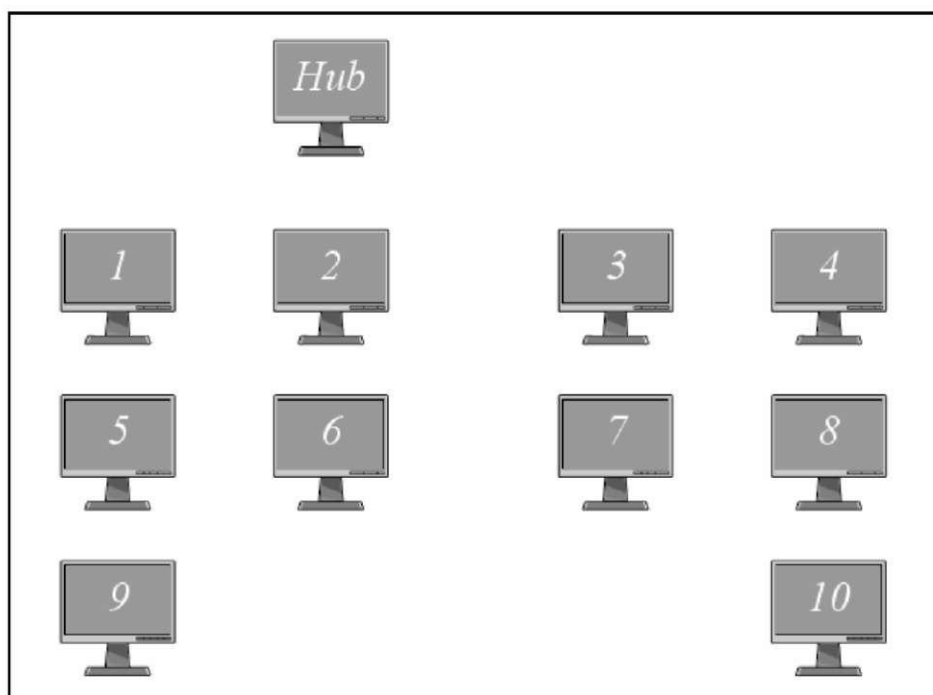


Рис. 2-1. Схема класса.

### 2.2 Установка первого сервера

Все начинается с установки первого сервера домена. Она происходит в два этапа.

Первая стадия – выбор компонентов и копирование файлов. Эту фазу часто называют инсталляцией. Вторая стадия – конфигурирование системы. При первом запуске после инсталляции сервер видит, что ему для работы не хватает многих необходимых вещей, и запускает *Java*-конфигуратор, в котором и производится первичная настройка сервера.

#### 2.2.1 Инсталляция сервера

Итак, инсталляция первого сервера. Дистрибутив сервера находится в директории *C:\Distr\Domino R8*. Запускается файл *Setup.exe*, и процесс пошел. Вначале предлагается прочитать лицензионное соглашение, с которым мы согласимся не читая. Затем предлагается

выбрать директорию, в которую будут установлены программные файлы сервера. По умолчанию это *C:\Program Files\Lotus\Domino*, мы же выберем местом установки *C:\Lotus\Domino* (нам так будет проще в дальнейшем).

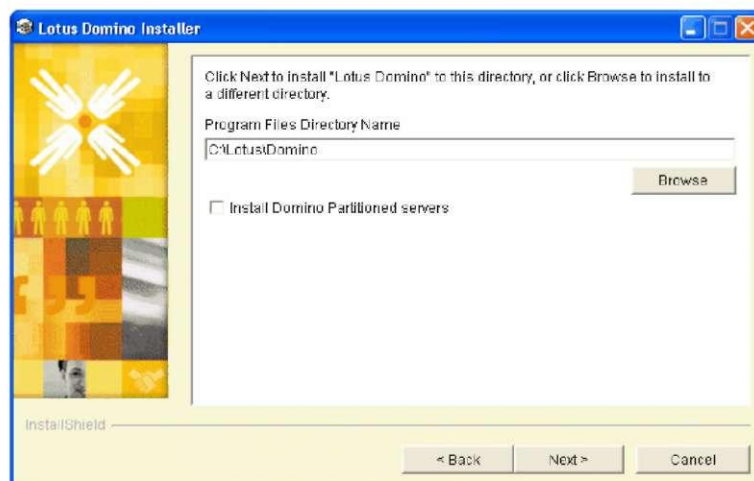


Рис. 2-2. Выбор программной директории сервера.

Чек-боксе "*Install Domino Partitioned servers*" оставим пустым. Мы не будем устанавливать *Partitioned Servers*, но упомянуть о такой возможности следует. *Domino* предусматривает возможность установки нескольких независимых серверов на один компьютер. Количество серверов ограничено только производительностью компьютера, рекомендуется иметь не менее одного процессора на сервер, оптимально – два. Эти сервера могут использовать либо разные *IP*-адреса, либо разные порты, в зависимости от настроек. Программная директория у них общая, а директории данных – у каждого своя. С моей точки зрения – это далеко не лучший выбор, так как остается открытым вопрос о финансовой целесообразности такого решения, да и администрировать сервера, установленные на одном компьютере, крайне неудобно. Кроме того, с развитием технологии виртуализации предпочтительным вариантом для размещения нескольких серверов *Domino* на одном компьютере является установка отдельной виртуальной машины для каждого сервера.

Следующий экран позволяет выбрать директорию, где будут храниться базы данных и их шаблоны.

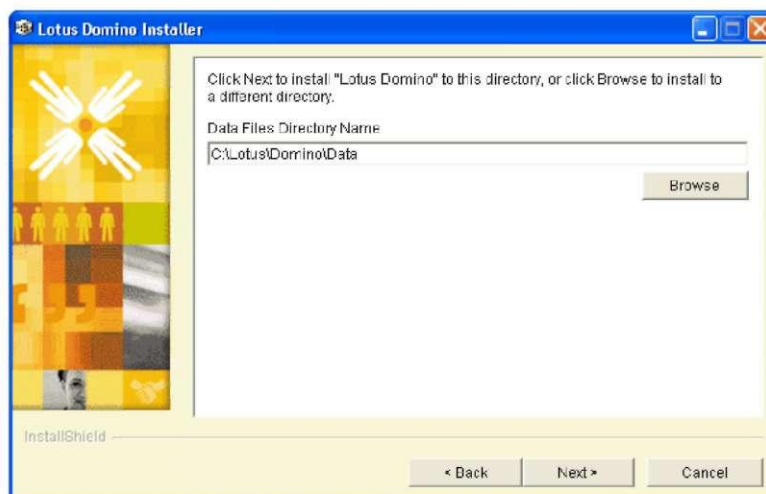


Рис. 2-3. Выбор директории данных.

Рекомендуется располагать программные файлы и данные на разных дисках, но мы это правило нарушим за неимением другого диска на наших компьютерах.

Следующим шагом предлагается выбрать тип установки:



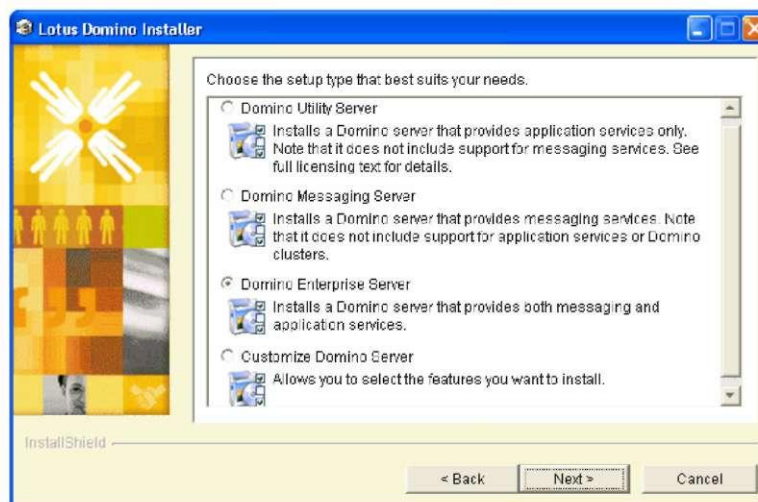


Рис. 2-4. Тип установки.

Возможны четыре варианта установки:

- *Domino Utility Server*: устанавливается как сервер приложений, без поддержки почтовых служб;
- *Domino Messaging Server*: устанавливается как почтовый сервер, без поддержки приложений и объединения серверов в кластер;
- *Domino Enterprise Server*: все в одном флаконе, устанавливается полный комплект
- *Custom*: набор функций выбирается вручную.

Все варианты поддерживают установку сегментированных (*Partitioned*) серверов. Поддержку *xSP* (о ней позже) обеспечивает только *Domino Enterprise Server*.

После этого остается только проверить, все ли сделано правильно:



Рис. 2-5. Итоговая таблица настроек.

На этом первая фаза установки сервера завершена. Теперь в директории *Domino* находится набор библиотек (\*.dll) и исполняемых файлов (\*.exe). Имена большинства исполняемых файлов начинаются на "N": "N\*\*\*.exe". Это серверные процессы, которые сервер знает под такими именами:

*NRouter.exe* => *Router*

*NPOP3.exe* => *POP3*

*NEvent.exe* => *Event*

И так далее.

Два файла являются стартовыми: *NServer.exe* и *NService.exe*. Как можно понять из названий, первый из них запускает сервер как приложение, второй – как сервис.

Функционально это одно и то же. Разница, конечно, есть, но она имеет отношение только к правам и безопасности, поэтому о ней позже.

В директории *Domino\Data* на данный момент располагаются в основном шаблоны приложений. Самих приложений пока почти нет.

### 2.2.2 Первоначальная настройка первого сервера домена

Вторая стадия – первоначальная настройка сервера. При первом включении сервера запускается *Java*-конфигуратор. Но сначала нужно ответить на вопрос: как мы собираемся запускать сервер – как сервис или как *Windows*-приложение?

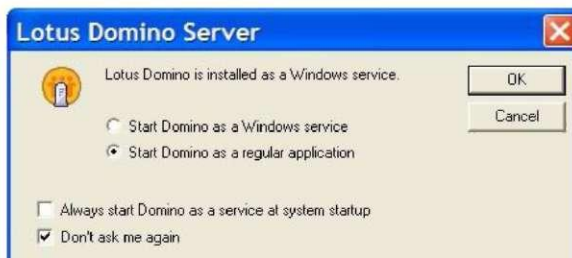


Рис. 2-6. Тип запуска сервера – как сервис или как приложение.

Имеется в виду следующее. Если выбрать запуск *Domino* как *Windows*-приложения, то будет стартовать *NServer.exe* от имени текущего пользователя. Если же выбирается другой вариант, то будет запущен *NService.exe*, причем как служба, от имени системной учетной записи:

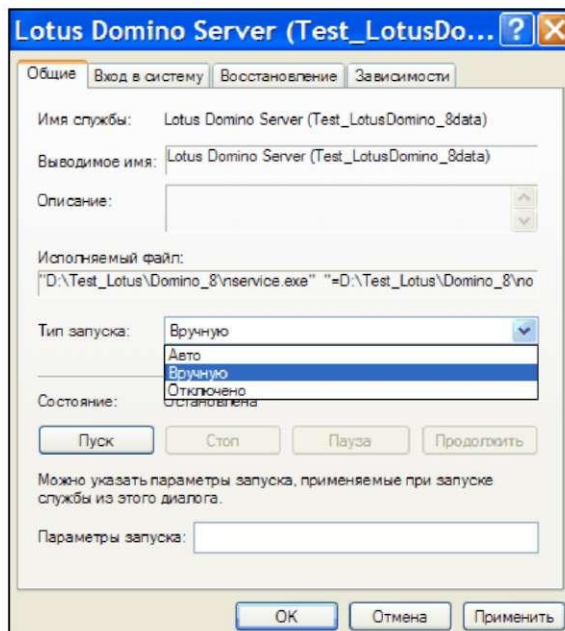


Рис. 2-7. Запуск *Domino* как сервиса.

Обычно выбирают вариант "запускать как сервис" – это удобнее по разным причинам. Например, в такой конфигурации возможен автоматический запуск сервера при старте операционной системы. В *Domino R8* по умолчанию сервис запускается автоматически. Нам же, в учебной конфигурации, удобнее второй вариант – запуск *Domino* как обычного *Windows*-приложения. Так мы и поступим. Для сервиса "*Lotus Domino Server*" изменим тип запуска с автоматического на ручной, а в окне Рис. 2-6 выберем "*Start Domino as a regular application*" и "*Don't ask me again*".

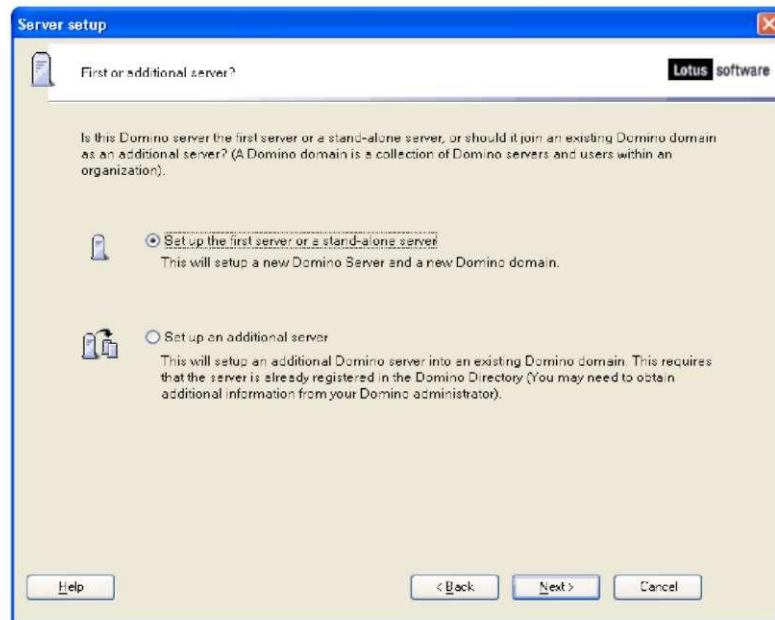


Рис. 2-8. Выбор варианта – первый или дополнительный сервер?

Весьма принципиальный момент. Для работы сервера требуется наличие *ID*-файла, *Domino Directory*, ряда других системных баз данных. Выбрав вариант "*First server or stand-alone server*" (первый или единственный сервер), мы предписываем *Java*-конфигуратору создать их. Для этого потребуются указать, как будет называться сервер, организация и домен. Вариант "*Additional Server*" (дополнительный сервер) подразумевает, что все эти базы, а также *ID*-файл сервера, уже существуют, и требуется лишь указать, откуда их взять. Названия им также уже даны, ничего придумывать не надо.

Поскольку сейчас у нас ничего этого нет, следует выбрать вариант "первый сервер".

Следующим шагом предлагается дать имя серверу:

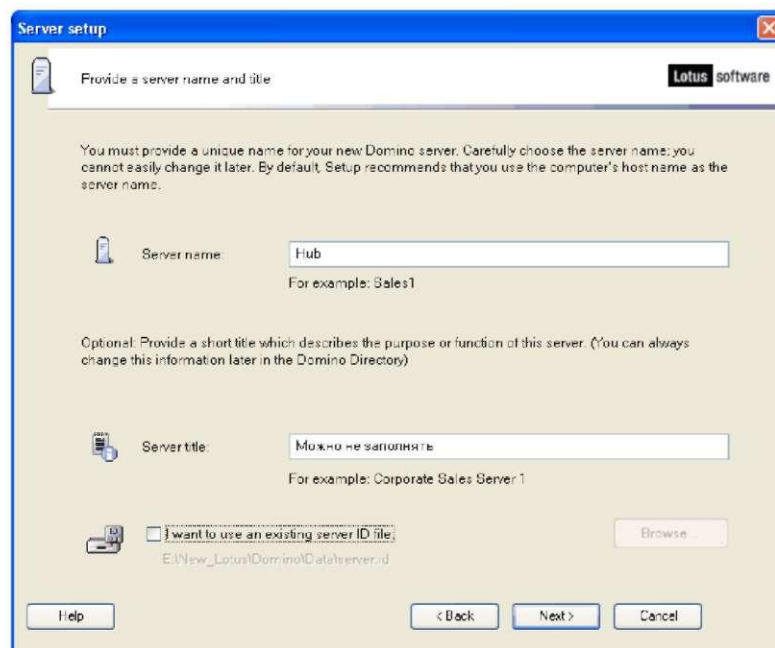


Рис. 2-9. Имя сервера.

Рекомендуется, чтобы имя сервера совпадало с *DNS*-именем хоста, на котором он располагается. Это правило не является обязательным, но может существенно упростить работу администратора. Крайне желательно, чтобы имя сервера содержало только буквы *A-Z*, цифры *0-9*, дефис "-" и не включало пробелов и символов подчеркивания. Отсутствие пробелов в имени не только позволит уйти от проблем с определением *IP*-адресов, но и поможет



администратору избежать дополнительных нажатий на кнопки: иначе все консольные команды, обращенные к серверу с составным именем, придется заключать в кавычки.

Поле "*Server Title*" – не обязательно, это просто комментарий.

Чек-бокс "*I want to use an existing server ID file*" в нашем случае не применим – у нас нет существующего *ID*-файла первого сервера, конфигуратору придется его создать. Иногда эта опция бывает полезной при радикальной переустановке сервера. Например, когда после краха от сервера ничего, кроме *ID*-файла, не осталось. Наш первый сервер мы назовем "*Hub*". В централизованной структуре так обычно называют центральный сервер ("*Hub*" – ступица, топологическую схему с выделенным центральным сервером называют "*Hub and Spokes*", т.е. "ступица и спицы").

Следующий экран предлагает дать имя будущей организации *Domino*. Это очень важный момент, стоит остановиться на нем подробнее.

В *Notes/Domino* имеются два принципиально разных принципа структурного деления серверов и пользователей. Это деление на домены *Notes* и деление на организации *Notes*. Их часто путают, хотя они никак не связаны между собой.

**Домен *Notes*** (не надо путать с доменом *Internet!*) – это множество серверов *Domino*, на которых расположена реплика одной и той же основной конфигурационной базы данных. В этой базе (*Names.nsf*), которая до 4-й версии *Notes* включительно называлась Адресной Книгой Домсена (*Names&Address Book, NAB*), а начиная с 5-й версии, носит название *Domino Directory*, кроме имен и адресов пользователей, находятся основные настроечные параметры всех серверов домена, соединения между ними, общие для всего домена конфигурационные документы и много всяких других полезных вещей. За счет регулярной репликации между серверами эти реплики поддерживаются в синхронном состоянии. В результате в каждый момент времени каждый сервер домена *Notes* представляет себе топологию домена, настройки своих соседей по домену, состав пользователей и многое другое, точно так же, как и остальные сервера в том же домене *Notes*. Не только *Domino Directory* является общей для серверов домена. Такие базы, как *Admin4.nsf* (база административных запросов), *Events4.nsf* (настройки мониторинга), часто – *CertLog.nsf* (журнал сертификации), *Catalog.nsf* (каталог баз домена) и некоторые другие, являются общими для всего домена *Notes*. Таким образом, в *Notes/Domino* понятие домена связано с общими для серверов, входящих в домен, конфигурационными базами данных, в первую очередь *Domino Directory*. Оно относится скорее к топологии и к правилам маршрутизации почты, чем к правам доступа и безопасности. Это понятие не является иерархическим, так как домены не делятся на "родителей" и "потомков", "старшие" и "младшие" домены.

**Организация *Notes*** – это множество серверов и пользователей, чьи *ID*-файлы имеют сертификаты, выданные одним и тем же корневым сертифицикатором и его потомками. Все пользователи и сервера одной организации доверяют одному и тому же сертифицикатору. Следовательно, они доверяют и друг другу.

Это принципиально иерархическое понятие, так как с помощью корневого сертифицикатора может быть зарегистрирован дочерний, с его помощью – сертифицикатор следующего, более низкого уровня, и так далее. Всего в организации *Notes* может быть до пяти уровней сертифицикаторов, включая корневой. Сертифицикатором любого из этих уровней может быть выдан сертификат для *ID*-файла пользователя или сервера. При этом все они принадлежат одной организации, поскольку родительский (корневой) сертифицикатор у них общий. Имена пользователей и серверов отражают всю цепочку сертифицикаторов, которыми они были заверены. Таким образом, понятие организации имеет отношение к *ID*-файлам, сертификатам и сертифицикаторам, а также к правилам построения имен. Организации *Notes* не имеет отношения к топологии и маршрутизации почты.

Совершенно не обязательно, чтобы организация и домен *Notes* совпадали по составу. В одном домене может быть несколько организаций, и наоборот – в одной организации может быть несколько доменов. Если адресная книга домена (большинство администраторов *Domino* по привычке продолжают называть так *Domino Directory*) общая для нескольких серверов, а сертификаты в их *ID*-файлах выданы разными корневыми сертифицикаторами, это значит, что несколько организаций объединены в один домен. Если корневой сертифицикатор общий, а адресные книги разные – это значит, что одна организация разделена на несколько доменов.

Очень важно не путать эти два понятия (а путают постоянно!). Настолько важно, что я даже не поленился нарисовать две схемы.



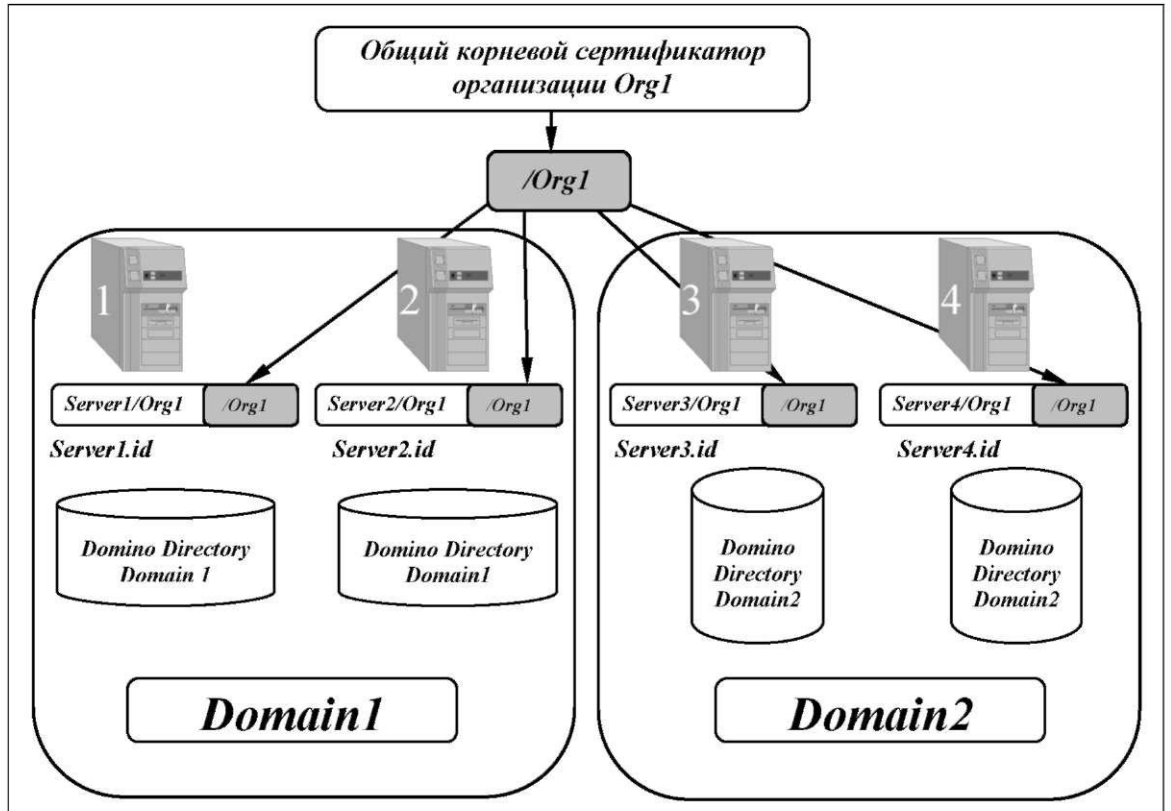


Рис. 2-10. Одна организация (Org1) разделена на два домена (Domain1 и Domain2).

На Рис. 2-10 показана одна организация, разделенная на два домена. Организация одна, поскольку в ID-файлах всех серверов (и пользователей, которые не изображены на схеме из соображений наглядности) имеются сертификаты, выданные одним и тем же сертификатом Org1. Домены разные, поскольку имеется две базы Domino Directory. Две реплики одной базы лежат на серверах Server1 и Server2, две реплики другой базы – на серверах 3 и 4.

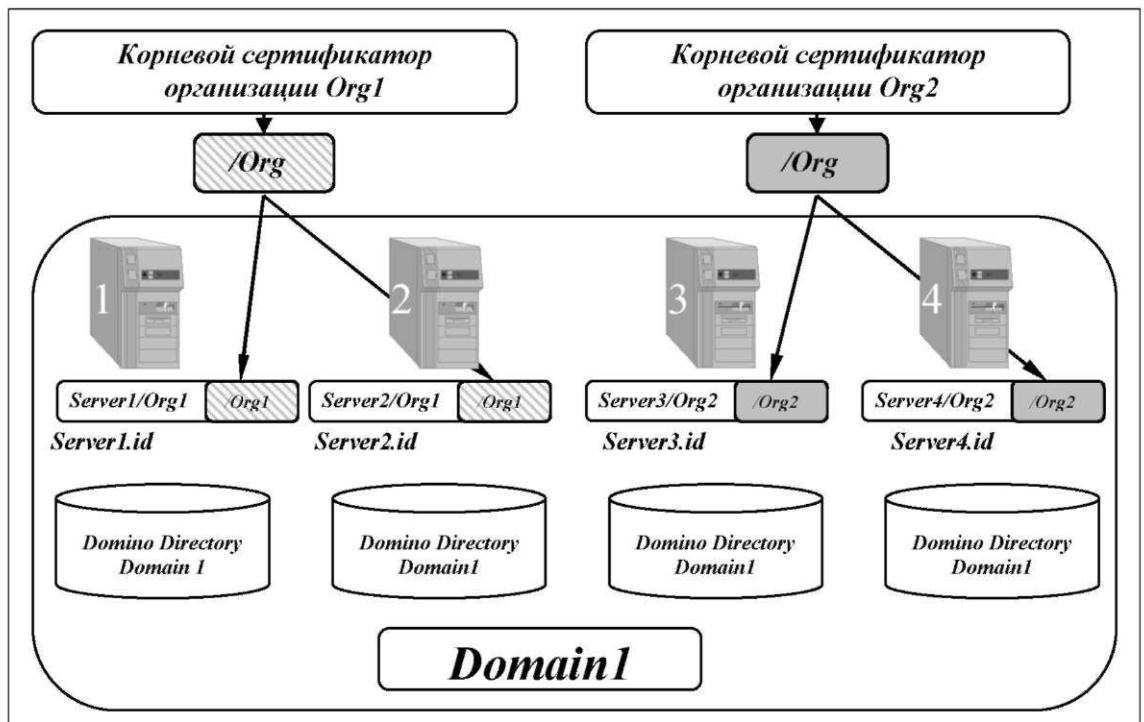
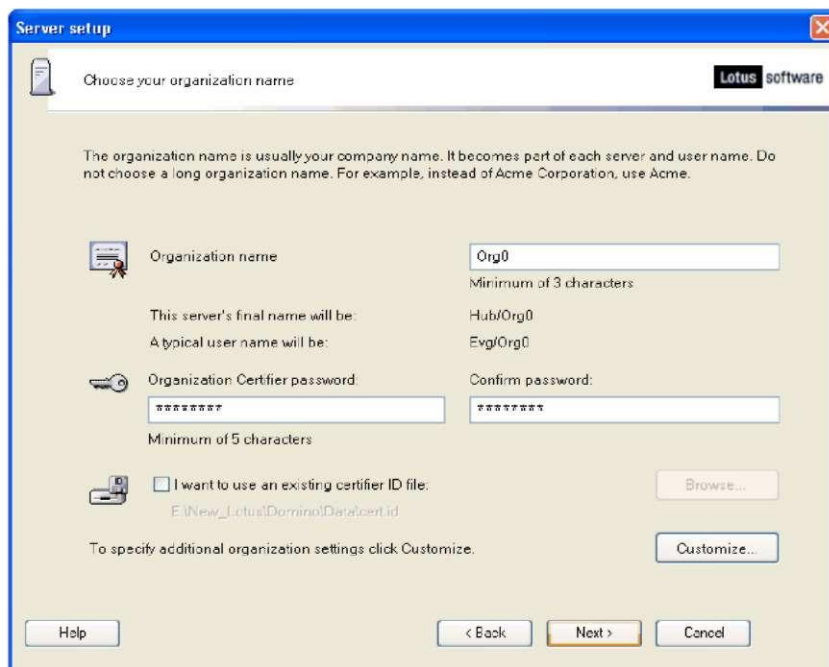


Рис. 2-11. Две организации (Org1 и Org2) объединены в один домен.

На Рис. 2-11 изображена схема двух организаций, объединенных в один домен. Организации две, так как в *ID*-файлах серверов *Server1* и *Server2* имеются сертификаты, выданные сертифицированным *Org1*, а в *ID*-файлах 3-го и 4-го серверов – сертификаты, выданные сертифицированным *Org2*. Домен общий, так как на всех четырех серверах находятся реплики одной и той же базы – *Domino Directory* домена *Domain1*.

И организации, и домены *Notes* изначально появляются во время процедуры конфигурирования первого сервера домена.

Поскольку в начале установки был выбран вариант "первый сервер", это означает, что *Java*-конфигуратору поручается создать корневой сертифицированный. Это зародыш будущей организации *Notes/Domino*, им (и его потомками) будут выдаваться сертификаты всем остальным серверам и пользователям.



**Рис. 2-12.** Корневой сертифицированный появляется на свет при первичной настройке первого сервера организации. Администратор определяет его имя (это и есть имя будущей организации) и его пароль.

Предполагается, что после окончания первоначального конфигурирования сервера в директории *Domino\Data* мы сможем найти файл *Cert.id*, который и является этим самым корневым сертифицированным. Он будет защищен паролем, который мы указываем здесь же, в поле "*Organization Certifier Password*". И здесь, и далее в качестве пароля мы будем использовать слово "*password*". Саму организацию мы назовем нехитрым именем "*Org0*". Поскольку имена объектов связаны с сертифицированным организации, полное имя нашего сервера будет таким: *Hub/Org0*.

Существующему корневному сертифицированному, о котором говорится в поле "*I want to use an existing certifier ID*", в нашей ситуации взяться неоткуда.

Кнопка "*Customize*" позволяет указать дополнительные настройки, связанные с расположением сервера в иерархической структуре. Кроме корневого сертифицированного, здесь можно указать организационную единицу, в которую будет входить сервер, и код страны. Это приведет к тому, что кроме корневого сертифицированного будет создан еще один, так называемый сертифицированный организационной единицы. Подробнее об этих сертифицированных будет рассказано позднее, в разделе **Ошибка! Источник ссылки не найден.**