

Всероссийская научно-практическая конференция студентов, аспирантов и молодых ученых
«Современные технологии принятия решений в цифровой экономике»

СОСТАВЛЕНИЕ И РЕАЛИЗАЦИЯ ПЛАНА МИГРАЦИИ ПРИЛОЖЕНИЙ В ОБЛАКО

Разумников С.В., к.т.н.

Юргинский технологический институт (филиал) Национального исследовательского
Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26, тел. 8(38451)77764

E-mail: demolove7@inbox.ru

Аннотация. Любое крупное начинание должно проходить по плану. Разобравшись с портфелем приложений и выбрав их приоритеты, можно приступить к фактической миграции приложений в облако. В этой статье мы рассмотрим составление фактического плана, и его практическое применение.

Ключевые слова: план, стратегия, облачные технологии, миграция

Введение. При осуществлении перехода к облачным технологиям планирование и проектирование, основанные на выработанной стратегии, могут помочь сократить сроки развертывания и обеспечить успешную эксплуатацию сложных облачных моделей. На этом этапе требуется тщательная координация действий членов рабочей группы, партнеров и других поставщиков, а также детальная проработка архитектуры, компетентность в вопросах, касающихся центров обработки данных, и комплексный план обеспечения безопасности.

Полученные проектные решения и планы (в том числе, например, полномасштабный проект архитектуры, план осуществления перехода на новую платформу, общая концепция управления, концепция применения технологий безопасности, проект обеспечения физической защиты и безопасности, план будущего развития облачной инфраструктуры) должны быть привязаны к сформированной стратегии и закладывать основу для последующего внедрения и интеграции [1, 2].

Начало работы со средой разработки и тестирования. Прежде чем проанализировать проблему переноса рабочих приложений, стоит сначала подумать о перемещении среды разработки и тестирования. Вот почему это необходимо:

- *Среды разработки и тестирования дорогие.* Можно сэкономить значительное количество средств, переместив эти среды. Часто для каждого приложения есть три среды, полностью изолированные от рабочей: среда разработки, среда тестирования, используемая при проверке качества для выполнения модульных и автоматических тестов, и среда для пользовательского приемочного теста (UAT). Они могут использовать много серверов и виртуальных машин, которые больше не нужны в центре обработки данных.
- *Это прекрасный способ изучить инструменты миграции.* Группы могут ознакомиться со средствами миграции, не влияя на рабочую среду. Так что при переносе в рабочую среду используются преимущества, полученные при работе со средой разработки и тестирования.
- *Как правило, разработчики более устойчивы к ошибкам, чем пользователи.* Другими словами, при ошибке пользователи рабочей среды не пострадают, а разработчики и специалисты ИТ-служб смогут извлечь ценные уроки.
- *Порядок не так важен.* Рабочие приложения следует переносить в соответствии с рекомендациями по приоритетности. Это не так важно с приложениями в среде разработки и тестирования при условии, что правила и нормы соответствия для защищенных данных, например личных сведений, применяются как в среде разработки и тестирования, так и в рабочей среде (поэтому в среде разработки часто используются анонимные данные).

План миграции в облако. Разработка стратегии, позволяющей изменить текущее состояние на желаемое, – это основа плана миграции в облако. В плане миграции используется стратегия, к которой добавляются некоторые особенности, такие как приоритеты и последовательность. Приоритеты в плане нужно задавать с учетом разных бизнес-факторов, факторов, касающихся оборудования и программного обеспечения, а также других технических факторов. Чтобы составить согласованный список приоритетов, группа связи, операционная группа и другие связанные бизнес-подразделения должны работать сообща [3-7]. На рис. 1 показаны принципы, которые помогут составить рекомендации относительно приоритетов.

Для последовательной миграции рабочих нагрузок следует начать с простых проектов и после их переноса постепенно переходить к более сложным. Как и при работе с пилотным проектом, при переносе простых приложений с низкими бизнес-рисками вы получите ценный опыт, благодаря которому вы сможете подготовиться к миграции более сложных критически важных бизнес-приложений. План миграции в облако будет скорее динамичным, а не статичным документом. По сути, он будет состоять из целого ряда кратких планов, в соответствии с которыми выполняется ми-

грация рабочих нагрузок каждого отдела в заданной последовательности. Как правило, каждая миграция выполняется по следующему шаблону.

1. *Анализ.* Вы сможете определить, чего вам не хватает для переноса рабочей нагрузки в облако. Для этого может понадобиться изменить архитектуру рабочей нагрузки или полностью переписать программу. Кроме того, чтобы повысить производительность и масштабируемость многих устаревших программ, потребуются значительные усилия. При анализе рабочей нагрузки нужно определить соответствующие задачи.

В рамках анализа нужно полностью разобраться с требованиями к безопасности и соответствию оцениваемого приложения, чтобы точно определить любые личные данные и другую конфиденциальную информацию. Если такая информация присутствует, внедрите надлежащие средства контроля. Для этого обратитесь в службу ИБ и отдел управления рисками своей компании.



Рис. 1. Пример стратегии установки приоритетов при миграции

2. *Миграция приложений.* Если необходимо переместить определенную рабочую нагрузку в облако, лучше всего создать ее версию с минимальным объемом данных, чтобы приложение смогло работать или чтобы создать версию приложения в облаке. Если приложение уже выполняется на виртуальной машине, можно просто переместить ее в облако без дополнительных изменений. В целом многие локальные приложения можно выполнять в Microsoft Azure с минимальными изменениями или без них, но это не означает, что приложение будет оптимизировано для обеспечения определенного уровня производительности, масштабируемости и безопасности. Возможно, придется в определенной степени реконструировать приложение с применением современных сервисно-ориентированных принципов.

3. *Сетевые подключения.* Существует много способов безопасного подключения локальных приложений к облаку. Они варьируются от простых виртуальных частных сетей (VPN) до выделенных каналов с различной производительностью и ценами.

4. *Миграция данных.* Этот процесс похож на миграцию приложений тем, что данные можно переместить без изменения их структуры в реляционное или нереляционное расположение в облаке. Некоторые из этих видов миграции чрезвычайно просты. Их можно выполнить с помощью мастера, например мастера миграции SQL Server Azure. Однако можно рассмотреть вопрос о перепроектировании модели данных в качестве новой базы данных SQL Azure, чтобы улучшить производительность, масштабируемость, устойчивость и безопасность. Если нужно синхронизировать данные между локальной средой и базой данных SQL или между различными серверами базы данных SQL, нужно установить и настроить службу синхронизации данных SQL. Кроме того, рекомендуется составить и настроить план восстановления данных на случай ошибки пользователя или стихийных бедствий.

5. *Оптимизация и тестирование.* Переместив приложение и данные в Azure, протестировать функциональность и производительность. На этом этапе протестировать приложение в облаке и убедиться, что оно работает надлежащим образом. Затем сравните показатели производительности в локальной среде и в Azure. После этого устраните все проблемы с функциями, работой, производительностью или масштабируемостью в облачном приложении.

6. *Эксплуатация и управление.* После тестирования и оптимизации настроить и внедрить мониторинг и трассировку, используя службу Azure Application Insights, с помощью которой можно собирать с приложения телеметрию и анализировать ее. Эти данные можно использовать для отладки и устранения неполадок, измерения производительности, мониторинга использования ресурсов, анализа трафика, планирования емкости и аудита.

Каждая рабочая нагрузка, которую необходимо переместить, пройдет эти шесть этапов миграции. Однако также есть итеративный процесс, объемнее, чем любая миграция, в течение которого можно начать перенос приложений, отвечающих первоначальному минимальным стандартам, с учетом приоритетов и последовательности. Переместив первоначальную группу, можно приступить к обеспечению требований для приложений и оборудования. Для этого следует обновить версии операционной системы и SQL, установить актуальные обновления для системы безопасности, перенести приложения физических компьютеров на виртуальные машины и устранить проблемы при использовании нескольких IP-адресов и т. д.

Управление подписками. Начав работать с приложениями в облаке, следует рассмотреть вопрос управления подписками. Сначала может показаться оптимальным вариантом использовать одну облачную подписку для всех пользователей на предприятии, но в такой модели есть существенный недостаток: сложно управлять приложениями и/или центром или отделом учета затрат в модели, предусматривающей одну подписку. Кроме того, управление большим количеством приложений (в рабочей среде, среде тестирования и т. д.) может отнимать много ресурсов. И наконец, администратор одной подписки может получать слишком много новых запросов на виртуальные машины и другие ресурсы.

Обычно эффективнее назначать подписки отдельным центрам учета затрат, приложениям или группам приложений (например, приложения для продаж). При этом анализ затрат по подразделениям точнее, а ИТ-директора могут назначить каждой группе целевые показатели по издержкам, которыми они могут независимо управлять.

Чтобы добиться оптимальной прозрачности и отчетности в крупной организации, может потребоваться создать иерархию управления облаком, как показано на рис. 2.



Рис. 2. Иерархия управления подписками

В этой модели используется единый корпоративный портал, на котором можно просматривать расходы по всему предприятию. Учетные записи на уровне отделов могут содержать одну или несколько подписок для отдельных центров учета затрат или областей решения.

Заключение. Планирование и проектирование является вторым этапом перехода к облачным технологиям после формирования стратегии. В этой статье было рассмотрено составление фактического плана миграции приложений в облачную среду, и его практическое применение.

Список литературы:

1. Холодков А. ИТ-стратегия, часть 1: общий стратегический процесс в организации // ИТ-консультант.рф – [Электронный ресурс]. – Режим доступа: <http://www.kholodkov.ru/it/?p=671>. Дата обращения: 10.09.2015.

2. Михайлов А. Семь подходов к разработке ИТ-стратегий // Директор информационной службы, № 2, 2004 с. 48-52.
3. Разумников С.В. Интегральная модель оценки эффективности и рисков облачных ИТ-сервисов для внедрения на предприятии // Фундаментальные исследования. - 2015 - №. 2-24. - С. 5362-5366.
4. Разумников С. В. Моделирование оценки рисков при использовании облачных ИТ-сервисов // Фундаментальные исследования. - 2014 - №. 5-1. - С. 39-43.
5. Разумников С.В. Модель поддержки принятия решений о миграции корпоративных приложений в облачную среду // Научные труды Вольного экономического общества России. - 2015 - Т. 194. - С. 490-502.
6. Разумников С.В. Методика поддержки принятия решений при выборе облачных ИТ-сервисов для внедрения на предприятии // Труды Вольно экономического общества России. – 2018. – Том 212. – С. 339-362.
7. Разумников С.В., Фисоченко О.Н., Лунегов В.Ю. Информационная система оценки возможности корпоративных ИТ-приложений для миграции в облачную среду [Электронный ресурс] // Современные проблемы науки и образования. - 2014 - №. 4. - С. - Режим доступа: <http://www.science-education.ru/118-13924>.

СИСТЕМА РЕШЕНИЯ ОБРАТНОЙ ЗАДАЧИ ФОРМИРОВАНИЯ МАРЖИНАЛЬНОЙ ПРИБЫЛИ ПРЕДПРИЯТИЯ

Е.Б. Грибанова, к.т.н., И.Н. Логвин, студент

Томский государственный университет систем управления и радиоэлектроники

634050, г. Томск, пр. Ленина 40, тел.(3822) -70-15-36

E-mail: igor31821788@gmail.com

Аннотация. Рассмотрено решение обратных задач экономического анализа с помощью обратных вычислений в случае наличия зависимости между аргументами функции. Предложен алгоритм поиска решения обратной задачи при стохастической зависимости между аргументами, включающий решение задачи оптимизации. Приведено описание программы для решения задачи формирования маржинальной прибыли предприятия.

Понятие обратной задачи было впервые использовано Тихоновым А.Н., который определил содержание таких задач как восстановление значений неизвестных величин по заданному следствию [1]. Одиновым Б.Е. [2–3] был предложен аппарат обратных вычислений для решения следующих обратных задач экономики: определение приращений аргументов функции на основе начальных значений аргументов, заданного значения функции и экспертной информации, в качестве которой выступает направление изменения показателей и коэффициенты относительной важности [4].

Для функции двух аргументов ($y = f(x_1, x_2)$) решение обратной задачи с помощью обратных вычислений заключается в решении системы:

$$\begin{cases} y \pm \Delta y = f(x_1 \pm \Delta x_1(\alpha), x_2 \pm \Delta x_2(\beta)); \\ \frac{\Delta x_1}{\Delta x_2} = \frac{\alpha}{\beta}; \\ \alpha + \beta = 1, \end{cases}$$

где y - исходное значение функции;

x_1, x_2 – исходные значения аргументов;

$\Delta x_1, \Delta x_2$ – приращения аргументов;

α, β – коэффициенты приоритетности аргументов x_1, x_2 соответственно.

Знак перед приращением указывает направление изменения показателя: увеличение («+»), уменьшение («-»).

Решение задачи позволяет определить, как можно добиться желаемого показателя деятельности экономического объекта, полученная информация может быть использована для принятия управленческих решений. Изменяя экспертную информацию можно рассматривать различные варианты достижения цели.

В случае наличия стохастической зависимости между аргументами функции необходимо осуществить модификацию аппарата обратных вычислений. Для выполнения задачи выполняется построение регрессионного уравнения между аргументами: собираются статистические данные за предыдущие периоды и осуществляется определение параметров регрессии методом наименьших квадратов.