

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РЕЗЕРВНОГО КОПИРОВАНИЯ ВИРТУАЛЬНЫХ МАШИН

В.П. Елисеев, К.В. Кудряшов, М.В. Носиков

В статье разрабатывается программное обеспечение (ПО) для создания расписания запуска готовых сторонних скриптов, нацеленных на создание полных резервных копий виртуальных машин (ВМ), развёрнутых в инфраструктуре виртуальных машин гипервизоров семейства XenServer от компании Citrix Systems, Inc. [1]. Разработанное ПО позволит: системному администратору создавать расписание автоматического запуска задач резервного копирования из интуитивно-понятного графического веб-интерфейса, доступного с любой рабочей станции внутри локальной сети предприятия, при условии наличия установленного браузера для просмотра веб-страниц, отслеживать наличие свежих резервных копий всех виртуальных машин, оповещать администратора системы об окончании выполнения задачи резервного копирования на всех подчинённых ему гипервизорах и о сбоях при выполнении поставленных задач. ПО позволяет сократить время и вычислительные ресурсы при копировании и восстановлении виртуальных машин, развёрнутых в распределенной вычислительной сети объектов автоматизации и энергетических объектов.

Ключевые слова: удаленный объект, визуализация, контроль, синхронизация, энергетика, датчик, локальная сеть, резервная копия, виртуальная машина.

Введение

В настоящее время диапазон применения разнообразных автоматизированных устройств становится все более широким. Внедрение передовых технологий дает возможность создавать системы управления, имеющие новые функции и в зависимости от поведения объекта управления, могут предоставить пользователю (администратору) большой спектр программных модулей управления. Это особенно актуально при построении распределённых вычислительных сетей и развертывании на их базе комплекса виртуальных машин. Внедрение новейших технологий и методик в процесс разработки сервисного программного обеспечения (ПО), необходимого для управления данными комплексами, позволяет снизить нагрузку на вычислительные ресурсы системы в целом, а также уменьшить количество управляющих воздействий, необходимых для настройки сервисного ПО,

что позволяет снизить нагрузку на системных администраторов и повысить качество и скорость обслуживания системы управления.

Широкое распространение получили бесплатные интранет-технологии, позволяющие решить многие задачи сбора, обработки и представления данных. Сочетание данных технологий с традиционными информационными технологиями дает возможность получения высоких результатов, как в скорости выполнения вычислительных операций, так и в функциональном наполнении приложений.

При возникновении аварийных ситуаций, связанных в основном с выходом из строя аппаратной части системы и сбоями в системном программном обеспечении, выдвигают требования по использованию дополнительного специального программного обеспечения. Данное ПО необходимо для проведения архивирования критических частей основного системного ПО и их восстановления после аварийных ситуаций.

Теоретический анализ

Виртуализация берёт своё начало в 1960-х годах прошлого столетия, когда ведущие в то время компании в сфере информационных технологий (General Electric, Bell Labs, IBM), искали средства для расширения размеров оперативной памяти. Системное хранилище памяти и хранилище, предназначенное для пользовательских программ, были разделены. Данная разработка инженеров Университета Манчестера и была первым шагом по направлению к созданию уровня абстракции, который позже будет использоваться во всех технологиях виртуализации.

Сегодня виртуализация стала неотъемлемой частью инфраструктуры ИТ на большинстве предприятий. Администрирование гипервизора (программного комплекса, обеспечивающего выполнение нескольких операционных систем (ОС) на вычислительном ресурсе), который инсталлирован поверх мощной аппаратной платформы, с запущенными на нём множеством виртуальных машин, каждая из которых выполняет свою отдельную функцию (веб-сервер, почтовый сервер, контроллер домена, прокси и т.д.) намного удобнее администрирования той же мощной аппаратной платформы, поверх которой установлена обычная ОС (без возможности виртуализации), а вышеупомянутые роли сервера запущены в качестве служб.

Одним из основных преимуществ виртуализации является возможность, без остановки гостевой операционной системы произвести резервное копирование не только всей файловой системы, но и конфигурации виртуальной машины (ВМ). Под конфигурацией ВМ понимается количество оперативной памяти (ОП), объём виртуального жёсткого диска, количество используемых ядер процессора, используемый виртуальный сетевой адаптер и его MAC-адрес (Media Access Control – управление доступом к среде, – уникальный идентификатор, присваиваемый каждой единице активного оборудования или некоторым их интерфейсам в компьютерных сетях Ethernet). Все эти параметры хранятся в специальном xml файле конфигурации ВМ.

Основной концепцией гипервизора является домен. Доменом называется запущенная копия виртуальной машины. Если виртуальная машина перезагружается, то её домен завершается (в момент перезагрузки) и появляется новый домен. Более того, даже при миграции содержимое копируется из одного домена в другой домен. Таким образом, за время своей жизни практически все виртуальные машины оказываются по очереди в разных доменах. Xen оперирует только понятием домена, а понятие «виртуальной машины» появляется на уровне администрирования (прикладных программ, управляющих гипервизором). Структура информационной сети с использованием гипервизоров приведена на рис. 1.

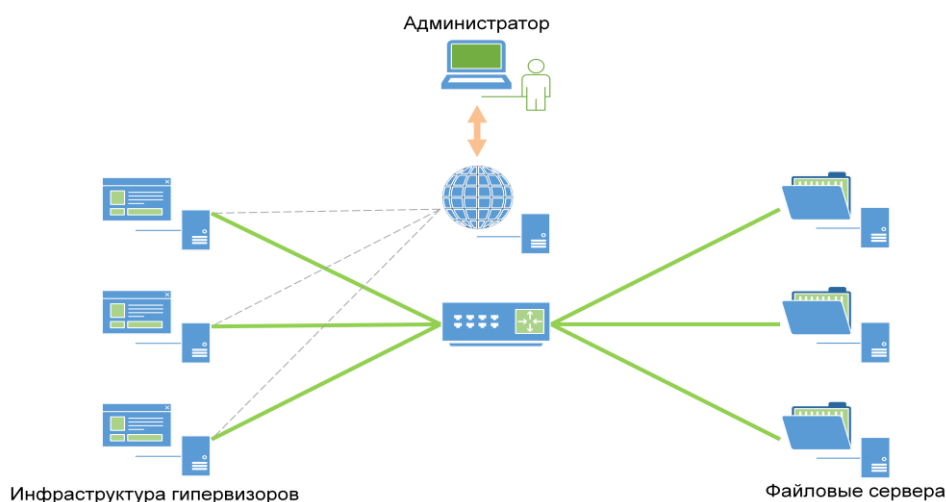


Рис. 1. Структура информационной сети с использованием гипервизоров

На рынке существует довольно широкий спектр различных программ для резервного копирования инфраструктуры ВМ, однако почти все эти программы имеют ряд недостатков:

1) высокая стоимость ввиду наличия расширенного набора функций, которые не всегда востребованы в повседневной деятельности системного администратора;

2) отсутствие кроссплатформенности (необходимо устанавливать данное ПО непосредственно на файловую систему ОС Windows, которая впоследствии и будет хранилищем для резервных копий);

3) закрытый исходный код;

4) ограничение на количество подключаемых хостов (гипервизоров);

5) зачастую сложный человеко-машинный интерфейс;

6) все резервные копии хранятся в собственном (закрытом) формате компании разработчика, их нельзя развернуть средствами с открытым исходным кодом, например, Xen API или программ для администрирования XenServer.

После анализа данных недостатков было принято решение по разработке методики построения ПО для резервного копирования виртуальных машин.

Структура программного обеспечения

Из вышесказанного можно сделать вывод, что практически во всех системах резервного копирования виртуальных машин необходимо применение прикладного программного обеспечения. В рамках реализации системы было принято решение использовать бесплатное программное обеспечение и/или программное обеспечение с открытым исходным кодом. Для реализации проекта использовались следующие программные компоненты:

- 1) серверная ОС FreeBSD;
- 2) система управления базами данных MySQL;
- 3) веб-сервер Apache;
- 4) командный процессор BASH;
- 5) интерфейс прикладного программирования XenAPI;
- 6) скриптовый язык программирования PHP.

Архитектура программного обеспечения представлена на рис. 2.

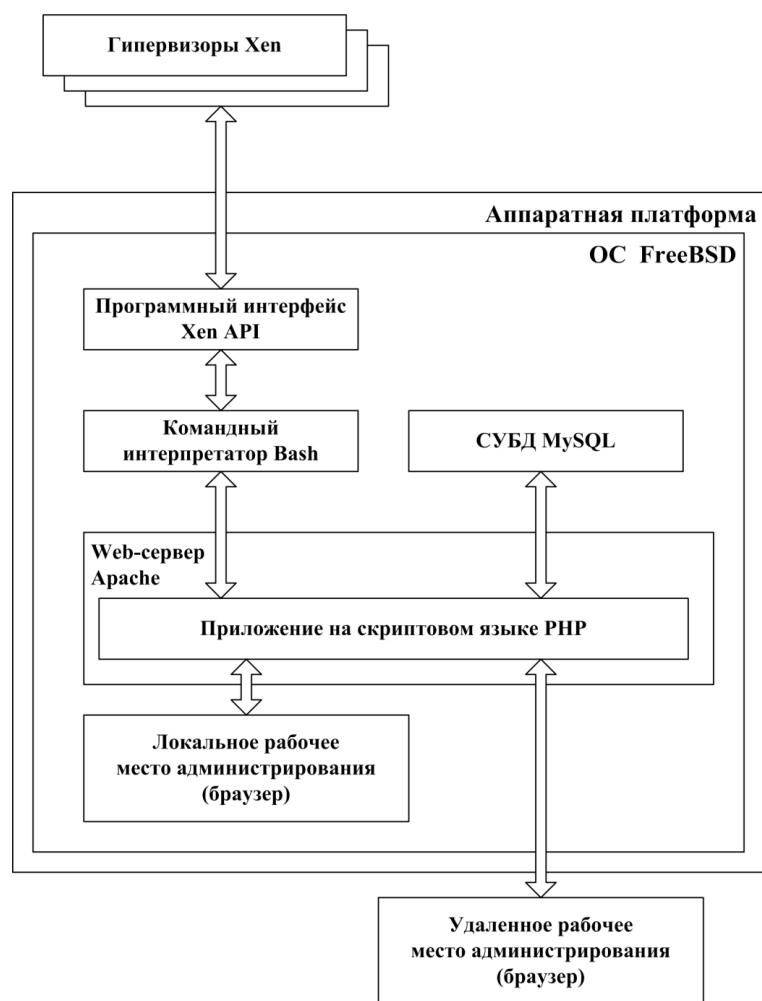


Рис. 2. Архитектура программного обеспечения резервного копирования VM

Разработка базы данных

Для разработки проекта было принято решение использовать систему управления базами данных (СУБД), основанную на реляционной модели данных. Термин «реляционный» означает, что теория основана на математическом понятии «отношение» (relation). Критериями выбора данной СУБД стали ее широкая распространенность, достаточно высокое быстродействие, наличие удобных средств администрирования СУБД.

Одним из средств администрирования и разработки структуры базы данных (БД) стал программный продукт phpMyAdmin.

Реляционная структура БД приведена на рис. 3. В состав БД входят основные и дополнительные таблицы данных и связи между ними.

Основное назначение БД (СУБД) в данном проекте – обеспечение хранения набора информационных записей о событиях и этапах формирования резервных копий виртуальных машин с одного или нескольких гипервизоров Xen. Данные этапы, помимо записываемых в файловую систему образов VM, сопровождаются дополнительной информацией, как, например, имя пользователя, время создания образа и т.д. Помимо этого, в зависимости от назначенного виртуальной машине функционала (например, почтового сервера), требуются дополнительные таблицы со структурой, специфичной для конкретного серверного функционала. В данной СУБД таковыми являются таблицы `mail_config`, `ntp_servers` (рис. 3).

Отдельным вопросом является организация хранения структурированных данных, оформленных в виде XML-структур. В настоящее время исходный формат XML и его производные являются достаточно популярными во многих приложениях сферы информационных технологий.

Подходы к хранению XML-документов в базе данных фактически являются подвидами следующих основных моделей хранения:

1. XMLType CLOB (Character Large Object). В некотором смысле, использование некоторого XMLType CLOB – это простейший способ хранения XML-файла. При этом документы в формате XML интерпретируются именно как документы. XML-файл сохраняется в структуре внешней памяти как полный текстовый документ (с пробелами (whitespace), комментариями и т.д.), при этом он запоминается просто как одна запись (строка символов) в базе данных. Следовательно, файлы любого размера и глубины (уровня вложенности, depth) могут быть сохранены, если они сформированы в формате XML. В данном подходе возможно определить типы данных (datatypes) в документе для проверки по схеме (базы данных), однако этими данными нельзя манипулировать или выбирать их с применением SQL-запросов. Из-за этого ограничения для поиска данных в документе нужно использовать механизм текстового поиска (text search engine), а не SQL-запросы, в которых можно воспользоваться функциональностью `query rewrites`, функциональными индексами и т.д. Эффективное изменение документа ограничено, так как оно требует выборки всего файла, совершения изменений и замены документа (в базе данных).

2. Представления типов данных XML. Альтернативой для CLOB XML Type является создание виртуального XML-документа «поверх» набора реляционных таблиц, известного как «XML view». Такой подход позволяет пользователю вставлять, изменять и уничтожать данные в XML-файле таким же образом, как если бы это были SQL-данные. Так как виртуальный XML-документ определяется «поверх» структуры хранения данных, то снимаются ограничения на представление данных только в одном (исходном) виде, и возможно создание («сборка») нескольких XML-документов исходя из задачи или требований пользователя.

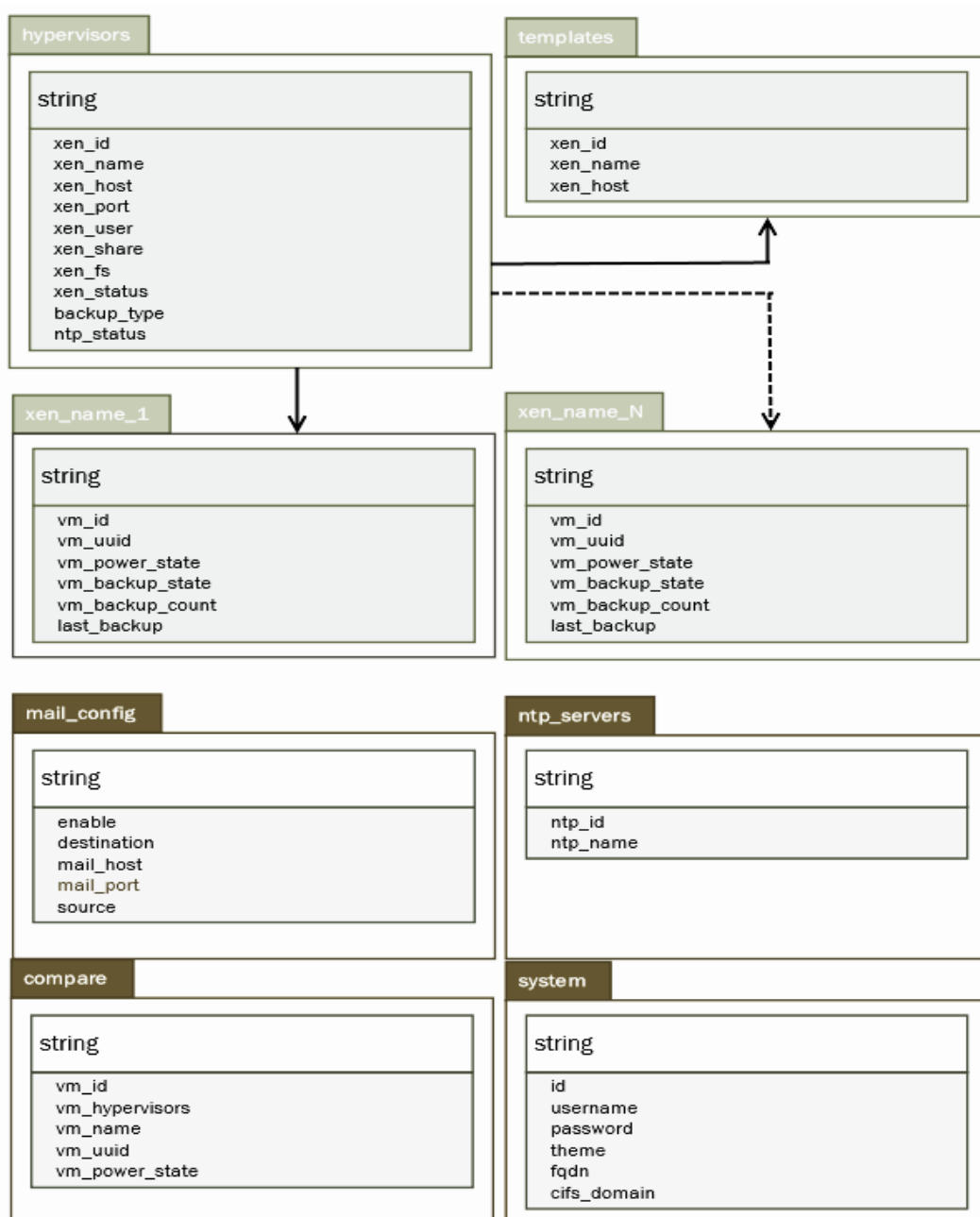


Рис. 3. Структурная схема таблиц данных системы резервного копирования VM

Большим преимуществом применения CLOB является полное сохранение структуры документа и гарантия его сохранности на уровне байтов. При использовании XMLType View теряется гарантия сохранения упорядоченности документа, так как многие элементы (такие как комментарии и инструкции по обработке) исчезнут при размещении (shredding) данных документа в таблицы (базы данных).

Практическая реализация системы резервного копирования

Разработанная система позволяет в интерактивном (диалоговом) режиме посредством стандартных Web-браузеров. Пользователь имеет возможность сформировать перечень используемых в сети Xen-гипервизоров, задать временную циклограмму (моменты времени) формирования резервных копий заданных виртуальных машин, критерии и условия их резервирования. При необходимости заданные (выбранные) копии виртуальных машин могут удаляться или передаваться в архив. Часть типовых операций может быть сформирована в виде шаблона и далее быть применена к назначенным виртуальным машинам.

Пример диалогового окна формирования шаблона резервного копирования представлен на рис. 4.

Шаблон для:

xenserver-1 ▼

Часы

19 ▼

Минуты

00 ▼

Дни месяца

Каждый день ▼

Месяц

Каждый месяц ▼

Дни недели

Пятница ▼

Синхронизировать время гипервизора с ArcticFox

ДОБАВИТЬ ➤

Рис. 4. Пример диалогового окна создания шаблона

Заключение

Разработанное ПО повышает надежность вычислительных систем в части быстрого восстановления работоспособности после сбоев или выхода из строя аппаратуры. Система уменьшает нагрузку на администраторов и повышает время бесперебойной работы вычислительной системы. Это актуально для вычислительных систем использующих сетевое взаимодействие и работающих на предприятиях ключевых отраслей (химическое производство, ядерная промышленность, предприятия ОПК и др.).

[К содержанию](#)