**Введение**

Рабочие станции (компьютеры, используемые отдельными сотрудниками) являются основным инструментом работы в большинстве организаций. Именно от их надёжной работы зависят производительность пользователей и эффективность процессов в компании. Администрирование рабочих станций охватывает задачи установки, настройки, обновления и обслуживания программного обеспечения, а также управления их сетевыми параметрами.

**Ключевые аспекты:**

* Рабочие станции требуют стандартизации для снижения затрат на поддержку.
* Использование автоматизированных инструментов упрощает управление большими группами устройств.
* Качественное обслуживание рабочих станций способствует сокращению времени простоя и улучшению рабочего опыта пользователей.

### ****3.1.1. Установка операционной системы (ОС)****

Установка операционной системы — это один из ключевых этапов управления рабочими станциями. Главной задачей здесь является автоматизация процесса. Это позволяет исключить человеческие ошибки и значительно сократить время, затрачиваемое на развертывание новых устройств. Для автоматизации используются такие инструменты, как PXE Boot и Kickstart для Linux, а также Windows Deployment Services (WDS) для Windows. Эти системы позволяют настроить процесс так, чтобы установка выполнялась автоматически — от загрузки установочного образа до финальной настройки параметров системы.

Для упрощения работы администраторов создаётся стандартный образ операционной системы. Этот образ включает в себя всё необходимое: базовые программы, драйверы, обновления и настройки безопасности. Стандартные образы позволяют быстро и последовательно развертывать рабочие станции. Для разных категорий сотрудников, таких как разработчики или офисные работники, можно подготовить отдельные образы, учитывающие их специфические задачи.

Без автоматизации установка ОС становится трудоёмким и непредсказуемым процессом, что может привести к несогласованности настроек и повышению нагрузки на IT-отдел. Стандартизация и автоматизация являются критически важными для эффективного управления рабочими станциями.

### ****3.1.2. Обновление системного ПО и приложений****

Обновление программного обеспечения играет важную роль в поддержании стабильной работы рабочих станций. Регулярное обновление устраняет уязвимости, повышает производительность и обеспечивает безопасность. Одной из ключевых целей является защита от кибератак, так как устаревшие версии программ становятся лёгкой мишенью для злоумышленников.

Процесс обновления необходимо автоматизировать. Для этого используются такие инструменты, как Windows Server Update Services (WSUS) для Windows или APT для Linux. Эти системы позволяют централизованно управлять обновлениями: настраивать приоритеты, выбирать время для установки и проводить тестирование перед развёртыванием обновлений.

Тестирование обновлений перед их установкой на все рабочие станции — это обязательный этап. Сначала обновления развёртываются на ограниченной группе устройств, что позволяет выявить и устранить возможные проблемы. После этого обновление можно применять ко всей сети. Такой подход снижает риск сбоев и минимизирует влияние на пользователей.

Автоматизация и планирование обновлений делают процесс менее трудоёмким и помогают поддерживать программное обеспечение в актуальном состоянии, что критически важно для успешной работы компании.

### ****3.1.3. Настройка сетевых параметров****

Настройка сетевых параметров рабочих станций — это важный аспект их администрирования. Для упрощения процесса используется централизованное управление через протокол DHCP. Это позволяет автоматически назначать IP-адреса, шлюзы и DNS-серверы каждому устройству, избавляя администратора от необходимости вручную настраивать каждую станцию.

Ключевые узлы сети, такие как серверы, обычно получают статические IP-адреса. Это необходимо для обеспечения их доступности и стабильной работы. Динамические адреса в таких случаях могут привести к потере связи или сбоям в работе служб.

Важно избегать создания избыточных зависимостей в сетевой конфигурации. Например, использование динамического DNS в сочетании с DHCP на критически важных системах может вызвать проблемы при смене адресов. Поэтому для ключевых узлов рекомендуется использовать предсказуемые и стабильные параметры.

Правильная настройка сетевых параметров обеспечивает бесперебойную работу всей сети и облегчает её администрирование, особенно в масштабах больших организаций.

#### **3.2.1. Уверенность в завершении процессов**

Одной из важных задач администратора является контроль за тем, чтобы каждый этап установки, обновления или настройки завершался успешно. Для этого необходимо использовать систематизированные подходы, например контрольные списки. Такие списки помогают убедиться, что ни один важный шаг не был пропущен, и делают процесс более предсказуемым.

Например, после установки ОС можно проверить, настроены ли сетевые параметры, установлены ли необходимые драйверы и приложения, а также работают ли службы безопасности. Это избавляет от необходимости возвращаться к настройке устройства позже.

#### **3.2.2. Вовлечение пользователей**

В процессе стандартизации конфигураций важно учитывать мнение конечных пользователей. Если сотрудники будут вовлечены в этот процесс, они с большей вероятностью примут изменения и стандарты. Например, разработчики могут предложить, какие именно инструменты им необходимы для работы, а сотрудники отдела продаж — указать на удобные для них приложения.

Такой подход помогает избежать ситуации, когда стандарты навязываются сверху, но не учитывают реальных потребностей пользователей. Вовлечённость сотрудников повышает их удовлетворённость работой и снижает вероятность сопротивления изменениям.

#### **3.2.3. Разнообразие стандартных конфигураций**

Хотя стандартизация упрощает обслуживание, не всегда целесообразно использовать единый стандарт для всех рабочих станций. Разнообразие стандартных конфигураций позволяет учитывать различия в потребностях сотрудников. Например:

* Для дизайнеров можно включить мощные графические редакторы.
* Для программистов — инструменты разработки и виртуальные машины.
* Для офисных сотрудников — офисные пакеты и минимальный набор приложений.

Каждая конфигурация должна быть тщательно документирована и протестирована. Это уменьшит количество ошибок при её развёртывании и упростит управление.

### ****3.3. Заключение****

Глава о рабочих станциях подчеркивает важность системного подхода к их управлению. Успешное администрирование включает в себя не только установку, настройку и обновление программного обеспечения, но и внедрение стандартов, автоматизацию процессов и учёт потребностей пользователей.

Ключевым элементом является стандартизация, которая делает рабочие станции предсказуемыми и упрощает их обслуживание. Автоматизация установки операционной системы и обновлений позволяет минимизировать ошибки и сэкономить время. Настройка сетевых параметров, организованная централизованно, улучшает стабильность работы и снижает вероятность проблем с сетью.

Дополнительно, вовлечение пользователей в процесс разработки стандартных конфигураций помогает создать рабочие станции, которые соответствуют их потребностям, повышая производительность и удовлетворённость сотрудников.

Таким образом, успешное управление рабочими станциями требует сочетания технической компетенции и организационных навыков. Следование рекомендациям, изложенным в главе, позволит не только повысить качество IT-обслуживания, но и упростить работу системного администратора.

**Глава 4. Серверы**

Серверы отличаются от рабочих станций тем, что их работа рассчитана на множество пользователей, а не на одного человека. Это делает надежность и бесперебойную работу основными приоритетами. Чтобы серверы выполняли свои задачи, важно минимизировать время восстановления, создать комфортное рабочее окружение и уделять особое внимание настройке конфигурации.

К серверам может подключаться большое количество пользователей — от сотен до миллионов. Это создает определенные ограничения на повышение их производительности или надежности, так как нагрузка растет пропорционально количеству подключений. Кроме того, серверы должны работать более продолжительное время, чем рабочие станции, что требует больших затрат. Однако покупка сервера с избыточной мощностью становится выгодной, так как это увеличивает срок его службы.

**4.1. Основы**

Серверное оборудование заметно отличается от того, что используется в рабочих станциях. Оно разрабатывается по другой экономической модели, имеет специальные функции и рассчитано на выполнение задач под высокой нагрузкой. Установка и поддержка серверов требуют особого подхода.

Как правило, серверы поставляются с дополнительными возможностями, такими как контракты на обслуживание, системы резервного копирования, предустановленные операционные системы и доступ для удаленного управления. Кроме того, их размещают в вычислительных центрах, где обеспечиваются такие условия, как контролируемый микроклимат и ограниченный доступ к оборудованию. Эти особенности важно учитывать при выборе серверного оборудования, чтобы оно соответствовало вашим задачам и требованиям.

### 4.1.1. Покупайте для серверов серверное оборудование

Системы, продаваемые как серверы, значительно отличаются от обычных настольных компьютеров, предназначенных для использования в качестве клиентов. Иногда пытаются сэкономить, покупая настольный компьютер и устанавливая на него серверное ПО. Это может быть временным решением, но для долгосрочных и крупных проектов это не лучший выбор, так как такие решения могут быть ненадежными.

Серверное оборудование дороже, но оно оправдывает свою цену благодаря дополнительным возможностям:

* **Расширяемость**: Серверы обычно имеют больше физического пространства для жестких дисков, слотов для карт расширения и процессоров. Также предусмотрены разъемы с высокой пропускной способностью для подключения специализированных периферийных устройств. Оборудование и ПО могут поддерживать кластеризацию, распределение нагрузки, автоматическое переключение на резервные мощности в случае отказа оборудования.
* **Производительность центральных процессоров**: Серверы часто оснащены несколькими процессорами и дополнительными возможностями, такими как упреждающая выборка данных и динамическое распределение ресурсов между процессорами. Более скоростные процессоры могут быть дорогими, но на сервере, который поддерживает множество пользователей, это может быть оправдано.
* **Высокопроизводительные системы ввода-вывода**: Серверы обычно имеют более мощные системы обмена данными, чем клиентские компьютеры. Это может включать использование жестких дисков с интерфейсами SCSI или FC-AL, высокоскоростных шин и сетевых интерфейсов.
* **Возможности модернизации**: Серверы часто можно модернизировать, а не просто заменять. Они имеют возможность добавлять или заменять процессоры без значительных аппаратных изменений.
* **Монтирование в стойку**: Серверы должны быть спроектированы для установки в стойки, чтобы эффективно использовать пространство. Они должны иметь прямоугольную форму и подходить для монтажа с учетом охлаждения и вентиляции. Все элементы, требующие обслуживания, должны быть доступны без необходимости извлечения сервера из стойки.
* **Не требуется доступ с боковых сторон**: При установке в стойку сервер должен быть проще в обслуживании, и для этого не должно быть необходимости в доступе к боковым стенкам. Все кабели и отсеки для приводов должны быть расположены сзади и спереди соответственно.
* **Дополнения для повышения надежности**: Многие серверы имеют дополнительные функции для повышения надежности, такие как дублированные источники питания, RAID, несколько сетевых карт и компоненты с поддержкой "горячей" замены.
* **Контракт на обслуживание**: Поставщики предлагают контракты на обслуживание с оговоренными гарантийными сроками и условиями замены запасных частей.
* **Альтернативные варианты управления**: Идеальные серверы поддерживают функции удаленного управления, такие как доступ через последовательный порт для диагностики и восстановления после сбоев. Некоторые серверы оснащены датчиками температуры и другими аппаратными средствами мониторинга.

Поставщики постоянно совершенствуют серверы, чтобы удовлетворить потребности бизнеса. Например, они адаптируют их для размещения в колокейшн-центрах, где важны компактность и эффективное управление для минимизации простоя.

### 4.1.2. Выбирайте поставщиков, известных надежностью продукции

Выбор надежных поставщиков крайне важен, поскольку это гарантирует высокое качество серверного оборудования. Некоторые поставщики экономят на компонентах, используя детали потребительского класса, в то время как другие используют компоненты, соответствующие военным стандартам MIL-SPEC.

Поставщики с многолетним опытом на рынке обычно предлагают более качественные серверы, а также дополнительные функции, которые могут быть доступны только при длительном присутствии на рынке. Неопытные поставщики могут предложить лишь базовую замену неисправных компонентов без предоставления качественного технического обслуживания.

Чтобы выбрать надежного поставщика, полезно узнать мнение других системных администраторов. Ресурсы сообщества, такие как SAGE (Гильдия системных администраторов) и LOPSA (Лига профессиональных системных администраторов), могут быть полезными для обмена опытом и рекомендациями.

Оборудование может быть однотипным (все от одного поставщика или из одной линейки) или разнотипным (от разных поставщиков или из разных линеек). Однотипное оборудование проще обслуживать, так как для него нужно меньше запасных частей и меньше времени на подготовку. Однако разнотипное оборудование также имеет свои преимущества: оно снижает зависимость от одного поставщика и обеспечивает конкуренцию между ними, что может привести к улучшению обслуживания.

### 4.1.3. Реальные расходы на серверное оборудование

Чтобы оценить полные расходы на серверы, нужно понимать, что входит в цену оборудования.

Большинство поставщиков предлагают три категории продукции: для дома, для бизнеса и серверное оборудование.

* **Домашняя серия**: Наименьшая цена, но компоненты могут часто меняться, чтобы удержать цену на минимальном уровне. Эти устройства имеют дополнения для игр и мультимедиа, но они не предназначены для долгосрочной работы.
* **Серия для бизнеса**: Более высокая цена, но устройства рассчитаны на долгосрочное использование. Эти устройства имеют более стабильные компоненты, которые не меняются так часто, что облегчает обслуживание. Гарантии на компоненты обеспечивают их доступность на протяжении нескольких лет.
* **Серверное оборудование**: Ориентировано на наилучшее соотношение цены и производительности. Например, файловые серверы оцениваются по производительности на единицу стоимости, что делает их более эффективными в долгосрочной перспективе, несмотря на более высокую начальную цену.

Серверное оборудование обычно дороже по нескольким причинам:

* Корпуса, удобные для обслуживания, могут быть дороже в производстве.
* Ограничения по расположению панелей и отсеков для компонентов требуют более сложной и дорогой конструкции.
* Однако более высокая начальная цена оправдана экономией на обслуживании в будущем, так как она снижает время простоя и облегчает ремонт (MTTR).

Не стоит считать, что серверы дороже настольных ПК. Разница в предназначении оборудования: серверы обслуживают множество пользователей, выполняют миллионы транзакций, и простой сервера может обходиться гораздо дороже, чем простой настольного ПК. Дополнительное оборудование для серверов, такое как компоненты с поддержкой "горячей" замены, значительно снижает время простоя, что оправдывает вложения.

Иногда покупка сервера "с запасом" может быть выгодной стратегией, так как это отложит необходимость в модернизации и увеличении производительности в будущем.

### 4.1.4. Контракты на обслуживание

Этот отрывок подробно объясняет важность выбора правильного контракта на обслуживание серверов и запасных компонентов для эффективного управления инфраструктурой. Приводятся различные сценарии, которые помогают выбрать оптимальные условия в зависимости от характеристик серверов и их роли в сети.

1. **Не критически важный сервер**: Для серверов, которые не являются критическими для бизнеса, можно выбрать контракт с более длинным сроком обслуживания (например, на следующий день).
2. **Большая группа идентичных серверов**: В случае большого количества однотипных серверов можно хранить запасные компоненты на месте и выполнить их замену самостоятельно.
3. **Постепенная модернизация**: Когда старые компоненты устаревают, их можно заменить новыми с обязательным приобретением запасных частей.
4. **Важные узлы**: Для ключевых серверов может быть полезно хранить запасные части, чтобы быстро заменить часто выходящие из строя компоненты, такие как жесткие диски и источники питания.
5. **Большое количество моделей от одного поставщика**: Если серверы от одного производителя занимают важную роль в сети, возможно, потребуется контракты с техникой и выделенным складом запасных частей.
6. **Критически важный узел**: Для серверов, на которых зависит весь бизнес, стоит рассмотреть более дорогие контракты с возможностью выделенного техника и дублирующих машин для замены.

Важно сбалансировать расходы на хранение запасных частей и сервисные контракты. Также рекомендуется регулярно пересматривать контракт, чтобы учесть новые серверы и исключить устаревшие компоненты.

При выборе подходящего контракта стоит учитывать как экономические факторы, так и оперативные потребности бизнеса, чтобы минимизировать время простоя и снизить возможные расходы на сервисное обслуживание.

4.1.5. **Обеспечение целостности данных**

Сервера часто хранят важные и уникальные данные, которые требуют надежной защиты от потерь и повреждений. Клиентские рабочие станции обычно серийные, с однотипными конфигурациями, и их данные часто хранятся на серверах, что устраняет необходимость в их резервном копировании. Однако на рабочих станциях часто сохраняются пользовательские данные или локально устанавливается программное обеспечение, что может усложнить восстановление системы в случае поломки. На Windows-платформах, например, невозможно избежать локального хранения данных, таких как настройки реестра или установленное ПО. UNIX-системы более защищены от локальных изменений, но требуют регулярного резервного копирования нескольких конфигурационных файлов.

**4.1.6. Размещение серверов в вычислительном центре**

Сервера должны размещаться в надежных условиях, включая обеспечение энергоснабжения, охлаждения и физической безопасности. Лучшей практикой является заранее резервировать место для серверов в вычислительном центре или другом специализированном помещении. Важно установить сервер в его конечное местоположение сразу после сборки, чтобы избежать проблем с перегрузкой и физической безопасностью. Даже если компания небольшая, она должна иметь выделенную комнату или шкаф с необходимым оборудованием для защиты серверов и обеспечения бесперебойного питания.

**4.1.7. Конфигурация клиент-серверной ОС**

Сервера не обязаны использовать ту же ОС, что и клиенты, хотя для правильного функционирования они должны поддерживать одинаковые протоколы. Например, веб-серверы часто работают под другой ОС, чем клиентские машины, и для них можно использовать минимальную ОС, которая выполняет только нужные функции. Однако, в некоторых случаях, на серверах устанавливается та же ОС, что и на клиентах, но с модификациями для работы с большим количеством процессов или для специфических задач, как это бывает в UNIX-сетях.

Настройка серверов может варьироваться в зависимости от задач, которые они выполняют. Например, в случае с сервером Solaris можно использовать бездисковые клиенты для загрузки пакетов с сервера, в то время как сервер с Red Hat Linux обычно требует минимальной установки, на основе которой устанавливаются дополнительные специализированные пакеты.

### 4.1.8. Обеспечьте удаленный доступ через консоль

В этом разделе описывается важность обеспечения удаленного доступа к консоли серверов, чтобы системные администраторы могли оперативно управлять и обслуживать серверы без необходимости находиться в серверной комнате. Рассматриваются различные технологии и устройства, обеспечивающие удаленный доступ, включая переключатели КВМ и IP-КВМ. Также упоминается использование последовательных портов и специализированных решений для доступа к консоли, а также способы управления системами через удаленные соединения, такие как VPN и SSH.

**Основные идеи:**

1. **Удаленный доступ:** Благодаря использованию переключателей КВМ и IP-КВМ, системные администраторы могут управлять несколькими серверами через одну клавиатуру, мышь и монитор, что позволяет значительно экономить пространство в серверной.
2. **Безопасность:** При удаленном доступе важно обеспечить надежную аутентификацию и защищенные каналы связи (например, с использованием SSH и одноразовых паролей), чтобы предотвратить несанкционированный доступ.
3. **Использование консольных портов:** Старые устройства подключались через последовательные порты, которые теперь все чаще используются для удаленного мониторинга и управления, а сообщения с этих портов могут быть полезны для диагностики.
4. **Эргономика и безопасность:** Рабочее пространство для администраторов должно быть в офисах, а не в серверных комнатах, так как последние часто небезопасны и неудобны для длительного нахождения.

Удаленный консольный доступ позволяет улучшить эффективность работы и уменьшить затраты, однако требует продуманной системы безопасности и учета ограничений, таких как невозможность физической замены оборудования удаленно.

### 4.1.9. Зеркалирование загрузочных дисков

В данном разделе обсуждается важность зеркалирования загрузочных дисков для обеспечения надежности серверов. Зеркалирование представляет собой создание копии загрузочного диска, чтобы в случае отказа одного из дисков система могла автоматически переключиться на второй работоспособный диск.

**Ключевые моменты:**

1. **Зеркалирование и RAID 1:** Зеркалирование дисков используется для защиты загрузочного диска. Это достигается с помощью технологии RAID 1, при которой данные записываются на два диска одновременно, что минимизирует риски простоя при сбоях.
2. **Производительность:** Чтение данных с зеркалированных дисков ускоряется, так как информация может считываться с двух дисков одновременно. Однако процесс записи несколько замедляется, так как данные записываются на два диска. В системах с кэшированием записи (например, в UNIX) эта проблема минимизируется.
3. **Простой и замена дисков:** С зеркалированием сбой одного диска не приводит к простоям, так как система может продолжать работу с зеркальной копией. Заменить неисправный диск можно во время работы системы без значительных последствий для работы сервера.
4. **Ограничения зеркалирования:** Зеркалирование защищает от сбоев оборудования, но не от программных или пользовательских ошибок. Например, если на основное зеркало внесены ошибочные изменения, они сразу же копируются на второй диск, и восстановить прежнее состояние можно только с помощью резервного копирования.
5. **Резервное копирование:** Несмотря на защиту зеркалированием, для полноценной защиты данных необходимо регулярно выполнять резервное копирование. Пример из реальной практики показывает, что в случае программных ошибок (например, неправильное использование указателя памяти), данные на зеркале могут быть повреждены, что приведет к долгим простоям и финансовым потерям.

### 4.2. Тонкости

В этом разделе обсуждаются методы повышения надежности серверов и удобства их обслуживания. Рассмотрены различные подходы, которые могут значительно снизить риски сбоев и упростить работу с серверами.

### 4.2.1.1. ****Одноцелевые серверы****

Одноцелевые серверы — это устройства, специально предназначенные для выполнения одной задачи, например, файловые серверы, веб-серверы, серверы электронной почты и другие. Они обеспечивают более высокую эффективность и надежность по сравнению с универсальными системами, так как полностью оптимизированы для своей задачи.

**Преимущества:**

* **Экономия времени:** Одноцелевые устройства позволяют системным администраторам экономить время, так как их настройка и поддержка проще, чем настройка универсальных серверов.
* **Качество:** Они обеспечивают более высокое качество работы по сравнению с универсальными решениями, поскольку каждое устройство специализировано.
* **Дополнительные возможности:** Одноцелевые серверы часто предлагают уникальные функции, которые могут быть недоступны в универсальных устройствах, например, настраиваемые снимки файловой системы.

### 4.2.1.2. ****Резервные блоки питания****

Блоки питания являются одним из компонентов, наиболее склонных к сбоям, после жестких дисков. Чтобы повысить надежность серверов, рекомендуется использовать **резервные блоки питания** с избыточностью n + 1. Это значит, что если один блок питания выйдет из строя, система продолжит работать благодаря резервному блоку.

**Ключевые моменты:**

* **Отдельные кабели питания:** Каждый блок питания должен быть подключен к отдельному кабелю. Это предотвращает проблемы, если один из кабелей выдернут или перегружен.
* **Резервные источники питания (UPS):** Для повышения надежности можно подключить каждый блок питания к отдельным источникам бесперебойного питания (UPS), что исключает риски отказа от одного UPS.

### 4.2.1.3. ****Полная избыточность, или n + 1****

**Избыточность n + 1** означает, что система спроектирована таким образом, что продолжает работать даже при сбое одного из компонентов. Например, RAID-массивы обеспечивают продолжение работы даже при выходе из строя одного из дисков.

**Полная избыточность** подразумевает наличие двух полностью идентичных систем, где одна работает, а вторая готова взять на себя нагрузку при сбое первой. Это может быть реализовано с автоматическим или ручным переключением.

**Преимущества:**

* **Экономия:** Когда n >= 2, избыточность n + 1 оказывается более выгодной с точки зрения стоимости, чем полная избыточность.
* **Использование в серверных подсистемах:** Избыточность n + 1 чаще используется для серверных подсистем, а не для всех компонентов системы, поскольку она более экономична.

4.2.1.4. **Компоненты, поддерживающие "горячую" замену**

"Горячая" замена компонентов означает возможность заменить неисправный элемент системы без ее отключения, что позволяет значительно уменьшить время простоя. Это особенно важно в случае сбоя компонентов, так как дает возможность оперативно восстановить работоспособность системы без необходимости ожидать запланированной перезагрузки.

Основное преимущество "горячей" замены заключается в том, что компоненты можно менять даже во время работы системы, что критически важно для поддержания ее функционирования в случае неисправности одного из элементов. Например, в RAID-системах используется дополнительный диск, который активируется автоматически при сбое другого диска, что позволяет системе продолжать работать в режиме n+1.

Однако стоит учитывать, что компоненты с "горячей" заменой могут значительно повысить стоимость системы. Это оправдано в случае, когда простои системы недопустимы, например, для критически важных приложений или в случае, когда технические перерывы можно проводить лишь один раз в год.

Перед приобретением компонентов с "горячей" заменой, важно задать поставщикам конкретные вопросы: какие компоненты поддерживают "горячую" замену, а какие нет, а также как будет происходить процесс замены и насколько длительным будет простой системы при ее выполнении.

**Пример:** Сетевой коммутатор, поддерживающий "горячую" замену, может требовать перезагрузки системы после установки нового компонента, что приведет к кратковременному простою, несмотря на его "горячее подключение". Важно внимательно ознакомиться с требованиями поставщика и протестировать компонент перед внедрением.

4.2.1.5. **Раздельные сети для административных функций**

Разделение сетевого трафика для административных задач (например, для резервного копирования и мониторинга) может существенно повысить надежность работы серверов. В частности, выделенная сеть для резервного копирования помогает избежать ситуации, когда операции с данными будут конкурировать за ресурсы с пользовательским трафиком.

Создание такой отдельной сети позволяет не только повысить пропускную способность, но и защитить административный доступ к серверам в случае проблем с основной сетью. Это дает возможность администраторам выполнять необходимые действия, даже если основная сеть временно выходит из строя.

Таким образом, создание отдельных сетей для административных функций позволяет минимизировать влияние проблем на основную сеть, улучшить производительность и повысить общую стабильность инфраструктуры.

### 4.2.2. Альтернатива: множество недорогих серверов

Этот раздел описывает стратегию использования недорогих серверов для создания масштабируемых кластерных систем, где предпочтение отдается количеству серверов вместо их высокой стоимости или производительности.

Главный акцент делается на том, что если система управляется правильно и сбои предсказуемы, то использование множества дешевых серверов (например, блейд-серверов или виртуальных серверов) может быть более выгодным и гибким решением, чем приобретение одного дорогого высокопроизводительного сервера.

Пример с веб-фермами показывает, как можно обрабатывать огромные объемы запросов, разделяя их между множеством серверов, что позволяет поддерживать работоспособность даже при сбоях. Однако существует и недостаток в том, что часто приходится управлять большим количеством машин, что требует правильной автоматизации и мониторинга.

Кроме того, в тексте подчеркивается, что серверное оборудование на базе серийных компонентов (например, для массового производства персональных компьютеров) стало доступным и эффективным для использования в таких системах, а виртуализация и технологии блейд-серверов позволяют значительно снизить затраты на физическое оборудование.

При этом важен и правильный подход к проектированию и управлению такими системами, чтобы избежать ситуации, когда процесс разработки и обслуживания становится неуправляемым из-за сложности взаимодействия с множеством отдельных серверов.

Этот подход идеально подходит для компаний, которым нужно быстро масштабировать инфраструктуру и эффективно управлять сотнями серверов, при этом экономя средства и ресурсы.

### 4.3. Заключение

В заключении подчеркивается, что при выборе серверов необходимо учитывать несколько ключевых факторов: надежность, безопасность, резервирование и масштабируемость. В отличие от персональных компьютеров, серверы должны быть высококачественными и соответствовать специфическим требованиям, которые позволяют обеспечивать бесперебойную работу множества пользователей и служб.

Основные моменты заключаются в следующем:

1. **Экономические и технические факторы**: При покупке серверов важно учитывать такие факторы, как стоимость, производительность, требования к охлаждению, электроснабжению и пропускной способности сети. Все эти элементы необходимо предусматривать заранее.
2. **Надежность и резервирование**: Для повышения надежности серверов часто используются системы с дополнительными компонентами, такими как зеркальные диски, резервные источники питания и прочие компоненты для предотвращения сбоев. Это также помогает сокращать время простоя и упрощает процесс восстановления после сбоев.
3. **Удаленное администрирование**: Серверы должны поддерживать удаленное администрирование, что дает возможность системным администраторам работать из удаленных точек, освобождая пространство в серверных и упрощая техническое обслуживание.
4. **Одноцелевые серверы**: Для определенных задач могут быть использованы одноцелевые серверы, которые сконфигурированы для выполнения конкретных функций. Эти серверы могут быть более надежными и простыми в управлении, но их настройка для нестандартных задач может быть сложной.
5. **Планирование и подготовка**: Прежде чем приобретать серверы, необходимо тщательно спланировать расположение оборудования, системы охлаждения и другие аспекты инфраструктуры, чтобы избежать проблем в будущем.

В итоге, грамотное решение этих вопросов на ранних этапах помогает обеспечить эффективную работу серверов и ускоряет процессы восстановления при сбоях, что критично для обеспечения бесперебойной работы всей системы.

### Глава 5

Глава 5 описывает ключевые аспекты предоставления и управления сервисами в системах, ориентированных на стабильность, масштабируемость и доступность. Основные идеи главы включают следующие моменты:

1. **Сервер и сервис**: Сервер — это оборудование, а сервис — это функция, предоставляемая сервером. Сервис может быть распределен на несколько серверов, которые работают совместно для обеспечения функционала.
2. **Основные обязанности системного администратора**: Один из главных аспектов работы системного администратора — обеспечение сервисов для пользователей, их настройка, поддержка и решение возникающих проблем. Эта задача требует внимания к деталям, поскольку от качества настройки сервисов зависит не только стабильность работы, но и удовлетворенность пользователей.
3. **Типы сервисов**: Существует множество типов сервисов, включая базовые (например, DNS, электронная почта, аутентификация, подключение к сети, печать), а также более специализированные сервисы для бизнеса, такие как резервное копирование, доступ к интернету, DHCP и другие. Все эти сервисы играют ключевую роль в функционировании как небольших, так и крупных корпоративных сетей.
4. **Надежность и масштабируемость сервисов**: Для того чтобы сервисы работали эффективно, они должны быть не только правильно настроены, но и обладать возможностью масштабирования, чтобы удовлетворить растущие потребности бизнеса. Надежность сервисов — еще один важный аспект, поскольку сбои в работе могут повлиять на многие процессы.
5. **Поддержка и обслуживание**: Обслуживание сервисов и мониторинг их состояния являются важными задачами, чтобы обеспечивать бесперебойную работу и своевременно реагировать на любые сбои.

### 5.1. Основы

Глава 5.1 описывает основные принципы создания исправных и надежных сервисов, которые должны отвечать множеству критериев, включая потребности пользователей, безопасность и простоту обслуживания. Вот ключевые моменты:

1. **Учет потребностей пользователей**: На всех этапах разработки сервиса системный администратор должен учитывать потребности пользователей. Важно выяснить, что пользователи ожидают от сервиса, чтобы правильно настроить его функционал.
2. **Зависимости сервисов**: Сервисы часто зависят от других сервисов, например, от DNS, электронной почты, аутентификации, сети. Это важно учитывать при проектировании, чтобы предотвратить каскадные сбои, когда отказ в одном сервисе может привести к отказу нескольких других.
3. **Надежность и безопасность**: Создание сервиса на серверном оборудовании должно гарантировать, что его надежность соответствует надежности всех сервисов, от которых он зависит. При этом важно ограничить доступ к серверу только для системных администраторов, чтобы минимизировать риски.
4. **Простота серверов**: Чем проще сервер и его конфигурация, тем выше его надежность и легче устранять проблемы. Серверы должны отвечать минимальным требованиям для работы сервиса, а доступ к ним должен быть ограничен только для обслуживающего персонала.
5. **Выбор оборудования и избыточность**: Системный администратор должен принимать решение о том, какое оборудование выбрать, использовать ли несколько серверов для одного сложного сервиса и какой уровень избыточности предусмотреть для обеспечения его бесперебойной работы.
6. **Упрощение обслуживания**: Использование стандартных конфигураций и оборудования, а также централизованный подход к обслуживанию сервисов (например, использование нескольких крупных серверов, а не множества мелких) может существенно упростить работу администраторов.
7. **Независимость от конкретных машин**: Важно использовать глобальные псевдонимы для сервисов, чтобы их можно было легко перемещать между машинами без необходимости перенастройки на клиентских устройствах. Это позволяет минимизировать время простоя и упростить управление сервисами.

Пример, приведенный в главе, объясняет, как привязка сервисов к именам машин может привести к проблемам при изменении инфраструктуры, тогда как использование псевдонимов для сервисов помогает избежать таких проблем.

### 5.1.1. Требования пользователей

Создание нового сервиса начинается с учета требований пользователей, ведь именно для них он разрабатывается. Без понимания нужд пользователей весь процесс создания может оказаться напрасным. Для некоторых сервисов, таких как DNS, требования пользователей почти отсутствуют, но для других, например, электронной почты, важен ряд специфичных функций и возможностей. Системные администраторы должны точно понимать, как пользователи будут взаимодействовать с сервисом, какие функции им нужны и какие ожидания у них по поводу работоспособности и поддержки.

При разработке почтовой системы, например, необходимо оценить, сколько писем система должна будет обработать в самые загруженные дни и сколько места потребуется для их хранения. Это помогает точно спланировать технические возможности и ресурсы, которые понадобятся для нормальной работы сервиса.

Важным элементом разработки является соглашение об уровне обслуживания (SLA). Этот документ описывает, какие сервисы и в каком объеме будут предоставляться, а также уровень их поддержки. В SLA также прописываются сроки отклика на разные категории проблем, а также процесс эскалации, если проблема не решается вовремя. Этот документ помогает не только системным администраторам, но и пользователям понять, какие возможности и ограничения существуют, и какие ресурсы необходимы для достижения требуемых целей.

Разработка SLA – это всегда совещательный процесс. Важно найти компромисс между желаниями пользователей, техническими возможностями и реальными ограничениями бюджета и времени. Системный администратор должен уметь объяснить пользователям, что технически невозможно или слишком дорого для реализации, и помочь найти оптимальное решение, которое удовлетворяет обе стороны. Важно понимать, что если пользователи просят что-то невозможное, это не повод для разочарования, а возможность для обсуждения и нахождения общего языка.

### 5.1.2. Эксплутационные требования

Системные администраторы должны учитывать целый ряд факторов при разработке новых сервисов, даже если пользователи о них не знают. Важно продумать административный интерфейс, который будет взаимодействовать с существующими сервисами и может быть интегрирован в центральные системы, такие как аутентификация или каталоги. Кроме того, администраторы должны обратить внимание на масштабируемость сервиса. Потребности пользователей могут расти, и сервис должен быть готов к увеличению нагрузки с учетом роста компании. Необходимо предусмотреть способы масштабирования, чтобы не прерывать работу сервиса.

Также важным вопросом является обновление сервисов. Нужно продумать, как будет происходить процесс обновления, будет ли он сопровождаться перерывами в работе, и можно ли провести обновление постепенно, например, протестировав его на нескольких добровольцах. Чем проще и меньше прерываний в процессе обновлений, тем лучше для работы сервиса.

Надежность системы — еще один важный аспект, который нужно продумать заранее. Системные администраторы должны учитывать возможные требования к кластеризации, избыточности серверов, а также запуску на оборудовании с высокой отказоустойчивостью. Важно учитывать и вопросы производительности, особенно для сетевой инфраструктуры, учитывая, что некоторые пользователи могут работать удаленно с медленным подключением.

Не менее важным является понимание проблемы с пропускной способностью и временем ожидания. Пропускная способность – это количество данных, которые могут быть переданы за единицу времени, а время ожидания – это задержка до того, как данные будут приняты другой стороной. Для приложений с высоким временем ожидания это может стать проблемой, особенно когда связь осуществляется через большие расстояния или через каналы с высоким временем задержки.

Увеличение пропускной способности может не решить проблему времени ожидания, поскольку это не снижает задержку, связанную с маршрутизацией и оборудованием. Одним из решений может быть доработка программного обеспечения с целью оптимизации запросов, чтобы минимизировать зависимость между запросами и ответами. Например, можно отправлять несколько запросов одновременно или изменять алгоритмы таким образом, чтобы их выполнение не зависело от ожидания ответа.

Системные администраторы должны также учитывать, как время ожидания влияет на работу сервиса и разрабатывать эффективные методы мониторинга быстродействия и работоспособности. Для этого может потребоваться интеграция нового сервиса с существующими системами мониторинга и регистрацией неисправностей. Все эти вопросы должны быть учтены и в бюджете проекта. Если ресурсных средств недостаточно для выполнения всех требований, это также должно быть частью обсуждения соглашения об уровне обслуживания. После утверждения соглашения, важно следить за тем, чтобы проект не выходил за рамки выделенного бюджета.

### 5.1.3. Открытая архитектура

Этот раздел подчеркивает важность использования открытых протоколов и форматов при создании сервисов, чтобы обеспечить их совместимость и гибкость, а также избежать зависимости от одного поставщика. Вот основные моменты, которые стоит отметить:

1. **Открытая архитектура**: Если сервисы строятся на открытых протоколах и форматах, они могут работать с другими решениями, использующими те же стандарты. Это делает их более универсальными и удобными для интеграции с другими сервисами.
2. **Проблемы с проприетарными решениями**: Когда используются закрытые (проприетарные) протоколы, это ограничивает возможность взаимодействия с другими сервисами, создавая зависимость от одного поставщика. Часто возникают проблемы с обновлениями и совместимостью между версиями разных продуктов.
3. **Протокол и продукт — это не одно и то же**: Протокол — это стандарт, который описывает, как должна передаваться информация (например, SMTP для электронной почты), а продукт — это конкретное программное обеспечение, которое использует этот протокол. Иногда поставщики стараются замаскировать различие между ними, создавая видимость совместимости с открытым стандартом, но на самом деле ограничивают взаимодействие с другими решениями.
4. **Преимущества открытых протоколов**: Использование открытых стандартов позволяет выбирать лучшие решения для клиента и сервера. Это дает возможность работать с оптимальными инструментами и упрощает администрирование. Открытые протоколы делают системы более гибкими, надежными и масштабируемыми.
5. **Опасности проприетарных систем**: Пример с фармацевтической компанией, которая выбрала проприетарную почтовую систему, показывает, как использование нестандартных решений может привести к проблемам с безопасностью, потерей данных и трудностями с масштабированием.
6. **Шлюзы как решение, но не панацея**: Для обеспечения взаимодействия между проприетарными системами и открытыми стандартами часто используются шлюзы. Однако они усложняют систему, делают её менее надежной и требуют дополнительных усилий по поддержке и масштабированию.
7. **Выбор правильных стандартов**: Открытые протоколы — это путь к более стабильной и совместимой системе. Они уменьшают зависимость от одного поставщика и обеспечивают долгосрочную поддержку и развитие сервисов.

В конечном итоге, использование открытых стандартов и протоколов помогает создать более гибкие и надежные сервисы, которые проще интегрировать и масштабировать, минимизируя риски и затраты на поддержку.

**5.1.4. Простота**

При разработке нового сервиса важно ставить на первое место простоту. Наиболее простое решение, которое удовлетворяет всем требованиям, будет самым надежным, удобным в обслуживании, легко расширяемым и интегрируемым с другими системами. Сложные решения приводят к путанице, ошибкам, проблемам с использованием и замедляют работу системы. Кроме того, такие системы требуют больше ресурсов для разработки и обслуживания.

С развитием системы её сложность неизбежно увеличивается. Поэтому, создавая её как можно более простой, можно отсрочить момент, когда она станет слишком сложной. Например, если у одного поставщика есть система с 20 основными функциями, а у другого — с 200 дополнительными, можно ожидать, что во втором случае будет больше ошибок, а поддержка системы окажется более трудной.

Иногда небольшие изменения в требованиях пользователей или системных администраторов могут значительно усложнить систему. Если на стадии разработки возникают такие требования, стоит вернуться и заново оценить их важность. Следует объяснить пользователям, что, хотя эти требования можно удовлетворить, это приведет к снижению надежности и уровня поддержки. Затем можно попросить их пересмотреть свои требования с учётом этого и принять решение о том, что важно, а что можно оставить.

**5.1.5. Отношения с поставщиком**

При выборе аппаратного и программного обеспечения для сервиса необходимо иметь возможность посоветоваться с техническими консультантами поставщиков для выбора оптимальной конфигурации. Поставщики часто предлагают готовые конфигурации, которые могут быть идеально подходящими для конкретных приложений, например для баз данных или веб-серверов. Если вы работаете с несколькими поставщиками, постарайтесь использовать это для получения максимальной выгоды. Пусть поставщики соревнуются за ваш заказ — это позволит вам получить больше функциональности за те же деньги или сэкономить средства для улучшения других аспектов сервиса.

При выборе поставщика важно понимать, в каком направлении он развивает свой продукт. Если вы крупный заказчик, вы можете участвовать в бета-тестировании и влиять на развитие продукта. В случае ключевых сервисов, таких как аутентификация или каталоги, важно следить за развитием платформы, чтобы избежать ситуации, когда поставщик прекращает поддержку вашей платформы. Старайтесь выбирать поставщиков, которые разрабатывают продукты для вашей платформы, а не тех, кто просто портирует своё ПО на неё.

**5.1.6. Независимость от конкретной машины**

Пользователи должны иметь доступ к сервису через стандартные имена, основанные на назначении сервиса, например, календарь через сервер с именем "calendar", почта — через "pop" или "imap". Даже если сервисы размещаются на одной машине, доступ к ним должен обеспечиваться через функциональные имена. Это позволит в будущем разделить сервисы на несколько машин без необходимости переписывать настройки клиентов.

Основное имя машины не должно совпадать с названием функции. Например, сервер для календаря может иметь имя "dopey", но доступ к нему будет через "calendar". Это важно, потому что если потребуется перенести функцию на другую машину, её имя будет оставаться прежним, и все клиенты смогут продолжать работать без изменений.

Если сервис привязан к IP-адресам, можно выделить несколько виртуальных IP-адресов для каждой функции, что облегчит перенос сервиса на другую машину. При проектировании сервиса важно заранее продумать, как будет происходить его перенос на другие машины, чтобы упростить этот процесс в будущем.

### 5.1.7. Среда окружения

Для обеспечения надежности сервиса важно иметь стабильную и эффективную среду окружения. Это включает в себя надежное энергоснабжение, хорошее охлаждение, контролируемую влажность, системы пожаротушения и другие меры безопасности, чтобы минимизировать вероятность повреждения оборудования. Важной частью инфраструктуры являются вычислительные центры, где размещаются сервера, обеспечивающие высокоскоростное подключение и надежность работы. Также важно, чтобы все компоненты сервиса находились в вычислительном центре, а не зависели от внешних машин, что минимизирует риск сбоя.

Пример с NFS-сервером, который завис от настольного компьютера, показывает важность этого принципа. Отказ внешнего сервера может привести к полной остановке работы системы.

### 5.1.8. Ограничение доступа

Ограничение доступа к серверам является важной мерой для обеспечения надежности. Доступ должен быть ограничен только для системных администраторов, чтобы предотвратить случайные сбои, вызванные пользователями, которые могут перегрузить сервер или случайно его перезагрузить. Например, когда пользователи начинают запускать ресурсоемкие приложения на сервере, это может привести к снижению производительности всего сервиса.

Рекомендуется с самого начала ограничить доступ к серверам и обеспечить только нужный минимальный доступ для управления системой.

### 5.1.9. Надежность

Надежность сервисов можно повысить, используя избыточные компоненты и инфраструктуру. Например, использование нескольких источников питания для серверов или их распределение по разным коммутаторам и сетям повышает устойчивость системы к сбоям. Также важно делать сервисы как можно более простыми, чтобы минимизировать количество зависимостей и точек отказа.

Например, если сервис зависит от нескольких компонентов, то сбой одного из них может привести к отказу всей системы. Вместо этого можно минимизировать такие зависимости, например, сделав компоненты зависимыми от одного источника питания или одного коммутатора.

Кроме того, важно учитывать возможные сбои сети и планировать архитектуру так, чтобы сервисы оставались доступными, даже если связь с основной сетью временно прервана.

Также стоит обратить внимание на возможность частичного отказа, при котором сервис продолжает работать, но с ограниченной функциональностью. Это может помочь уменьшить последствия полных сбоев и минимизировать их влияние на работу системы.

**Глава 5.1.10. Один сервер или несколько**

При разработке сервисов важно определить, будет ли сервис работать на одном сервере или распределён на нескольких. Основным фактором является уровень финансирования и количество персонала. В идеале сервисы и демоны должны располагаться на отдельных машинах. Однако если сервис взаимодействует с несколькими приложениями или демонами, важно решить, размещать ли все компоненты на одной машине или распределить их между несколькими.

Этот выбор зависит от факторов безопасности, производительности и масштабируемости. Например, веб-сайт с базой данных имеет смысл развернуть так, чтобы база данных была на отдельном сервере. Это обеспечит защиту от внешних угроз и даст возможность масштабировать сервис, подключив дополнительные веб-серверы, не затрагивая сервер с базой данных.

Если же один из компонентов предполагается использовать только в рамках одного приложения, но впоследствии может быть задействован другими приложениями, его лучше разместить на выделенном сервере. Это позволит избежать зависимостей, которые могут помешать обновлению сервиса в будущем.

Если два компонента тесно связаны и никогда не будут использоваться отдельно, имеет смысл разместить их на одной машине, чтобы снизить зависимость от нескольких серверов.

**Глава 5.1.11. Централизация и стандарты**

Централизация сервисов и приложений подразумевает их управление с одного центрального узла, что позволяет сократить дублирование работы и снизить затраты. Все сервисы должны быть стандартизированы, чтобы упростить обслуживание и обеспечить одинаковые условия для всех сотрудников, работающих с сервисами.

Службы поддержки должны быть хорошо знакомы с сервисами и использовать стандартное оборудование, чтобы быстро реагировать на запросы. Поддержка нескольких копий одного сервиса может создать дополнительные сложности, так как будет необходимо точно отслеживать, на каком сервере работает тот или иной компонент.

В крупных организациях некоторые сервисы, такие как электронная почта или аутентификация, должны быть централизованы. Для других сервисов, например, файловых серверов или вычислительных ферм, более рациональным будет их размещение на уровне отдельных подразделений.

**Глава 5.1.12. Производительность**

Производительность сервиса играет ключевую роль для пользователей. При разработке важно учитывать, насколько быстро работает сервис, и стремиться к тому, чтобы система не теряла актуальность в течение нескольких лет.

Для достижения высокой производительности необходимо понимать архитектуру сервиса и продумывать способы масштабирования. Например, тестирование нагрузки помогает определить, как система работает при растущем числе пользователей и запросов. Важно проводить такие тесты заранее, чтобы понимать, сколько ресурсов потребуется, когда сервис будет запущен в эксплуатацию.

Планирование мощностей – это важный аспект, который помогает избежать проблем в будущем, таких как перегрузка системы. Для этого следует использовать серверы с достаточным запасом мощностей, чтобы система могла справляться с пиковыми нагрузками.

**Глава 5.1.13. Мониторинг**

Мониторинг – это неотъемлемая часть любого сервиса. Он должен быть настроен с самого начала, чтобы оперативно выявлять проблемы и предотвращать их до того, как они затронут пользователей.

Для этого необходимо наладить систему оповещений, которая позволит своевременно уведомлять службу поддержки о возникающих проблемах. Также важно следить за производительностью системы и планировать её возможности роста. Эффективный мониторинг использования ресурсов поможет предсказать возможные проблемы и заранее их устранить.

**Глава 5.1.14. Разворачивание сервиса**

Процесс внедрения нового сервиса так же важен, как и его разработка. От того, как пройдет развертывание, зависит, как сервис будет воспринят пользователями.

Чтобы создать положительное первое впечатление, нужно заранее подготовить документацию, обучить службу поддержки и разработать процедуры для устранения проблем. Также необходимо протестировать механизмы установки и настройки программного обеспечения на всех рабочих местах.

Важно использовать поэтапный подход к внедрению, начиная с тестовых групп, что позволит минимизировать риски и обеспечить более стабильное внедрение. Удобство для пользователей и минимизация необходимости вмешательства в настройки клиентов также играют ключевую роль для успешного развертывания сервиса.

**Глава 5.2. Тонкости**

Помимо того, что создаваемый вами сервис должен быть надежным, отслеживаемым, простым в обслуживании и поддержке, соответствующим всем основным вашим требованиям и требованиям пользователей, необходимо учитывать и некоторые другие аспекты. Если это возможно, для каждого сервиса стоит использовать выделенные машины. Это значительно упрощает обслуживание и поддержку сервисов. В крупных компаниях использование выделенных машин — одна из основ. Для небольших компаний стоимость установки выделенных машин может быть чрезмерно высокой.

Еще один идеал, к которому стоит стремиться при создании сервисов, — добиться для них полной избыточности. Некоторые сервисы являются настолько важными, что полная избыточность для них — это необходимость, не зависящая от размера компании. По мере роста компании вы должны стремиться к полной избыточности и других сервисов.

**Глава 5.2.1. Выделенные машины**

В идеале сервис необходимо создавать на выделенных машинах. В крупных сетях такая структура может быть оправдана на основе требований к сервисам. Однако в небольших сетях целесообразность такого шага будет значительно менее очевидной. Наличие выделенных машин для каждого сервиса повышает надежность сервисов, упрощает отладку при возникновении каких-либо проблем с надежностью, снижает масштаб простоев, а также значительно упрощает модернизацию и планирование мощностей.

В растущих корпоративных сетях, как правило, выделяется одна центральная административная машина, являющаяся ядром всех критически важных сервисов. Она предоставляет сервисы имен, аутентификации, печати, электронной почты и т. д. В результате из-за увеличения нагрузки эту машину приходится разделять, а сервисы — распределять по нескольким серверам. Зачастую к тому времени, как системным администраторам удается добиться финансирования для дополнительных административных машин, эта машина уже настолько загружена сервисами и зависимостями, что разделить её очень сложно.

Сложнее всего при распределении сервисов с одной машины на несколько разобраться с зависимостями IP-адресов. Некоторые сервисы подразумевают жестко запрограммированные на всех пользователей IP-адреса. В конфигурации многих сетевых продуктов, таких как брандмауэры и маршрутизаторы, IP-адреса жестко запрограммированы в их конфигурации.

**Пример из практики: Разделение центральной машины**

Компания Synopsys, будучи еще небольшой, начинала с типичной конфигурации одной центральной административной машины. Она играла роль NIS-мастера, DNS-мастера, сервера времени, принт-сервера, консольного сервера, почтового сервера и других критически важных сервисов. Время от времени один из новых системных администраторов случайно вводил команду *halt* на центральном сервере, что приводило к остановке работы всей компании.

С ростом компании возникла необходимость разделения функциональности этой машины на несколько серверов. Например, основные NIS- и DNS-сервисы были перенесены на три машины с множеством сетевых интерфейсов, а другие сервисы — на дополнительные машины. Это улучшило стабильность работы и позволило снизить нагрузку на центральную машину.

**Глава 5.2.2. Полная избыточность**

Наличие дублирующего сервера или набора серверов, которые готовы принять на себя роль основных серверов в случае сбоя, называется полной избыточностью. В случае сбоя переход роли основного сервиса к вторичному может происходить несколькими способами: потребуется вмешательство человека, переход может быть автоматическим при сбое основного сервера или основной и вторичный серверы могут разделить нагрузку до отказа одного из них, после чего оставшийся сервер возьмет на себя всю рабочую нагрузку.

Тип используемой вами избыточности будет зависеть от самого сервиса. Некоторые сервисы, такие как веб-серверы и вычислительные фермы, прекрасно запускаются на крупных фермах клонированных машин. Другие сервисы, такие как огромные базы данных, требуют более тесно связанной системы преодоления отказа. Программное обеспечение, которое вы используете для предоставления сервиса, может потребовать избыточности в виде постоянно подключенного пассивного подчиненного сервера, который реагирует на запросы только в случае отказа основного сервера.

**Избыточность для критических сервисов**

Для сервисов имен и аутентификации полная избыточность создается в первую очередь, частично из-за того, что программное обеспечение разрабатывается для вторичных серверов, и частично из-за их критической важности. Для других критических сервисов, таких как электронная почта, печать и сеть, избыточность создается гораздо позже, так как полную избыточность для таких сервисов создать сложнее и дороже.

**Глава 5.2.3. Потоковый анализ для масштабирования**

Если вы представляете себе отдельные компоненты типичной транзакции в сервисе, вы сможете масштабировать сервис более точно и эффективно. Опыт Страты в создании масштабируемых интернет-сервисов для интернет- и ASP провайдеров позволил ей создать потоковую модель для отдельных транзакций и объединить их в электронные таблицы, чтобы получить общую потоковую картину. На самом деле все это намного проще, чем представляется на первый взгляд.

### Потоковая модель

Потоковая модель — это список транзакций и их зависимостей с информацией о ресурсах, которые используются в процессе каждой транзакции. Эта информация может включать в себя:

* объем памяти, используемой на сервере для транзакции;
* размер и количество пакетов, используемых в транзакции;
* количество открытых сокетов и другие параметры.

При моделировании транзакции важно учесть все детали, даже такие, как поиск интернет-имен через DNS, чтобы получить точную картину транзакции. Даже внешние аспекты, на которые нельзя повлиять, например, поведение корневых DNS-серверов, могут оказывать влияние на моделируемую транзакцию. Например, если узкое место транзакции связано с поиском имен, можно внедрить внутренний кэширующий сервер имен, что ускорит процесс.

### Важность зависимости от инфраструктуры

Транзакции сервисов почти всегда зависят от инфраструктуры. При исследовании проблем с масштабированием часто выясняется, что узкое место находится именно в инфраструктуре. Для точного анализа масштабируемости сервиса важно учитывать все детали, включая внешние зависимости, которые могут повлиять на производительность.

### Локализация узких мест

Когда потоковая модель точно отображает сервис, можно локализовать проблемы с производительностью и масштабируемостью. Например, если база данных способна обрабатывать 100 запросов в секунду, и каждый запрос к домашней странице требует три запроса к базе данных, то сайт будет работать только при нагрузке не более 33 переходов в секунду. Однако, если увеличить производительность базы данных до 200 запросов в секунду (например, добавив еще один сервер), сайт сможет обрабатывать в два раза больше переходов в секунду.

### Проблемы с серверными ресурсами

Ресурсы сервера тоже могут стать проблемой. Например, если сервер предоставляет доступ к электронной почте через протокол IMAP, каждому пользователю может требоваться определенное количество оперативной памяти. Для правильного анализа важно учитывать распределение нагрузки и использование ресурсов сервером.

### Пример анализа потока данных

Рассмотрим ситуацию, когда сеть не была перегружена, но наблюдались проблемы с медленным открытием файлов. В дальнейшем исследовании было установлено, что сервер каталогов был перегружен, потому что каждый пользователь генерировал множество запросов на открытие файлов. Когда были подсчитаны запросы каждого пользователя и количество одновременно обращающихся пользователей, стало ясно, что для нормальной работы сервиса необходимо добавить дополнительный сервер каталогов.

**Глава 5.3. Заключение**

Разработка и создание сервисов — важная часть работы каждого системного администратора. От того, насколько качественно выполнена эта работа, зависит, насколько легко будет обслуживать и поддерживать сервис, его производительность и удовлетворенность пользователей.

### Улучшение обслуживания пользователей

Главной целью создания сервисов является улучшение обслуживания пользователей. Системный администратор может создать выделенные серверы, упростить управление, вести мониторинг серверов и сервисов, следовать стандартам компании и централизовать сервисы на нескольких машинах.

### Прогнозирование потребностей

Для успешного масштабирования важно видеть далеко за пределами текущих требований. Нужно учитывать, какие компоненты сервиса будут важны для пользователей в будущем и какие ресурсы потребуются для их обслуживания.

### Обеспечение надежности

Сервисы должны быть настолько надежными, насколько это необходимо для пользователей. Для крупных компаний важно обеспечивать избыточность сервисов, чтобы при сбое одного компонента сервис продолжал работать.

### Плавное развертывание

Развертывание сервиса должно быть постепенным, чтобы минимизировать помехи в работе пользователей. Мнение пользователей о сервисе формируется в основном в процессе внедрения, поэтому важно провести его правильно, чтобы обеспечить максимальную удовлетворенность.

**Глава 6. Вычислительные центры**

Эта глава посвящена созданию вычислительного центра — места, в котором находятся машины, предоставляющие общие ресурсы. Однако вычислительный центр — не просто комната с серверами. Как правило, он оснащен системами охлаждения, регулировки влажности, электропитания и противопожарными системами. Все эти системы составляют неотъемлемую часть вашего вычислительного центра. В теории, вы должны собрать все важные компоненты в одном месте и следить, чтобы оно оставалось достаточно надежным.

**Глава 6.1. Основы создания вычислительного центра**

Создание вычислительного центра на первый взгляд может показаться простым: достаточно большого помещения с необходимой мебелью и оборудованием. Однако на практике процесс намного сложнее. Важнейшими аспектами являются выбор качественных стоек, правильное подключение сетевых кабелей, обеспечение питания, охлаждения и системы пожаротушения. Важно продумать защиту от стихийных бедствий, а также правильно организовать работу помещения с учетом безопасности и транспортировки оборудования. Каждый элемент должен быть тщательно спланирован, чтобы обеспечить эффективность работы и долгосрочную эксплуатацию центра.

**Глава 6.1.1. Размещение вычислительного центра**

Основным этапом при создании вычислительного центра является выбор места его расположения. Это решение зависит от множества факторов, включая географическое положение, особенности региона и возможные угрозы, такие как стихийные бедствия. Например, если вычислительный центр находится в зоне частых землетрясений или ураганов, необходимо заранее предусмотреть меры по защите, такие как использование сейсмостойких конструкций и резервных источников питания. Также стоит учитывать вероятность случайных повреждений инфраструктуры, например, экскаватором, и заранее готовиться к таким ситуациям.

В некоторых случаях важно учитывать политические и экономические факторы. Например, в Калифорнии, где в 2000-2001 годах возникли проблемы с энергоснабжением, компании, размещавшие свои вычислительные центры в этом регионе, пострадали от перебоев в поставках электроэнергии. Планирование и распределение вычислительных мощностей между несколькими регионами могут помочь избежать таких проблем.

После выбора города и региона необходимо приступить к выбору конкретного места внутри здания для размещения вычислительного центра. Важно учитывать такие параметры, как прочность пола, доступность для транспортировки оборудования и возможные риски, связанные с наводнениями, землетрясениями и другими природными катастрофами. В случае с наводнениями, например, следует избегать размещения вычислительного центра в подвале или на первом этаже. Также необходимо учитывать расположение вспомогательных систем, таких как источники бесперебойного питания и системы охлаждения, чтобы обеспечить бесперебойную работу в случае их отказа.

Если требуется максимальная защита, примером может служить военный бункер, использующийся для размещения вычислительных центров. Такой подход обеспечивает стойкость к природным катастрофам и атакам. Однако чрезмерная защита может привести к логистическим проблемам, как это случилось с компанией HavenCo, которая разместила вычислительный центр на морском форте, что привело к множеству трудностей, включая транспортировку оборудования и кадровые проблемы.

Для вычислительных центров, расположенных в сейсмически активных регионах, важно предусмотреть установку сейсмостойких конструкций, которые могут обеспечить стабильность оборудования в случае землетрясений. Стоит выбрать стойки, которые могут выдержать вибрации, и закрепить оборудование таким образом, чтобы оно не выпало во время сотрясений. Также следует обратить внимание на прокладку кабелей, чтобы они не подвергались повреждениям при растяжении или сжатии.

Если вычислительный центр находится в регионе, подверженном частым грозам, необходима защита от молний и перепадов напряжения. Использование специализированных грозовых защит может предотвратить повреждения оборудования и обеспечить бесперебойную работу центра.

Для крупных организаций, предоставляющих веб-сервисы, часто требуется развертывание множества избыточных вычислительных центров по всему миру. Это позволяет снизить риски перебоев в обслуживании и обеспечить устойчивость к отказам оборудования. В случае необходимости отключения одного центра для профилактических работ, избыточные мощности обеспечивают непрерывность работы системы.

**Глава 6.1.2. Доступ к вычислительному центру**

При проектировании вычислительного центра важно учитывать местные законы, которые могут требовать наличия нескольких выходов и пандусов для инвалидов. Также следует продумать логистику перемещения оборудования в центр, так как некоторые элементы могут превышать стандартные размеры дверных проемов. Необходимо предусмотреть удобные проходы между стойками, усилить полы в местах установки тяжёлого оборудования и обеспечить доступ к погрузочной платформе для доставки и транспортировки оборудования.

Создание вычислительного центра требует внимательного подхода к каждому элементу, начиная от выбора места и защиты от природных катастроф до организации доступа и транспортировки оборудования.

6.1.3. **Безопасность**

Безопасность вычислительного центра – это не только защита от внешних угроз, но и создание условий для безопасной и бесперебойной работы сотрудников. Основные принципы безопасности включают в себя физическую охрану и контроль доступа, а также соблюдение строгих процедур для предотвращения ошибок, которые могут привести к дорогостоящим сбоям.

1. **Ограничение доступа**  
   Доступ к вычислительному центру должен быть строго ограничен. Только те, кто связан с обслуживанием оборудования или работает в экстренных случаях, должны иметь право входа. Это важно для обеспечения бесперебойной работы и поддержания высоких стандартов для оборудования и кабелей. Системные администраторы должны следить за соблюдением всех правил обслуживания.
2. **Физическая защита**  
   Запирание на ключ – это не лучший вариант, так как такие ключи легко копируются. В современных вычислительных центрах рекомендуется использовать системы бесконтактных пропусков, которые автоматически регистрируют вход. В некоторых случаях используются дополнительные меры, такие как сочетание ключей и пропусков, а также системы с одновременным присутствием двух человек, чтобы предотвратить несанкционированный доступ.
3. **Биометрическая безопасность**  
   В некоторых высокозащищенных центрах используются биометрические замки. Однако существует этическая проблема, связанная с рисками: например, если для доступа используется сканирование отпечатков пальцев, то потеря пальца может привести к угрозам безопасности. Биометрические системы могут требовать дополнительных проверок, чтобы убедиться в жизнеспособности сотрудника, но даже в этом случае есть риски. Кроме того, такие системы не всегда удобны для людей с ограниченными физическими возможностями.
4. **Контроль доступа для посетителей**  
   Существуют также строгие правила для посетителей. Например, нельзя оставлять их без присмотра в вычислительном центре. Если привлекаются сторонние работники, то их необходимо тщательно контролировать, чтобы они не нарушили рабочие процессы или не случайно повредили оборудование.

6.1.4. **Электричество и охлаждение**

Энергоснабжение и охлаждение в вычислительном центре тесно связаны между собой. Оборудование потребляет электроэнергию, и при его работе выделяется тепло, которое необходимо эффективно отводить для предотвращения перегрева и сбоев.

1. **Энергозатраты и охлаждение**  
   За каждое потребляемое ваттное потребление электроэнергии необходимо потратить как минимум 1 ватт на охлаждение. Если оборудование потребляет 10 кВт, то система охлаждения должна быть на 10 кВт (в реальности немного больше из-за неэффективности систем). Это означает, что половина расходов на электроэнергию идет на охлаждение, а другая половина — на питание оборудования. Также важно отметить, что аппаратура с меньшим потреблением энергии будет экономить больше, так как для нее требуется меньше мощности для охлаждения.
2. **Методы охлаждения**
   * **Фальшпол**. Один из способов охлаждения — использование фальшпола для проведения холодного воздуха. Система вентиляции нагнетает холодный воздух через фальшпол, и он подается снизу на оборудование. Это позволяет вытягивать тепло вверх. Однако важно не размещать кабели под полом, так как они будут блокировать поток воздуха.
   * **Подвесная система**. Другой метод — подавать холодный воздух сверху, обдувая оборудование сверху вниз. Это требует дополнительной работы по нагнетанию воздуха вниз, но позволяет использовать фальшпол только для прокладки кабелей.
3. **Энергоснабжение и охлаждение на долгосрочную перспективу**  
   Планирование мощности для электроснабжения и охлаждения должно учитывать развитие оборудования, поскольку оно имеет тенденцию уменьшаться в габаритах, но требовать больше мощности и охлаждения по мере роста вычислительных мощностей. С увеличением популярности серверных ферм, старые стандарты становятся неприменимыми. Важно рассчитывать, что системы охлаждения и электропитания должны быть спроектированы так, чтобы удовлетворить растущие потребности в мощности на протяжении следующих 8–10 лет.
4. **Контроль влажности**  
   Влажность воздуха в вычислительном центре также играет ключевую роль. Высокая влажность может привести к образованию конденсата, что может вывести оборудование из строя, а низкая влажность вызывает статические разряды, опасные для техники. Идеальная влажность для вычислительных центров составляет от 45 до 55%.
5. **Стабилизация электропитания**  
   Электропитание должно быть стабилизированным, чтобы защитить оборудование от скачков и колебаний напряжения в сети. Для этого обычно используются источники бесперебойного питания (ИБП), которые обеспечивают стабильное напряжение на протяжении всего времени работы вычислительного центра. ИБП подает питание от батарей, которые подзаряжаются от основной линии питания, когда напряжение сети стабильно. Эти системы необходимы для защиты оборудования от перебоев в электроснабжении.

6.2. **Тонкости**

Для улучшения работы вычислительного центра помимо стандартных подходов можно внедрить дополнительные меры. Хотя некоторые из этих улучшений потребуют дополнительных затрат, они могут повысить эффективность и надежность центра, что важно в условиях коммерческой необходимости.

### 6.2.1. ****Повышенная избыточность****

Для достижения высокой надежности важно предусмотреть избыточность в системах электропитания и отопления, вентиляции и кондиционирования воздуха.

* **Системы отопления, вентиляции и кондиционирования**. Рекомендуется иметь две независимые системы, работающие параллельно. В случае отказа одной система второго типа автоматически возьмет на себя всю нагрузку. Каждая система должна быть способна охлаждать помещение в одиночку.
* **Система энергоснабжения**. Для защиты оборудования от отказов важно учитывать несколько факторов:
  + Размещение источников бесперебойного питания (ИБП), генераторов и автоматических вводных устройств (АВР). Нужны дополнительные ИБП и генераторы на случай отказа.
  + Важно предусмотреть механизм обхода для аварийных ситуаций (например, сгорание одного ИБП). Нужно разместить устройства для обхода в месте, где их легко обслуживать.
  + Рекомендуется подключать оборудование к нескольким источникам питания, например, переменному и постоянному току, или даже использовать разные источники переменного тока для разных пар машин.

Использование разных линий питания для устройств в одном корпусе позволяет минимизировать риск одновременного отказа нескольких блоков питания.

### Высоконадежные вычислительные центры

В индустрии телекоммуникаций разработаны стандарты, которые могут быть полезны при проектировании надежных вычислительных центров. Например:

* **NEBS (Network Equipment Building System)** — американский стандарт, используемый для оборудования в центральных офисах телефонных компаний.
* **ETSI (European Telecommunications Standards Institute)** — европейский стандарт, регулирующий требования к оборудованию и помещениям.

Эти стандарты охватывают все аспекты, включая температурный режим, влажность, пожарную безопасность, электробезопасность и защиту от электромагнитных помех. Соблюдение этих стандартов важно для обеспечения долговечности и надежности оборудования.

Кроме того, вычислительные центры должны соответствовать требованиям регулирующих органов, например, **SAS-70**, который применяется к компаниям, предоставляющим услуги через интернет. Этот стандарт регулирует обработку данных и обязательства по аудиту в таких компаниях.

### 6.2.2. ****Больше пространства****

Если позволяют площади, следует делать проходы в компьютерных залах шире, чем того требуют нормы безопасности. Это позволяет более удобно перемещать оборудование и снижает риск повреждения устройств при транспортировке. Широкие проходы также помогают избежать застревания кабелей и делают работу с оборудованием более комфортной. Примером может служить вычислительный центр Cray в Игане, где проходы были в три раза шире длины самого крупного оборудования.

Создание такого пространства — это инвестиция в будущее, поскольку оно предоставляет гибкость в размещении оборудования и позволяет улучшить атмосферу в центре, способствуя лучшему обслуживанию и более комфортной рабочей среде.

### 6.4. ****Заключение****

Проектирование и планирование вычислительного центра требует тщательного подхода ко множеству аспектов, которые будут влиять на его долгосрочную эксплуатацию. Чтобы обеспечить правильную работу центра на протяжении многих лет, важно с самого начала учесть все ключевые моменты.

1. **Энергоснабжение и охлаждение**. Эти системы являются критически важными для бесперебойной работы вычислительного центра. Их недостаточная мощность или неправильная настройка могут вызвать серьезные проблемы. Грамотно спроектированный центр обеспечит не только стабильное питание, но и эффективное охлаждение, что избавит от множества технических сбоев и поможет избежать дорогостоящих ремонтов.
2. **Организация пространства и кабелей**. Беспорядок в проводах и кабелях — частая проблема, с которой сталкиваются многие операторы. Правильное планирование размещения проводов и кабелей не только способствует безопасности, но и улучшает удобство работы, уменьшая время, затраченное на техническое обслуживание.
3. **Безопасность и доступ**. Важный аспект — это организация системы доступа в центр. Доступ к вычислительному центру должен быть строго контролируемым, так как там находится дорогостоящее оборудование. При этом важно, чтобы система безопасности не мешала тем, кто работает с оборудованием и регулярно его обслуживает.
4. **Создание удобной рабочей среды**. При проектировании можно внедрить недорогие, но эффективные решения для улучшения рабочей атмосферы. Например, удобные рабочие места, наличие необходимых инструментов и материалов под рукой, а также ярлыки на оборудовании и специально отведенные места для мобильных устройств. Эти небольшие детали могут значительно повысить производительность и снизить время на обслуживание.
5. **Простор и избыточность**. Если площадь и бюджет позволяют, лучше сделать вычислительный центр просторнее и предусмотреть избыточность в системах питания и кондиционирования. Эти дополнительные инвестиции окупятся, обеспечивая более высокую надежность и бесперебойную работу центра в будущем.

В результате правильное планирование и внимание к деталям на стадии проектирования вычислительного центра позволяет создать надежную, высокоэффективную и безопасную инфраструктуру. Чем тщательнее будет спроектирован центр, тем меньше проблем возникнет в его эксплуатации, что сделает его более устойчивым к внешним и внутренним воздействиям в будущем.