



Программное обеспечение IBM Rational

Методология и инструментальные средства
разработки программных систем

Вступительное слово

Программное обеспечение IBM Rational

Данная брошюра посвящена обзору методологии и программного обеспечения IBM Rational, которое помогает предприятиям автоматизировать и интегрировать бизнес-процесс по разработке программного обеспечения.

Продукты, услуги и лучшие методы организации работ Rational лежат в основе IBM Software Development Platform – платформы для разработки и развертывания программного обеспечения в рамках бизнес-приложений, встроенных систем и программных продуктов. Это модульное и функционально законченное решение позволяет командам разработчиков использовать подход к разработке, учитывающий интересы бизнеса и основанный на открытых стандартах, включая оболочку Eclipse. Результатом этого подхода является исключительная эффективность бизнеса.

Продукты IBM Rational помогают организовать эффективную командную работу и привести в соответствие с требованиями современных стандартов все процессы, связанные с разработкой, внедрением и сопровождением программного обеспечения: управление проектами, бизнес-моделирование, анализ и проектирование, управление требованиями и изменениями, тестирование и конфигурационное управление. Применение методов управления ИТ-проектами с использованием методологии и инструментария IBM Rational обеспечивает сокращение совокупных затрат на разработку и сопровождение ПО, повышает показатель эффективности внедрения информационных технологий. Девяносто восемь компаний из списка Fortune 100 используют инструментальные средства Rational для ускоренного создания более качественного программного обеспечения.

Продукты Rational в составе программных решений IBM

После приобретения IBM весной 2003 года компании Rational Software, технологии Rational стали неотъемлемой частью портфеля программного обеспечения IBM. «Rational является неотъемлемым элементом нашей недавно объявленной инициативы в области электронного бизнеса по требованию, — говорит Стив Миллз (Steve Mills), вице-президент IBM и руководитель подразделения IBM Software. — Законченная, открытая программная платформа Rational позволит повысить скорость, качество и прогнозируемость проектов в области программного обеспечения. Продукты Rational станут великолепным дополнением к имеющимся у нас четырем семействам ПО — WebSphere, DB2, Lotus и Tivoli». Программные инструменты компании используют более 600 тысяч разработчиков программного обеспечения.

Компания Rational Software была основана Полом Леви (Paul Levy) и Майком Девлином (Mike Devlin) в 1981 г. Их вдохновили перспективы позитивного влияния программного обеспечения на мировую экономику. С тех пор постоянно растет роль корпоративного ПО как двигателя мировой экономики и ключевого элемента в конкурентной борьбе компаний, сервисов и продуктов. Компания Rational Software была основана с конкретной миссией, которая остается неизменной до сих пор — обеспечение успеха клиентов, разрабатывающих и развивающих программное обеспечение. Цель Rational Software — помочь клиентам в создании информационных систем. Подход компании Rational Software помогает решать проблемы разработки, развития, тестирования и управления разработкой, позволяя создавать корпоративное ПО более быстро, качественно, надежно и с наименьшим риском.

Свой первый продукт компания Rational Software поставила на рынок в конце 1984 г. С тех пор она выросла в мощную, высокотехнологичную компанию. К моменту слияния Rational Software была крупнейшим специализированным поставщиком средств разработки ПО. По итогам ее последнего финансового года (31.03.2002) объем доходов компании составил 689 млн. долл. На момент вхождения Rational в IBM в компании работало около 3400 сотрудников, действовали 80 офисов по всему миру, ее продуктами пользовались более 600 тысяч разработчиков. С этих пор продукты корпорации распространяются под торговой маркой IBM Rational (www.rational.com), а бывший руководитель и один из основателей Rational Software Майк Девлин встал во главе нового подразделения IBM.

IBM Rational Suite в свое время стала первым интегрированным решением для разработки программного обеспечения. Теперь идеи коллективной работы получили дальнейшее развитие на уровне всей компании, а IBM Software Development Platform стала первой интегрированной платформой, которая позволяет увязать задачи разработки ИТ с целями бизнеса.

Партнеры Rational помогают расширить диапазон предлагаемых опций

Критически важные для бизнеса приложения должны предоставляться по требованию и максимально оперативно. Вот почему сотни партнеров предлагают продукты семейства Rational и помогают расширить предлагаемые опции разработки и развертывания программных систем. Вложения в технологии Rational станут пропуском в сообщество, состоящее из более чем 600 партнеров, которые занимаются ИТ-аутсорсингом, разработкой технологий и консалтингом.

Содержание

IBM Software Development Platform – новый подход к разработке программного обеспечения	5
<i>Программное обеспечение как бизнес-процесс</i>	5
<i>Портфель решений, основанный на ролях</i>	7
<i>Техническая основа IBM Software Development Platform</i>	8
Eclipse	8
Среда моделирования Eclipse Modeling Framework	11
Открытая платформа MDD, построенная на основе Eclipse	11
UML2	13
Hyades	13
Модель программирования WebSphere	13
Современный подход к созданию сложных программных систем	14
<i>Основные принципы организации работы над проектом</i>	14
Итеративная разработка	15
Эффективное управление требованиями	17
Компонентный подход	18
Визуальное проектирование системы	18
Гарантия качества продуктов	19
Контроль изменений	19
<i>Методология Rational Unified Process</i>	20
RUP как методология	20
Структура жизненного цикла проекта	21
Важнейшие акценты RUP	22
Программные решения IBM Rational	23
<i>Инструментальные средства IBM Rational</i>	23
Средства по управлению проектами и портфелями	27
IBM Rational Portfolio Manager	28
Моделирование и проектирование	30
IBM Rational Software Architect (RSA)	32
IBM Rational Software Modeler (RSM)	35
IBM Rational Rose	38
IBM Rational XDE Modeler	40
IBM WebSphere Business Integration Modeler (WBI Modeler)	41
Средства разработки	43
IBM Rational Application Developer	43
IBM Rational Web Developer	48
IBM Rational XDE Developer	52
Тестирование	52
IBM Rational Performance Tester	54

IBM Rational Functional Tester	56
IBM Rational Manual Tester	58
IBM Rational XDE Tester	60
IBM Rational Robot	62
IBM Rational Purify	63
IBM Rational Quantify	65
Rational PureCoverage	66
TestFactory	67
Организация совместной работы на основе IBM Team Unifying Platform	67
Управление процессом разработки. IBM Rational Unified Process.	68
Управление требованиями. Rational RequisitePro	70
Управление конфигурациями и изменениями. IBM Rational ClearCase,	
IBM Rational ClearQuest	73
Rational ClearCase	73
Rational ClearQuest	75
Управление тестированием. IBM Rational TestManager	77
Управление выполнением проекта и документированием. IBM Rational ProjectConsole,	
IBM Rational SoDA	78
IBM Rational Project Console	78
IBM Rational SoDA	79
Примеры внедрения проектов на базе технологий IBM Rational	80
Swedbank повышает эффективность создания ИТ-приложений, ориентированных	
на потребности клиентов и цели бизнеса, с помощью разработки на основе моделей	80
Внедрение системы управления конфигурациями и изменениями повышает	
эффективность сопровождения банковских систем	82
Внедрение Unified Change Management в компании Protek поднимает	
управления проектами на качественно новый уровень	84
Business Objects повышает эффективность разработок с помощью	
IBM Rational ClearQuest	86
VIVEO ToolObject создает технологически независимую бизнес-модель для	
банковской индустрии на основе UML и средств моделирования IBM Rational	88
Эффективная организация процессов тестирования на базе методологии RUP	
и средств тестирования IBM Rational	90
Полезные источники информации и сервисы	91
Услуги IBM Rational	91
Полезные ссылки в Интернет	91
Рекомендуемые книги	93

IBM Software Development Platform – новый подход к разработке программного обеспечения

IBM Software Development Platform - платформа IBM для разработки программного обеспечения - представляет собой реализацию новой концепции роли ИТ, согласно которой разработка информационных систем является важной частью бизнеса любой организации, которая стремится быстро реагировать на меняющиеся требования рынка.

Разработка программного обеспечения становится сегодня одним из ключевых бизнес-процессов как для организаций, использующих программное обеспечение для решения повседневных задач, так и для компаний, для которых разработка программного обеспечения является существенной составляющей производства, дистрибуции и продажи их продукции.

Программное обеспечение как бизнес-процесс

Последнее десятилетие было отмечено ростом понимания значимости информационных технологий (ИТ) в развитии современной экономики вообще и каждого конкретного предприятия в частности.

Сегодня ни у кого не вызывает сомнений, что инвестиции в ИТ должны быть направлены на совершенствование бизнес-процессов компании с целью выделения ее конкурентных преимуществ, оптимизации затрат и поддержки изменений в бизнесе. Отсюда следует, что программное обеспечение играет все более важную роль в достижении организациями целей своего бизнеса.

Разработка программного обеспечения становится частью бизнеса компании и одним из ключевых бизнес-процессов, тесно интегрированным с другими процессами и инфраструктурой предприятия. Согласно этой концепции, связь бизнес-целей, операционной деятельности и задач автоматизации в компании представляет собой замкнутый цикл:

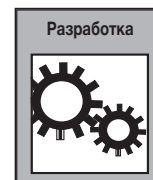
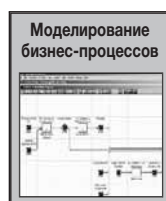


Новая концепция процесса разработки ПО

- Документирование и описание характеристик существующих процессов «как есть», с указанием основных метрик
- Определение и выработка целей, задач и требований
- Применение технологий для совершенствования процессов
- Моделирование процессов «как должно быть»



- Моделирование и симуляция бизнес-процессов
- Моделирование приложений и данных
- Анализ финансовых ресурсов и идентификация сфер, приносящих максимальную выгоду
- Идентификация и подготовка существующих ресурсов или их повторное использование



- Быстрая интеграция и/или разработка приложений
- Визуальное конструирование и генерация программного кода
- Выполнение функционального и нагрузочного тестирования
- Генерация кода XML и управление планами UML и потоками workflow
- Применение готовых паттернов для ускорения разработки

- Аудит процессов и их улучшений
- Осуществление последовательных улучшений
- Моделирование процессов «как есть» и «как должно быть» на следующем этапе



- Управление тестированием, требованиями, конфигурацией и управление проектом



Роль ИТ в компании, действующей в соответствии с моделями бизнес-процессов

цели бизнеса определяют новые потребности в автоматизации, которые затем закрываются в процессе создания или развития программных решений. Результат внедрения новых информационных систем, который может быть проанализирован и измерен, создает новые возможности для развития бизнеса, что, в свою очередь, создает новые потребности в автоматизации и т.д. Таким образом, успешные организации больше не ограничиваются просто автоматизацией бизнес-процессов, они также отслеживают процесс их выполнения и обеспечивают обратную связь в реальном времени, позволяющую совершенствовать бизнес-процессы в соответствии с изменяющимися потребностями клиентов. Несмотря на то, что ключевые элементы автоматизации многих бизнес-процессов сегодня можно купить в «готовом» виде, существует множество других задач, которые имеют уникальный характер для каждого бизнеса. Тиражируемые решения, такие как системы управления ресурсами предприятия (ERP), системы управления цепочкой поставок (SCM), системы управления взаимоотношениями с клиентами (CRM) и многие другие, предлагают стандартные варианты реализации соответствующих функций и обеспечивают минимальные риски в процессе их внедрения, эксплуатации и модернизации. Но чтобы выделить компанию в ряду конкурентов, внедрения этих продуктов недостаточно, нужны адаптированные решения, которые учитывают специфику данной компании и отражают ее уникальные ключевые практики. Автоматизацию подобных процессов невозможно осуществить, просто купив готовые пакеты приложений. Вот некоторые примеры таких процессов для конкретных отраслей:

- Страхование - заключение договоров, ранжирование клиентов и обработка заявлений от клиентов
- Финансы - трейдерские и брокерские услуги, управление инвестиционными портфелями и урегулирование спорных вопросов
- Туризм и транспорт - фрахтование, обслуживание инфраструктуры и эксплуатация оборудования

С другой стороны, если ранее каждый бизнес-процесс рассматривался как последовательность автономных, независимых друг от друга действий, которые поддерживались различными системами ИТ, то с течением времени процесс бизнес-интеграции дошел до такой стадии, на которой все действия рассматриваются как компоненты единого, горизонтально интегрированного бизнес-процесса, поддерживаемого единым программным обеспечением.

В связи с этим, все больше компаний начинают понимать ценность интегрированной платформы разработки приложений для повышения эффективности взаимосвязанных процессов при создании программных продуктов. А сам процесс разработки программного обеспечения становится важной составляющей бизнес-процессов компании, направленной на решение трех основных задач:

- Увязывание потребностей бизнеса с решениями в сфере ИТ
- Организация совместной работы специалистов
- Обеспечение прозрачности решений, контроля затрат и управления рисками

В этот процесс оказываются вовлечены сотрудники компании на всех уровнях, каждый из которых получает ощутимые преимущества от использования интегрированной платформы разработки ПО:

- Руководители высшего звена и бизнес-менеджеры получают возможность направить ИТ на реализацию целей своего бизнеса и отслеживать отдачу от инвестиций, как в рамках всего портфеля ИТ, так и в разрезе отдельных проектов
- Руководители отдела разработки ПО и руководители проектов получают новые возможности для контроля за состоянием проекта и обеспечения качества конечного продукта
- Члены команды разработки получают еще больше возможностей для эффективной, согласованной совместной работы, которая исключает непродуктивные операции, дублирование усилий или несогласованные действия
- Сотрудники службы внедрения и эксплуатации получают эффективные средства настройки, развертывания и сопровождения программных систем



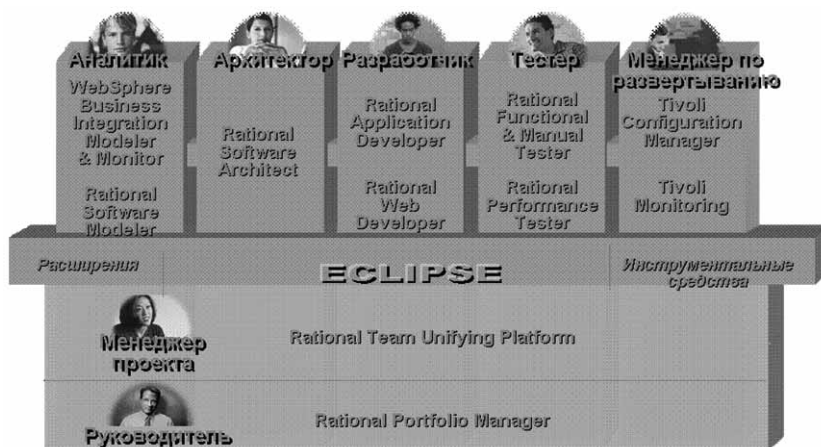
Организация коллективной работы с IBM SDP

IBM Rational Suite в свое время стала первым интегрированным решением для разработки программного обеспечения. Теперь идеи коллективной работы получили дальнейшее развитие на уровне всей компании, а IBM Software Development Platform стала первой интегрированной платформой, которая позволяет увязать задачи разработки ИТ с целями бизнеса.

Портфель решений, основанный на ролях

IBM Software Development Platform предоставляет полный набор средств, ориентированных на реализацию идеи о том, что для более четкого управления затратами и повышения предсказуемости бизнеса, инвестиции в ИТ должны быть увязаны с целями организации.

Предлагавшиеся ранее линейки средств поддержки жизненного цикла разработки были нацелены исключительно на то, чтобы команда могла успешно завершить проект по разработке программного обеспечения. Новые интегрированные средства поддержки разработки IBM существенно расширяют эти возможности и дополняют их поддержкой бизнес-ориентированного подхода к разработке программного обеспечения. При этом в платформе IBM Software Development Platform реализован ролевой подход к решению задач и обеспечивается максимум возможностей для их интеграции на протяжении всего цикла разработки программного обеспечения.



Интегрированная платформа IBM SDP, основанная на ролях

Например, в сфере проектирования и разработки предлагается решение широкого круга задач, с которыми специалисты ежедневно сталкиваются на практике. Разработчикам требуются высокопродуктивные инструменты для создания кода, а бизнес-аналитикам нужны средства описания задачи на языке, знакомом представителям бизнеса.

Примеры решений, ориентированных на задачи проектирования и разработки

Высокопроизводительная разработка, визуализация кода и визуальное редактирование	Rational Application Developer Rational Web Developer
Моделирование и сквозное проектирование - от начала до конца и обратно	Rational Application Developer Rational Modeler Rational Rose XDE Rational Rose XDE Developer Plus
Интеграция с унаследованными системами	WebSphere Studio Enterprise Developer
Быстрая разработка - прямая и архитектурно-ориентированная	Rational Web Developer
Бизнес-моделирование и исполнение моделей	WebSphere Business Integration Modeler
Поддержка требований, определяемых условиями разработки	Языки разработки <ul style="list-style-type: none"> • Операционные среды • Уровни профессиональной квалификации • Парадигмы моделирования

Техническая основа IBM Software Development Platform

Основу IBM Software Development Platform IBM представляет комплексный набор сервисов, базирующихся на открытых стандартах. Это позволяет обеспечить целостность и единообразие средств разработки, предлагаемых IBM, и гарантировать открытость системы для партнеров и заказчиков, которым предоставляется возможность доступа к имеющимся средствам разработки и расширения их функциональных возможностей за счет добавления собственных сервисов.

Техническая инфраструктура платформы IBM Software Development Platform состоит из пяти ключевых элементов:

- Открытая интегрированная платформа разработки Eclipse
- Среда моделирования Eclipse Modeling Framework
- Открытая платформа для Model-Driven Development (MDD)
- Модель программирования WebSphere
- Портфель решений для ролевой разработки

Кроме того, IBM внесла свой вклад в развитие систем с открытым кодом в виде метамodelей и инструментария средств разработки. Речь идет о метамодели UML2 и технологии Hyades Testing Framework, которые доступны бесплатно сообществу разработчиков в рамках проекта Eclipse. Оба проекта опираются на open source технологии, и, в свою очередь, вносят свой вклад в развитие систем с открытым исходным кодом, обеспечивая более совершенный уровень интеграции средств разработки в рамках IBM Software Development Platform.

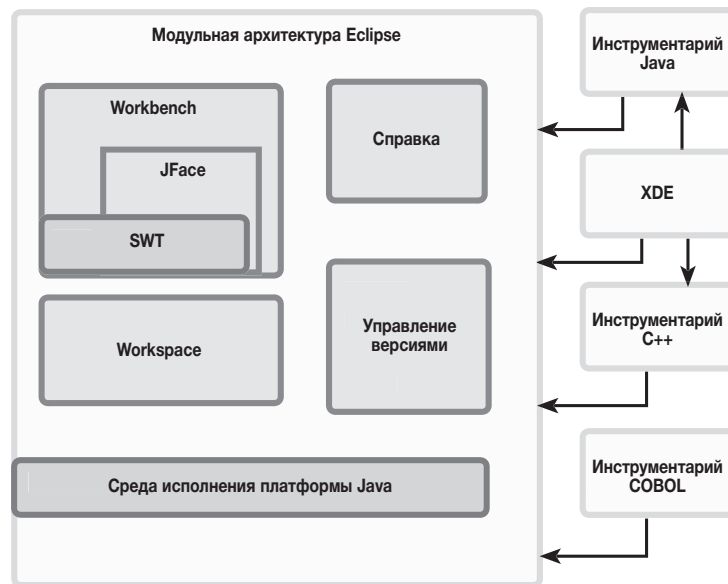
Eclipse

Eclipse представляет собой открытую, высоко интегрированную инструментальную платформу разработки, в состав которой входит общая платформа, обеспечивающая интеграцию инструментов, и среда разработки Java, основанная на этой платформе. Следует отметить, что многие другие open source проекты расширяют платформу Eclipse и используют созданные на ее базе инструменты, обеспечивающие поддержку специальных подходов и технологий разработки.

Основу Eclipse составляет инструментальная платформа, которая предлагает набор базовых функций для использования архитектуры подключаемых модулей (plug-in architecture), расширяющих и адаптирующих платформу Eclipse. Более того, архитектура Eclipse сама по себе состоит из основы и набора инструментальных средств, которые также состоят из отдельных подключаемых модулей.

В некотором смысле проект и платформа Eclipse напоминают более ранние проекты по созданию общей инфраструктуры для интеграции инструментальных средств, но обладает рядом существенных отличий:

- Гибкая архитектура подключаемых модулей. При разработке Eclipse значительное внимание было уделено созданию расширяемой архитектуры, основанной на использовании подключаемых модулей (plug-in's). Эта архитектура хорошо зарекомендовала себя, что подтверждается сотнями проектов по созданию расширений Eclipse. Многие из этих проектов имеют открытый код, но следует отметить, что появляется и все большее количество коммерческих продуктов на базе Eclipse.
- Популярные открытые технологии в основе Eclipse. Технология Eclipse базируется на ряде существующих open source технологий, широко используемых сообществом разработчиков, таких как Ant, JUnit, Xerces и другие. Таким образом, разработчикам практически не требуется времени, чтобы разобраться с возможностями новой платформы - она оказывается знакомой с самого начала и доступна для более глубокого изучения.



Модульная архитектура Eclipse

- Eclipse - открытый стандарт. Eclipse не только использует технологии с открытым кодом, но и вносит свой вклад в развитие open source проектов. Программное обеспечение Eclipse доступно на условиях Общей публичной лицензии (Common Public License, CPL), а работу по созданию Eclipse координирует некоммерческий фонд, члены которого являются сотрудниками различных организаций. Такой подход стимулирует максимальное широкое применение Eclipse и экспериментирование с ней как в коммерческих, так и в учебных организациях.
- Полнота функциональности интегрированных инструментальных средств. В отличие от многих предыдущих интегрированных платформ разработки, возможности платформы Eclipse не ограничиваются предоставлением общего формата данных для используемых инструментальных средств, но и поддерживает такие ключевые аспекты интеграции, как управление метаданными и создание пользовательского интерфейса (UI). Например, Eclipse включает инструментальный пользовательского интерфейса Standard Widget Toolkit (SWT), а также использует сервисы JFace, предлагающие стандартные способы отображения и управления элементами пользовательского интерфейса. Всё эти возможности объединены в рамках автоматизированного рабочего места (workbench), описывающего общую структуру интегрированной среды разработки (IDE) Eclipse.

Проект Eclipse изначально был ориентирован на разработчиков Java, которым были необходимы легко настраиваемые средства разработки следующего поколения. На сегодняшний день Eclipse завоевала большую популярность в сообществе Java разработчиков и достигнуты значительные успехи в области создания коммерческих продуктов на базе Eclipse, включая семейство продуктов IBM WebSphere Studio. Многие крупные компании, работающие с программным обеспечением - как поставщики, так и потребители, поддерживают Eclipse, являются членами Eclipse Foundation и вносят свой вклад в разработку программного обеспечения или используя Eclipse в качестве основного компонента их технической инфраструктуры. Около 200 поставщиков уже имеют или планируют к выпуску продукты, реализованные на основе Eclipse. Все это позволяет технологии Eclipse быстро развиваться, достигая нового уровня зрелости и ширины охвата.

Интеграция: использование Eclipse для обеспечения более тесной интеграции компонентов проекта



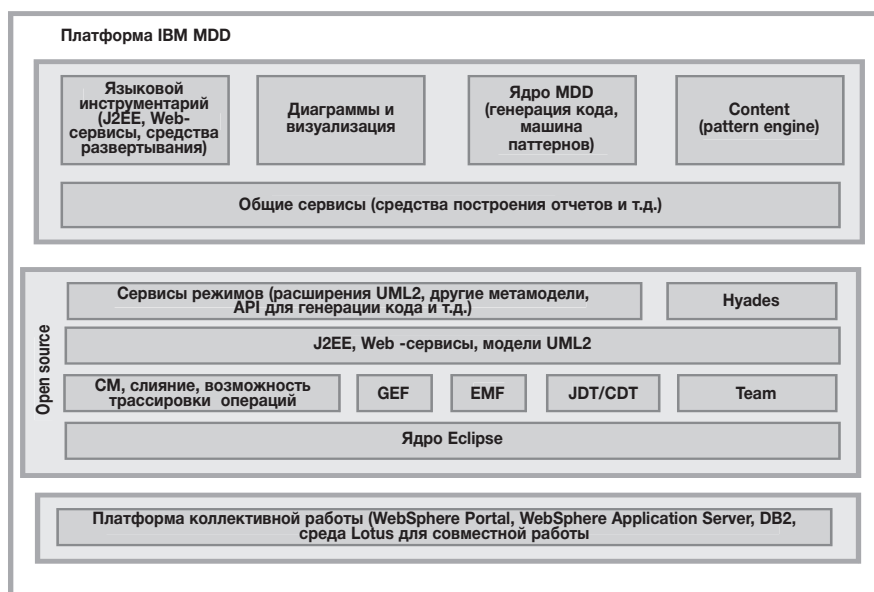
Архитектура Eclipse

В последнее время особое внимание уделялось развитию инфраструктуры Eclipse. С добавлением новых инструментальных средств после приобретения IBM CrossWorlds®, Holosofx® и Rational, IBM Software Group смогла расширить возможности предлагаемых средств разработки, которые теперь охватывают более широкий спектр ролей, проектов и предметных областей. Многие из приобретенных программных пакетов уже поддерживали определенный уровень взаимодействия с базовыми компонентами инструментальных средств IBM, например, при помощи общих форматов файлов или процедур импорта/экспорта, использующих стандартные механизмы обмена.

Тем не менее, важное значение имело именно объединение всех этих средств разработки на основе единой, технологической платформы, отвечающей таким потребностям заказчиков, как уровень интеграции, гибкость и расширяемость. Именно в таком, объединяющем качестве выступает технология Eclipse для IBM Software Development Platform. Технология Eclipse во многом ускорила создание многофункциональной, интегрированной платформы для разработки программного обеспечения благодаря возможностям организации взаимодействия между различными группами специалистов, работающих над IBM Software Development Platform.

Как показано на рисунке, платформа Eclipse выполняет три основных функции в IBM Software Development Platform. Во-первых, Eclipse предоставляет платформу и набор сервисов для графического интерфейса пользователя, являющиеся общими для всей платформы IBM Software Development Platform. Eclipse, таким образом, обеспечивает богатый пользовательский интерфейс для клиентов и высокий уровень целостности при выполнении различных действий в рамках IBM Software Development Platform. Во-вторых, Eclipse обеспечивает улучшенный механизм использования одних и тех же информационных массивов при выполнении различных действий в системе благодаря применению группы общих моделей, выраженных в технологии EMF. В-третьих, Eclipse реализует мощные функции инфраструктуры для коллективной работы в рамках IBM Software Development Platform.

Эти функции интегрированы в инфраструктуру Eclipse в виде подключаемых модулей и, соответственно, в таком же виде доступны другим подключаемым модулям IBM Software Development Platform.



Ключевая роль Eclipse в IBM Software Development Platform

Среда моделирования Eclipse Modeling Framework

Для того чтобы интеграция средств разработки имела смысл, необходимо наличие общих соглашений о множестве базовых артефактов и процессов, совместно используемых средствами разработки. Подробная семантика общего доступа задается с использованием Eclipse Modeling Framework (EMF) – среды моделирования для платформы Eclipse.

EMF – это инструментарий и средства генерации кода, которые обычно используются при описании структур данных, которыми манипулирует приложение. На основе UML модели, XML схемы или интерфейса Java EMF генерирует соответствующие классы реализации. Одна из основных задач, решаемых EMF, состоит в соотношении сущностей моделирования непосредственно с их реализацией. Это дает Eclipse преимущества моделирования в совокупности с низким уровнем затрат при переходе к кодированию.

EMF призвана обеспечить унифицированное представление "структур данных", описанных в приложении, независимо от того, были ли эти структуры определены при помощи языка UML, в XML-схеме, либо в виде интерфейсов Java. Например, при создании приложения, которое будет манипулировать структурой сообщений XML, можно определить схему XML. Затем при помощи EMF можно сгенерировать диаграмму классов UML для этой схемы. Кроме этого, можно сгенерировать набор Java классов реализации для обработки XML. Аналогичным образом, если вы начинаете с кода Java, описывающего ключевые интерфейсы в дизайне приложения, при помощи EMF можно сгенерировать соответствующую модель UML и структуру сообщений XML.

Модели, описываемые в EMF, представлены в виде внутренней модели, именуемой Ecore. EMF представляет собой реализацию Meta Object Facility (MOF), выполненную IBM. MOF представляет собой стандарт описания хранилищ метаданных, разработанный Object Management Group (OMG). MOF определяет подмножество операторов UML для описания понятий моделирования классов внутри хранилища объектов. Соответственно, MOF походит на Ecore возможностью описания структурных и поведенческих характеристик классов, наследования, пакетов и способов отображения (reflection). Отличие между MOF и Ecore состоит в том, что MOF обладает дополнительными комплексными функциями управления жизненным циклом, структурами данных, связями между пакетами и сложными ассоциативными типами.

Открытая платформа MDD, построенная на основе Eclipse

Теперь, познакомившись с описанием основных компонентов Eclipse, становится понятно, какую важную роль играет Eclipse в составе IBM Software Development Platform. Платформа Eclipse предоставляет открытую, расширяемую инструментальную среду, обладающую такими характеристиками, как богатый пользовательский интерфейс, архитектура на основе подключаемых модулей, облегчающая расширение системы, и мощная среда моделирования, обеспечивающая глубокий уровень семантической интеграции.

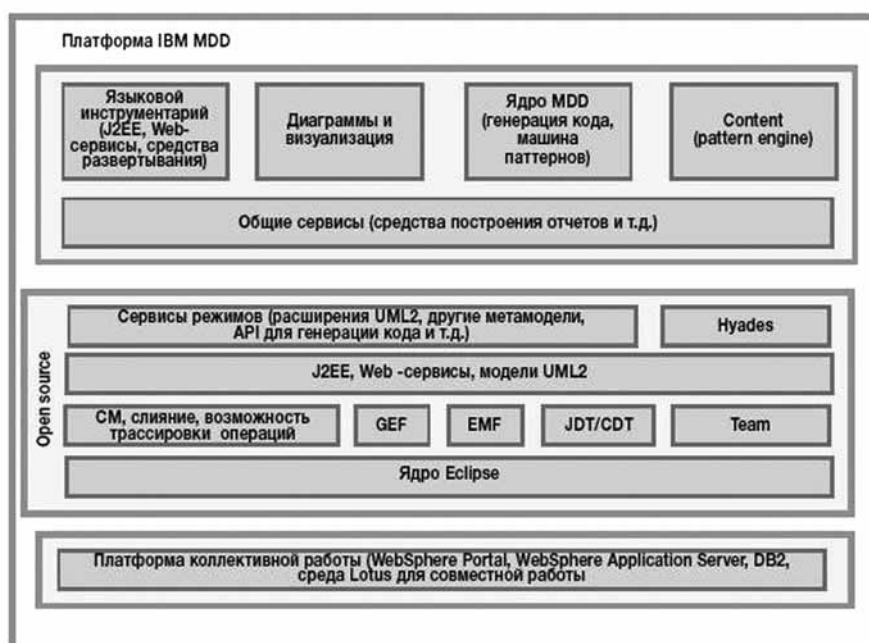
Таким образом, опираясь на платформу Eclipse, процесс интеграции можно описать при помощи следующих этапов:

- С помощью EMF разрабатываются метамодели, учитывающие специфику технологии или инструментальных средств. Если возможно, они базируются на стандартных метамоделях, применяемых в отрасли (например, UML), при необходимости стандартные метамодели расширяются и модифицируются.
- Большинство компонентов инфраструктуры, обеспечивающей интеграцию средств разработки, генерируются на основе моделей EMF. По мере эволюции инструментальных средств и механизма их интеграции компоненты инфраструктуры можно сгенерировать заново на основе упомянутых моделей, если возникнет такая необходимость.
- Производится уточнение и обеспечивается совместный доступ к общей семантике метаданных, описанной при помощи таких языков, как UML, XSD, XMI, Annotated Java и др.
- Все взаимодействия между инструментальными средствами осуществляются с использованием общего программного интерфейса (Java).
- Для обмена метаданными между инструментальными средствами внутри системы, а также для внешнего взаимодействия с инструментальными средствами, разработанными партнерами, применяется общий подход (XML)..
- На основе моделей генерируются простые функции реализации (операции CRUD, базовые функции редактирования и т.д.)
- Для осуществления внутренних преобразований применяются метамодели для J2EE, Web-сервисов и MDD.

Полученная в итоге техническая инфраструктура IBM Software Development Platform, как показано на рисунке, состоит из трех основных компонентов:

- В основе инфраструктуры лежит группа open source технологий, реализованных в рамках проекта Eclipse. Сюда относятся ядро Eclipse, различные подключаемые модули и группа метамodelей, описанных в EMF.
- Дополнительные возможности, предлагаемые IBM, реализованы на базе упомянутых выше open source технологий. Эти возможности используются во многих продуктах IBM и предоставляют техническим специалистам IBM целый спектр повторно используемых сервисов.
- В основе всех названных возможностей лежит платформа коллективной работы. Она включает в себя базовые технологии IBM для организации совместного доступа к данным, управления артефактами и сбора информации.

Две технологические разработки особенно ярко иллюстрируют мощные возможности и открытость IBM Software Development Platform, базирующейся на данной технологической инфраструктуре: это метамодель UML2 и инструментарий тестирования Hyades.



Компоненты технической инфраструктуры

UML2

Метамодель UML2 представляет собой реализацию основных элементов спецификации UML 2.0 в EMF. Эта метамодель является основной реализацией UML 2.0, лежащей в основе всех инструментальных средств, объединенных платформой IBM Software Development Platform, и, следовательно, представляет базовые семантические элементы, составляющие основу IBM Software Development Platform. Эта метамодель широко используется различными инструментальными средствами IBM; если необходимо, ее можно модернизировать для каждого из инструментальных средств, воспользовавшись стандартными механизмами расширения UML.

Метамодель UML2 разработана участниками open source проекта в рамках сообщества Eclipse. Задачами данного проекта являются создание удобных в использовании реализаций:

- метамодели, поддерживающей разработку средств моделирования;
- общей схемы обмена метаданными - XML Metadata Interchange (XMI) для упрощения обмена семантическими моделями;
- а также описание вариантов тестирования в качестве механизма проверки спецификации, и специфицирование правила проверки, как средства описания технических условий и обеспечения соответствия им.

Hyades

Инструментарий тестирования Hyades - Hyades Testing Framework - обеспечивает централизованный механизм управления ресурсами и жизненным циклом тестирования для всех средств разработки, объединенных платформой IBM Software Development Platform. Эта среда обеспечивает синхронизацию всех этапов тестирования на протяжении жизненного цикла, стимулируя упреждающее тестирование (test-first) в процессе разработки и улучшая возможности трассировки от тестовых артефактов к другим артефактам в рамках жизненного цикла. Hyades поддерживает весь спектр методологий тестирования за счет использования инфраструктуры с открытым исходным кодом, посредством которой средства тестирования и трассировки взаимодействуют друг с другом.

Hyades Testing Framework является open source проектом в рамках сообщества Eclipse. Цель проекта Hyades заключается в интеграции средств тестирования и трассировки в рамках среды Eclipse для обеспечения совместимости со средствами разработки на протяжении жизненного цикла программного обеспечения. Такая интеграция позволяет снизить затраты и облегчить процесс внедрения эффективных автоматизированных процессов контроля качества программного обеспечения. Hyades позволяет улучшить средства функционального и нагрузочного тестирования, используемые разработчиками, за счет повышения уровня совместимости и снижения затрат на приобретение инструментальных средств и последующих затрат на сопровождение.

Для описания всех артефактов, используемых в процессе тестирования (например, тестов и трассировок) Hyades использует язык UML. Для обеспечения совместимости инструментальных средств и применяемых понятий артефакты тестирования имеют форму, определенную в тестовом профиле OMG. Кроме того, артефакты совместимы со стандартом MOF, поэтому данные о них можно сохранять и извлекать при помощи EMF.

Модель программирования WebSphere

Одним из ключевых аспектов IBM Software Development Platform является использование модели программирования, реализованной в рамках платформы WebSphere, в ряде приложений IBM промежуточного слоя (DB2, Tivoli и Lotus) и, в частности, в рамках IBM Software Development Platform.

Ключевые элементы модели программирования, общие для IBM Software Development Platform и платформы IBM промежуточного слоя, включают в себя:

- Service Data Objects (SDO), в настоящий момент находятся в стадии стандартизации в рамках Java Community Process. Объекты SDO предоставляют упрощенную модель программного доступа к данным для различных ресурсов (как данных, так и корпоративных информационных систем - EIS), и служат дополнением к базовым стандартам для Web-сервисов, таким, как XML, Web Services Definition Language (WSDL) и Simple Object Access Protocol (SOAP).
- BPEL4WS, представляющий собой стандарт согласования сервисов (service orchestration) и написания скриптов для компонентов, поддерживающий интеграцию процессов workflow и бизнес-процессов.

- JSF, представляющий собой инструментарий Java, позволяющий ускорить процесс создания Web-приложений разработчиками, не являющимися большими специалистами в J2EE.
- Адаптация и настройка приложений с использованием внешних политик и правил. В настоящее время разрабатывается целая серия стандартов, регламентирующих описание и применение политик, в том числе такие стандарты, как Web Services Policy и OMG Business Semantics of Business Rules (BSBR).

Таким образом, IBM Software Development Platform – это самое полное и мощное в индустрии решение для создания программных систем и управления жизненным циклом приложений. Она обеспечивает поддержку автоматизации практически для всех аспектов разработки программного обеспечения.

Продукты, услуги и лучшие методы организации работ Rational лежат в основе IBM Software Development Platform. Это модульное и функционально законченное решение позволяет командам разработчиков использовать подход к разработке, учитывающий интересы бизнеса и основанный на открытых стандартах, включая оболочку Eclipse. Результатом этого подхода является исключительная эффективность бизнеса.

Современный подход к созданию сложных программных систем

При создании сложного программного обеспечения (ПО) для корпоративных информационных систем (ИС) требуется четко и грамотно организовать весь процесс разработки/заказа ПО — от написания технического задания до внедрения на предприятии и дальнейшего развития этого ПО. Среди основных проблем, возникающих при разработке ПО без использования специальных технологий, можно выделить следующие:

- **Разночтения в требованиях.** Разработчики и пользователи разговаривают на "разных языках", что не позволяет точно перевести разрозненные неформальные требования в целостную формальную спецификацию системы. В результате трудно создать систему, отвечающую требованиям пользователей. Необходимы постоянные доработки и изменения.
- **Отсутствие "чертежей".** Отсутствие проектных спецификаций ("чертежей") на систему приводит к отсутствию структуры и единой концепции системы. Развитие такой системы трудоемко и ведет к дальнейшему росту "хаотичности".
- **Документирование постфактум.** Трудоемкость документирования в ходе разработки выливается либо в неприемлемые сроки создания точной проектной документации в соответствии с требованиями стандартов, либо в неприемлемое качество документации, что влечет за собой проблематичность последующей модификации ПО ИС.
- **Ошибки проектирования.** Ошибки, возникающие на этапах анализа и проектирования, часто не удается обнаружить до самого начала внедрения, когда уже стоимость их исправления становится на порядок выше.
- **Отсутствие общего контекста проекта.** Подсистемы, создаваемые разными группами разработчиков, трудно интегрировать из-за отсутствия или недостаточной проработки общего контекста проекта.
- **Обособленность проекта.** Информационные системы не переносятся с одной платформы на другую, имеют сложное взаимодействие с внешними системами и являются тяжелыми для последующего развития. В результате разработка нового и изменение существующего программного обеспечения отнимают слишком много времени и средств.

Мировой опыт показывает, что для успешного создания подобного ПО необходимы апробированные современные методологии, опирающиеся на мощные и удобные инструментальные средства. Осуществление таких проектов в заданные сроки с высоким качеством невозможно без применения инженерных методов автоматизации программного производства, т.е. без современных CASE-технологий.

Основные принципы организации работы над проектом

Ведущие идеологи инструментальной инфраструктуры IBM Rational (Г. Буч, Дж. Рамбо, А. Джекобсон), проанализировав опыт различных проектов в области разработки ИС, выделяют следующие обязательные факторы для успешного ведения любого проекта:

- постоянное взаимодействие с потенциальными пользователями с целью выяснения реальных требований к системе;
- тщательно проработанная архитектура системы, открытая для возможных усовершенствований;
- наличие высококвалифицированных специалистов;
- грамотно подобранный инструментарий;
- определение верного направления работ;
- продуманный процесс разработки, обеспечивающий адаптацию к изменяющимся потребностям бизнеса и требованиям новых технологий;
- высокая степень управляемости проектом и получение достоверной информации по его состоянию в любой момент времени.

При наблюдаемом в настоящее время взрывном росте количества приложений, как для исполнителя, так и для заказчика, необходимо выполнение высококачественных программных проектов быстрее, чем когда бы то ни было.

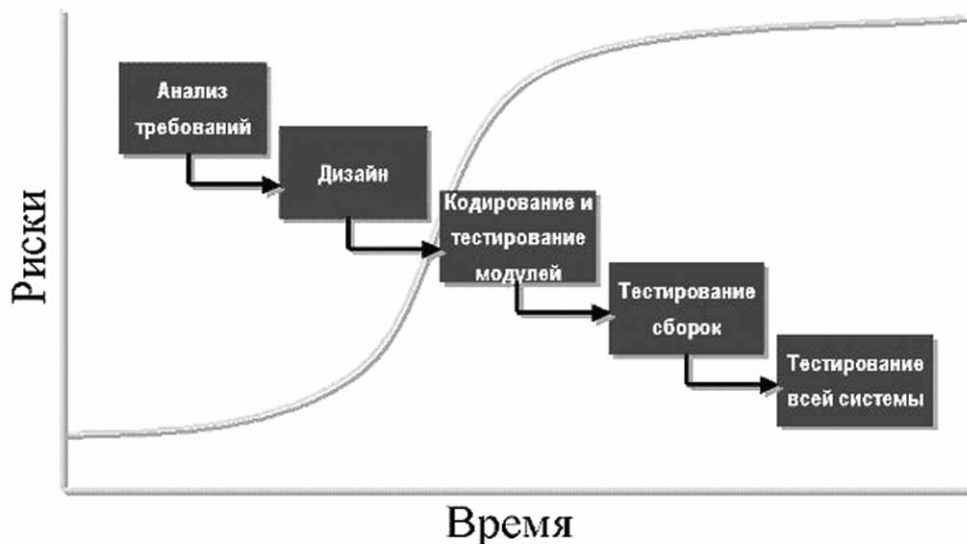
Программные проекты должны завершаться в ограниченные сроки и при этом оставаться в рабочем состоянии с гарантией качества. Возникает ключевая проблема — необходимо достичь баланса между качеством исполнения и скоростью разработки. Решения IBM Rational помогут вам преодолеть эту проблему, объединяя лучший опыт и методологии разработки, соответствующие требованиям качества SEI CMM/CMMI, с унифицированными инструментами и сервисом, ускоряющими промышленную разработку ПО.

Основная задача, которая должна быть решена для достижения успеха в проекте, — научиться разрабатывать и производить программное обеспечение наиболее предсказуемым и повторяемым образом. Участники проектов должны уметь повторять свой успех в будущих работах и своевременно устранять обнаруженные недостатки. Чтобы гарантировать успех в любом проекте, важно использовать стандартные практики, которые давно уже являются обязательными:

- разрабатывать ПО на основе итеративных принципов;
- управлять требованиями наиболее эффективным образом;
- использовать компонентный подход;
- проектировать систему с помощью визуальных средств;
- гарантировать качество создаваемых продуктов;
- контролировать любые изменения в ходе проекта.

Итеративная разработка

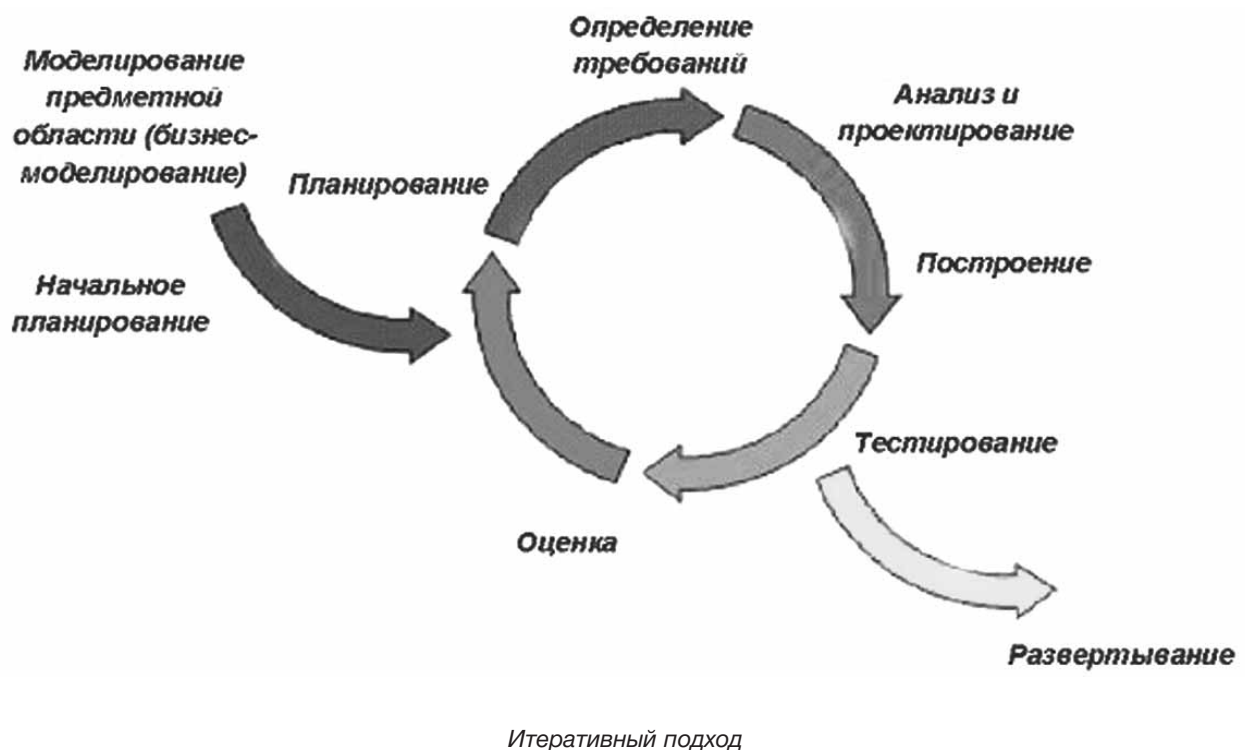
Классический подход, широко применявшийся в прошлом и до сих пор часто встречающийся в настоящее время, — разработка программного обеспечения по методу “водопада” (см. рис.).



Метод “водопада”

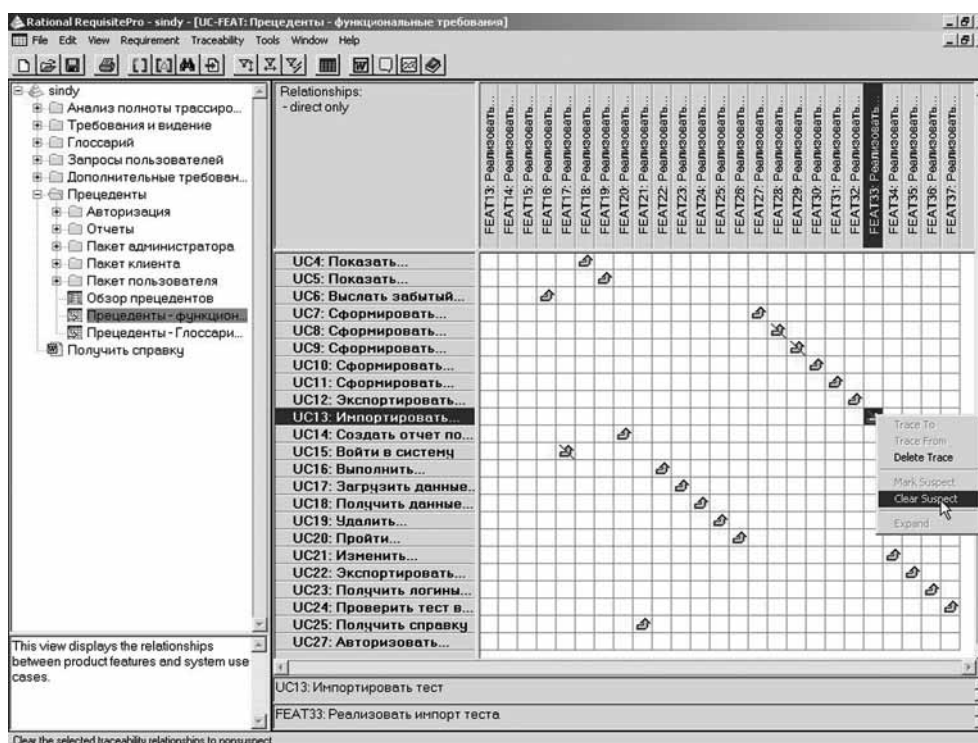
При этом подходе разработка ПО движется линейно через стадии анализа требований, дизайна (проектирования), кодирования и тестирования отдельных модулей (компонентов), тестирование сборок и интегрированное тестирование всего конечного продукта. Основная проблема здесь заключается в том, что происходит нарастание риска преждевременного крушения проекта из-за накопления различных ошибок, допущенных на ранних стадиях проекта. Если только к концу проекта, становится очевидно, что такие ошибки были допущены, то любой возврат к предыдущим стадиям с целью исправления ошибок становится крайне дорогостоящим. Метод “водопада” не позволяет эффективно выявлять и нивелировать последствия подобных рисков. “Если вы сами активно не атакуете риски, то потом они будут активно атаковать вас” (Том Глиб, IBM Rational).

Эффективной альтернативой методу “водопада” служит итеративный подход (см. рис.).



Основой итеративного подхода является непрерывный анализ выполненных работ, последующее проектирование и физическое воплощение результатов проектирования. Итеративный подход акцентирует работу команды в более предсказуемом и повторяемом направлении. Основные преимущества итеративного подхода:

- нивелирование воздействия серьезных рисков на ранних стадиях проекта, пока это еще можно сделать с минимальными затратами;
- возможность организовать плодотворную обратную связь с будущими конечными пользователями с целью создания системы, реально отвечающей их потребностям;
- акцент усилий на наиболее важные и критичные направления проекта.
- непрерывное итеративное тестирование конечного продукта, позволяющее оценить успешность всего проекта в целом;
- раннее обнаружение несоответствий между требованиями, моделями и программным кодом;
- более равномерная загрузка участников проекта;
- эффективное использование накопленного опыта;
- реальная оценка текущего состояния проекта и, как следствие, большая уверенность заказчиков и непосредственных участников в его успешном завершении.



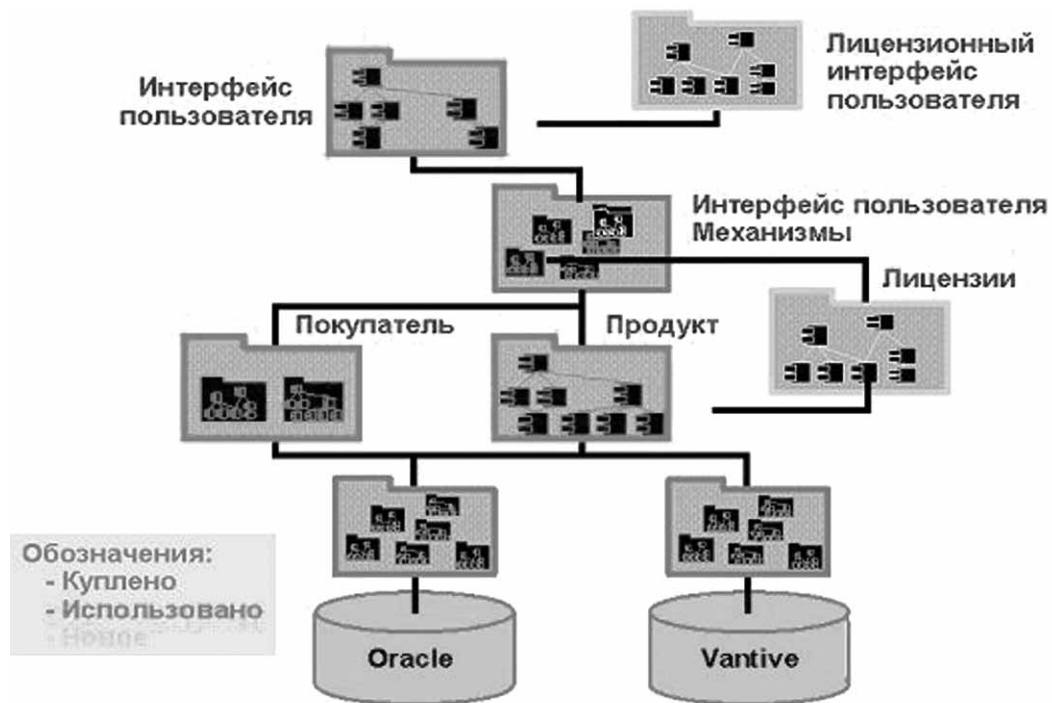
Управление требованиями

Эффективное управление требованиями

Неизбежной реальностью проектов создания современных ИС, является то, что список требований к системе динамически изменяется в ходе всего жизненного цикла. В любой момент следует ожидать пересмотра и корректировки требований. Более того, непрерывное выявление и уточнение реальных требований к системе — один из путей успешного ведения проекта. На старте проекта практически невозможно наиболее точным образом и в полном объеме определить необходимые требования, за исключением наиболее простых систем.

Реализация каждого отдельного требования представляет собой добавление в систему некоторой способности, которой та должна соответствовать. Процесс управления требованиями охватывает несколько деятельности: выявление, организацию и документирование. Существуют проверенные решения, которые позволяют гарантировать более эффективное ведение проекта, основанное на грамотном управлении требованиями. Они опираются на следующие аспекты:

- организованный подход к управлению требованиями;
- взаимодействие участников проекта на базе выявленных и утвержденных требований; ранжирование требований по приоритету, фильтрация их по необходимым параметрам и выявление зависимости между ними для контроля изменений;
- объективная оценка реализованного функционала и полученной производительности;
- раннее предсказание различных несоответствий и расхождений;
- использование инструментальных средств для организации более эффективного процесса управления требованиями.



Пример использования компонентного подхода

Компонентный подход

Компонентный подход при создании ПО крайне важен, т. к. он обеспечивает повторное использование уже существующих компонентов и позволяет эффективно распределить работу между участниками проекта.

Применение компонентного подхода дает возможность повысить эффективность процесса разработки следующим образом:

- использование компонентов повышает гибкость архитектуры создаваемой системы;
- модульность компонентов позволяет четко определить границы тех изменений, которые требуется внести в систему при ее доработке;
- наличие множества коммерческих компонентов, которые разработаны и протестированы третьими фирмами, а также построены на основе промышленных спецификаций COM+, CORBA, Enterprise Java Beans (EJB) и др., облегчает реализацию и позволяет экономить проектные ресурсы;
- программные компоненты задают естественную основу для конфигурируемости продукта;
- средства визуального моделирования обеспечивают автоматизацию процесса разработки, опирающегося на компонентный подход.

Визуальное проектирование системы

Модель — это абстрагирование, упрощение реальности, позволяющее описать систему в predetermined разрезе. Модели формируются для выявления того, какая система в конечном итоге должна быть построена.

Используя универсальный язык моделирования UML (Unified Modeling Language), участники проекта могут эффективно взаимодействовать друг с другом. Продуманное моделирование в различных его проявлениях предоставляет следующие возможности:

- однозначное описание функционального поведения системы с помощью прецедентов и сценариев;
- спецификация и анализ технических особенностей системы с помощью моделей;
- ранний акцент на построение гибкой и надежной архитектуры.
- сокращение излишней детализации;
- выявление и исключение на ранних стадиях проекта ошибок проектирования.



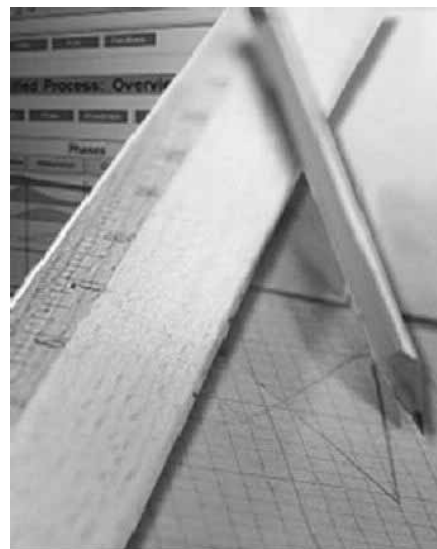
Визуальное моделирование

Гарантия качества продуктов

Чем раньше участники проекта начинают заботиться о качестве разрабатываемой системы, тем дешевле им это обходится. Непрерывный контроль качества реализуется с помощью тестирования. Данный процесс предполагает создание тестов для каждого ключевого сценария, реализуемого в системе. Качество системы проявляется, прежде всего, в количестве успешных и неуспешных сценариев, что как раз и выявляется в процессе тестирования. Тестирование и разработка новых тестовых сценариев проводятся на каждой итерации проекта. Наборы сценариев и программных скриптов дорабатываются итеративно вместе с создаваемым продуктом.

Непрерывный контроль качества приводит к следующим позитивным моментам:

- оценка состояния проекта приобретает большую объективность, т. к. оценивается реальное функционирование системы, а не качество проектной документации;
- оценка проекта позволяет раскрыть несоответствия в требованиях, моделях и реализации;
- тестирование акцентирует внимание на тех сторонах работы системы, которые имеют наибольшую важность и повышенный риск;
- дефекты выявляются на ранних стадиях, что снижает затраты на их устранение;
- автоматизированное тестирование обеспечивает высокий уровень функциональности системы, надежности и производительности.



Контроль изменений

Для любого проекта характерна задача по организации работы между различными людьми и их группами. Отсутствие продуманного управления процессом разработки неизбежно приводит к возникновению хаотичности проекта и снижению его эффективности. Основой для объективного мониторинга проекта служит налаженный контроль изменений, который позволяет своевременно реагировать на возникающие проблемы. Контроль изменений приводит к следующему:

- деятельность по изменению требований становится предсказуемой и повторяемой;
- запросы на изменения формируют основу четкого и налаженного взаимодействия между участниками проекта;
- статистика по запросам на изменения создает отличную базу для оценки состояния проекта в любой момент времени;
- все изменения проекта находятся под контролем;
- с помощью контролируемых изменений строится система, адекватно отвечающая потребностям заказчиков.

Методология Rational Unified Process

В основе многих программных методологий, объединяющих инженерные методы создания ПО, лежит "пошаговый подход". Он определяет этапы жизненного цикла, контрольные точки, правила работ для каждого этапа и, тем самым, упорядочивает проектирование и разработку ПО.

Для каждого этапа жизненного цикла методология задает:

- состав и последовательность работ, а также правила их выполнения;
- распределение полномочий среди участников проекта (роли);
- состав и шаблоны формируемых промежуточных и итоговых документов;
- порядок контроля и проверки качества.

Ведущей методологией, в которой инструментально поддерживаются все этапы жизненного цикла разработки ПО, является методология Rational Unified Process (RUP). Она опирается на проверенные практикой методы анализа, проектирования и разработки ПО, методы управления проектами. RUP обеспечивает прозрачность и управляемость процесса и позволяет создавать ПО в соответствии с требованиями заказчика на момент сдачи ПО, а также в соответствии с возможностями инструментальных средств поддержки разработки.

RUP как методология

Методология RUP позволяет объединить проектную команду, предоставляя в ее распоряжение проверенные мировой практикой лучшие подходы к разработке ИС. К ним относятся такие процессы жизненного цикла создания ПО, как управление проектами, бизнес-моделирование, управление требованиями, анализ и проектирование, тестирование и контроль изменений. Внедрение RUP в организации способствует выработке качественных внутрикорпоративных стандартов и повышению общей культуры разработки.



Основа RUP — итеративный процесс разработки. В условиях активно развивающегося мирового бизнеса практически невозможно создавать современные сложные программные системы последовательно, т. е. сначала выявлять все проблемы, затем принимать проектные решения, потом формировать программное обеспечение и, наконец, проверять полученное изделие. Итеративный подход позволяет улучшать понимание проблем на основе последовательных усовершенствований и конкретизировать их в эффективных решениях. Этот подход обеспечивает большую гибкость при изменяющихся требованиях и тактических коррективах в бизнес-целях, что позволяет более эффективно и заблаговременно идентифицировать и снижать проектные риски.

RUP — управляемый процесс. Итеративный подход предполагает управление требованиями и изменениями, чтобы между всеми участниками проекта обеспечивать единое понимание ожидаемых функциональных возможностей, требуемый уровень качества, наилучшее управление затратами и графиками выполнения работ.

RUP — процесс создания и физического воплощения визуальных моделей. RUP фокусирует внимание не на создании большого количества бумажных документов, а на развитии и применении визуальных моделей — семантически богатых представлений разрабатываемой ИС. RUP сосредотачивает внимание на разработке и дальнейшем развитии надежной и гибкой архитектуры, которая облегчает параллельную разработку, минимизирует необходимость изменений, увеличивает возможность многократного использования и надежность эксплуатации системы. Подобная архитектура применяется для планирования использования программных компонентов и управления их развитием.

RUP — процесс управления действиями с помощью прецедентов, определяющих функционал системы. Понятия прецедентов и сценариев работы способствуют эффективному управлению технологическим маршрутом от бизнес-моделирования и требований вплоть до испытаний. Они обеспечивают связанные и доступные для анализа направления разработки и развертывания системы.

RUP поддерживает объектно-ориентированную технологию. Моделирование по методологии RUP является объектно-ориентированным и базируется на понятиях объектов, классов и зависимостей между ними. Эти модели, подобно многим другим техническим искусственным объектам (артефактам), в качестве единого стандарта для организации взаимодействия участников проекта используют Unified Modelling Language™ (UML) — универсальный язык моделирования.

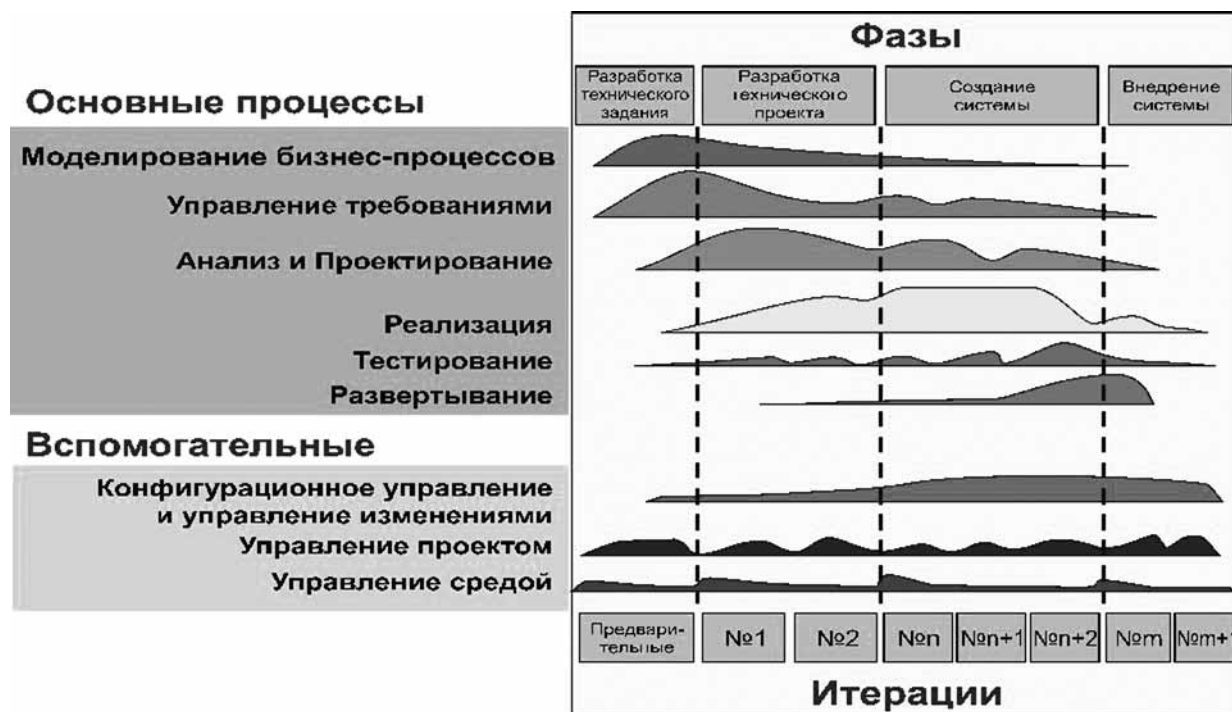
RUP поддерживает компонентно-ориентированный подход. Компоненты — это нетривиальные модули или подсистемы, которые выполняют конкретную функцию и могут быть использованы многократно. Как правило, компоненты соответствуют одной из промышленных спецификаций, таких как CORBA, COM/DCOM, ActiveX, Enterprise Java Beans и др.

RUP — адаптируемый и конфигурируемый процесс. Опыт единичного проекта, даже успешно завершено, вряд ли подойдет для создания ПО во всех случаях и условиях. Но способность RUP к адаптации подойдет как маленьким группам разработчиков, так и большим организациям. RUP содержит рекомендации по конфигурированию процесса для удовлетворения потребностей практически любых компаний и их подразделений.

RUP поддерживает объективно осуществляемое управление качеством. Оценка качества всех работ, выполняемых любыми участниками проекта, использует объективные метрики и критерии. Методология RUP создавалась с прицелом на поддержку управления качеством в рамках требований SEI CMM/CMMI.

Структура жизненного цикла проекта

Структуру жизненного цикла проекта, выполняемого по технологии RUP удобно рассматривать на координатной плоскости. При этом по горизонтальной оси отложено время, а по вертикальной — основные деятельности, которые обычно выполняются в ходе любого проекта, претендующего на статус успешного.



Процессы и фазы жизненного цикла проекта

Время отражает динамический аспект жизненного цикла проекта, выраженный в терминах циклов, фаз, итераций и контрольных точек, которые разделяют две отдельные фазы. Вертикальная ось отражает статический аспект проекта. Он описывается в терминах процессов, артефактов (единица информации, создаваемая или модифицируемая в ходе любого процесса) и ролей (ответственность за адекватное выполнение той или иной деятельности в процессе). RUP организует выполнение проекта по фазам, каждая из которых состоит из одной или нескольких итераций. При итеративном подходе объем работ по каждому процессу варьируется в течение всего жизненного цикла. Контрольные точки в конце фаз позволяют оценить, насколько успешной была предыдущая фаза и насколько успешен весь проект в целом. RUP определяет следующие основные процессы:

- моделирование бизнес-процессов;
- управление требованиями;
- анализ и проектирование;
- реализация;
- тестирование;
- развертывание;
- конфигурационное управление и управление изменениями;
- управление проектом;
- управление средой.

Моделирование бизнес-процессов применяется с тем, чтобы разобраться в структуре исследуемой предметной области, обеспечить единство понимания основных автоматизируемых процессов среди всех участников проекта и определить высокоуровневые требования, которые должны быть реализованы в ходе проекта.

Управление требованиями позволяет прийти к соглашению с заказчиками и конечными пользователями, определить, что должна уметь делать создаваемая система, предоставить более четкие инструкции участникам проекта о возможностях системы, создать базу для успешного планирования работ в проекте и оценки его статуса в любой момент жизненного цикла.

Анализ и проектирование служат для последовательного преобразования выявленных требований к системе в спецификации особого вида, которые описывают, как следует конкретно реализовать конечный продукт. Следует при этом делать различия между анализом и проектированием. Основное из них состоит в том, что спецификации анализа не зависят от конкретной платформы и технологии, для которой осуществляется создание ИС. А спецификации проектирования являются точным представлением проектируемой системы, часто позволяя автоматизировать процесс генерации на их основе программного кода.

Реализация необходима для выявления порядка организации программного кода в терминах отдельных подсистем, преобразования исходного кода в выполняемые компоненты, тестирования созданных компонентов и интеграции отдельных компонентов в подсистемы и системы.

Тестирование позволяет определять и контролировать качество создаваемых продуктов, следить за тем, насколько качественно осуществлена интеграция компонентов и подсистем, все ли требования к системе реализованы и все ли выявленные ошибки устранены до того, как система будет развернута на оборудовании конечного пользователя.

Развертывание является процессом, в ходе которого осуществляется доставка разрабатываемого продукта к конечному пользователю. В ходе данного процесса производится новый выпуск системы, распространение ПО, его установка на стороне конечного пользователя, обучение последнего навыкам эффективной работы с поставленным ПО, предоставление услуг по технической поддержке, бета-тестирование и т. п.

Конфигурационное управление и управление изменениями позволяет организовать эффективную работу с артефактами проекта, контролировать и управлять доступом к ним, вести историю изменений, обеспечить эффективное взаимодействие участников проекта, как в простых командах, так и в распределенных, находящихся на большом удалении друг от друга.

Управление проектом включает в себя непосредственное формирование условий для эффективного хода всего проекта, определение руководств и руководящих принципов для планирования, формирования команды и мониторинга проекта, выявление и управление рисками, организацию работы участников проекта, формирование бюджета, планирование фаз и итераций.

Управление средой позволяет осуществить поддержку всех участников проекта. В эту поддержку входят выбор инструментария и его приобретение, настройка и установка, конфигурирование процесса, доработка и адаптация методологии, используемой для ведения проекта, обучение.

Важнейшие акценты RUP

Главная цель любой организации, занимающейся созданием информационных систем — работать эффективнее, а значит, быстрее создавать более качественные продукты и получать бизнес-преимущества от успешного ведения проектов. Внедрение передовой методологии, подобной RUP, позволяет гарантировать выработку и дальнейшее развитие в организации необходимых для этого навыков.

Однако внедрение методологии – не столь уж простой процесс, как это может показаться на первый взгляд. Очень важно, стремясь к более эффективному ведению проектов, не разрушить то, что уже

достигнуто. Особенность методологии RUP в том, что она может быть настроена и адаптирована в соответствии с особенностями и требованиями организации-разработчика, при этом варианты внедрения RUP могут варьироваться в зависимости от конкретных условий.

Для упрощения перехода к методологии RUP допускается постепенное его внедрение. Но при этом RUP акцентирует внимание на нескольких важнейших элементах, без которых сложно гарантировать успех в проекте.

- **Общее видение проекта.** Сюда относятся первоначальный анализ будущего проекта, организация единого словаря для общения и ведение спецификации требований. Это важно потому, что участники проекта должны четко понимать цели проекта.
- **Бизнес-перспективы проекта.** Важны потому, что в основном проект выполняется для реализации каких-либо бизнес-целей. И если такие цели существуют, то имеет смысл начинать процесс разработки. Это не относится напрямую к научным и исследовательским проектам, т. к. их финансирование имеет иные корни.
- **План работ.** Позволяет определить ресурсы проекта, привязать их к задачам и рассчитать бюджет проекта. Таким образом, имеется возможность заранее спрогнозировать, насколько выгодно вести конкретный проект и какие могут быть при этом затраты.
- **Анализ рисков.** Важен потому, что обычно намного легче и дешевле выявить и устранить возможные проблемы заранее, чем делать это уже на поздних стадиях проекта.
- **Гибкая и надежная архитектура системы.** Гарантирует, что проект не потерпит крушение задолго до его завершения, что разработчики смогут развивать данную систему при изменении условий и правил ведения бизнеса на стороне заказчика.
- **Управление запросами на изменения.** Позволяет организовать эффективную работу и взаимодействие участников проекта. Возрастает контроль за качеством выполнения задания любого уровня, отслеживанием устранения ошибок и обработки предложений по дальнейшему развитию ИС.
- **Тестирование.** Дает возможность гарантировать высокое качество продукта, а, следовательно, не даст заказчику повод усомниться в возможностях организации-разработчика.
- **Акцент на самом продукте.** Крайне важен, потому что продукт — конечная цель любого проекта. Надо помнить в любой момент, что важны не модели или многочисленные документы проекта сами по себе, а именно конечный продукт. Все остальное создается только с тем, чтобы как можно скорее создать качественный продукт.
- **Документы для поддержки пользователя.** Необходимы, т. к. без них многие сильные стороны созданного продукта могут остаться неизвестными и недоступными.
- **Измерение проекта.** В любой момент времени необходимо, чтобы вовремя реагировать на возможные отклонения проекта от бюджета и на перерасход ресурсов.

Программные решения IBM Rational

Инструментальные средств IBM Rational

Инструментальные программные средства платформы IBM Rational интегрированы на основе среды Eclipse и ориентированы на решение следующих основных задач:

- Управление проектами и портфелями.
- Поддержка основных процессов создания программного обеспечения – моделирования, разработки и тестирования.
- Организация совместной работы проектной команды.

Для управления проектами используется пакет **IBM Rational Portfolio Manager**, обеспечивающий управление портфелем проектов разработки программного обеспечения.

Средства моделирования, разработки и тестирования программного обеспечения, создаваемого на платформе J2EE входят в состав пакета **IBM Rational Professional Bundle**, который объединяет новые продукты линейки IBM Rational, получившей название Atlantic. Отличительной чертой пакета IBM Rational Professional Bundle является тесная интеграция средств проектирования, разработки и тестирования в рамках единой среды разработки на базе платформы Eclipse. Это позволяет быстро переходить от проектирования к разработке и легко визуализировать код в графическом представлении на языке моделирования UML. Новые инструменты интегрированы со средствами организации совместной работы IBM Rational Team Unifying Platform на основе среды Eclipse.

IBM Rational Professional Bundle включает полный комплект инструментальных средств, необходимых для проектирования, разработки и тестирования приложений, ориентированных на J2EE/порталы/службы, на платформах Windows и Linux, а также для тестирования .NET-приложений:

- IBM Rational Software Architect - инструменты проектирования и разработки приложений на основе моделей на языке UML, которые позволяют выполнять детальное проектирование информационной системы и обеспечивать создание качественной архитектуры. Включает такие средства как IBM Rational Software Modeler, IBM Rational Application Developer и IBM Rational Web Developer
- Инструменты тестирования в составе новой линейки продуктов Atlantic – это средство нагрузочного тестирования IBM Rational Performance Tester и средство функционального тестирования IBM Rational Functional Tester (включает средство ручного тестирования IBM Rational Manual Tester)

Для организации совместной работы по моделированию и тестированию программных систем для различных платформ разработки, не поддерживаемых IBM Rational Professional Bundle, успешно используется пакет IBM Rational Suite, объединяющий такие хорошо знакомые разработчикам средства, как IBM Rational Rose XDE DeveloperPlus, IBM Rational Robot, IBM Rational PurifyPlus for Windows, а также средства организации коллективной работы проектной команды IBM Team Unifying Platform.

Пакет IBM Rational Team Unifying Platform обеспечивает совместную работу проектной команды по управлению требованиями, изменениями, тестированием, конфигурациями, документированию и отслеживанию состояния проекта в соответствии с методологией процесса разработки программного обеспечения. В состав пакета входят следующие инструментальные средства: IBM Rational RequisitePro, IBM Rational ProjectConsole, IBM Rational ClearQuest, IBM Rational ClearCase LT, IBM Rational TestManager, IBM Rational SoDA, IBM Rational Unified Process.

Состав пакетов и краткое описание основных средств IBM Rational, предназначенных для автоматизации процессов разработки программного обеспечения, представлены в следующей таблице.

Задача	Продукт	Описание
Управление проектами	IBM Rational Portfolio Manager	Эффективное решение, автоматизирующее управление и Managerи мониторинг портфеля проектов организации
Моделирование, разработка и тестирование	IBM Rational Professional Bundle	<p>Пакет, объединяющий новую линейку продуктов Atlantic для моделирования, разработки и тестирования программного обеспечения, создаваемого на платформе J2EE.</p> <p>В состав пакета входит весь комплекс программных средств, необходимых предприятию для разработки, конструирования и тестирования приложений, ориентированных на J2EE/порталы/службы, на платформах Windows и Linux, а также для тестирования приложений .NET. Интегрируется со средствами организации совместной работы IBM Rational Team Unifying Platform на основе среды Eclipse.</p> <p>В состав пакета входят следующие средства: IBM Rational Software Architect IBM Rational Performance Tester IBM Rational Functional Tester</p> <p>Пакет приобретается одновременно с единым контрактом на техобслуживание</p>
	IBM Rational Suite	<p>Пакет средств организации совместной работы по моделированию и Suite тестированию программных систем для различных платформ разработки.</p> <p>В состав пакета входят следующие средства: IBM Team Unifying Platform IBM Rational PurifyPlus for Windows IBM Rational Rose XDE DeveloperPlus IBM Rational Robot</p>
Моделирование и разработка	IBM Rational Software Architect	<p>Инструмент проектирования и разработки приложений на основе моделей на языке UML. Позволяет выполнить детальное проектирование информационной системы и обеспечить создание качественной архитектуры.</p> <p>Инструменты моделирования интегрированы со средствами разработки.</p> <p>В состав пакета входят следующие инструментальные средства:</p>

		<p>IBM Rational Software Modeler IBM Rational Application Developer IBM Rational Web Developer Существует как отдельный продукт и входит в состав пакета IBM Rational Professional Bundle</p>
Моделирование	IBM Rational Software Modeler	Инструмент визуального верхнеуровневого моделирования и проектирования, позволяющий описывать и документировать представления разрабатываемой системы для различных категорий заинтересованных лиц. Существует как отдельный продукт и входит в состав пакета IBM Rational Software Architect
	IBM Rational Rose	CASE-средство визуального моделирования информационных систем, позволяющее описывать их общесфункциональные аспекты, проводить визуальный анализ и проектирование, разрабатывать компонентную архитектуру и описывать схемы развертывания этих систем. Существует как отдельный продукт и входит в состав пакета IBM Rational Suite.
	IBM Rational XDE Modeler	Средство, предназначено для визуального моделирования информационных систем на основе UML v1.4. Существует как отдельный продукт.
Разработка приложений	IBM Rational Application Developer	Средство для быстрого проектирования, разработки, анализа, тестирования, профилирования и развертывания приложений Web, Web-служб, Java, J2EE и порталов в полной среде IDE на базе Eclipse. Пришел на смену Rational Rapid Developer. Существует как отдельный продукт и входит в состав пакета IBM Rational Software Architect
	IBM Rational XDE Developer	Инструментарий, интегрируемый в рабочую среду разработчика (Microsoft Visual Studio .NET, IBM Websphere Studio Application Developer, Eclipse IDE) и позволяющий организовать Round-Trip разработку информационных систем. Существует как отдельный продукт.
Разработка Web-сайтов	IBM Rational Web Developer	Средство для быстрой разработки статических и динамических Web сайтов, их тестирования и развертывания. Существует как отдельный продукт и входит в состав пакета IBM Rational Software Architect.
Нагрузочное тестирование	IBM Rational Performance Tester	Средство нагрузочного тестирования в линейке продуктов Atlantic. Существует как отдельный продукт и включает в состав своего пакета IBM Rational Functional Tester
	IBM Rational Robot + Test Manager	Используется для нагрузочного тестирования, поддерживает большое количество протоколов.
Функциональное тестирование	IBM Rational Functional Tester	Средство функционального тестирования в линейке продуктов Atlantic. В состав пакета входит: IBM Rational Manual Tester
	IBM Rational Manual Tester	Средство для организации и проведения ручного тестирования информационных систем в линейке продуктов Atlantic.

	IBM Rational Robot	Средство, предназначенное для разработки и записи скриптов IBM Rational Application Developer IBM Rational Web Developer. Существует как отдельный продукт и входит в состав пакета IBM Rational Professional Bundle
Тестирование	IBM Rational XDE Tester	Средство, ориентированное, главным образом, на функциональное тестирование Java приложений и Web-сайтов.
	IBM Rational Purify	Средство, предназначенное для помощи разработчику в выявлении ошибок, связанных с обращением к динамической памяти. Входит в состав пакета IBM Rational Suite
	IBM Rational Quantify	Средство, предназначенное для помощи разработчику в выявлении узких мест в коде, оказывающих влияние на производительность разрабатываемой информационной системы. Входит в состав пакета IBM Rational Suite
	IBM Rational Application Developer	Средство для быстрого проектирования, разработки, анализа, тестирования, профилирования и развертывания приложений Web, Web-служб, Java, J2EE и порталов в полной среде IDE на базе Eclipse. Пришел на смену Rational Rapid Developer. Существует как отдельный продукт и входит в состав пакета IBM Rational Software Architect
	IBM Rational PureCoverage	Средство, предназначенное для помощи разработчику в определении полноты тестирования кода. Входит в состав пакета IBM Rational Suite
	IBM Rational TestFactory	Средство, предназначенное для полуавтоматического формирования набора тестовых скриптов, предназначенных для проведения функционального тестирования и обеспечивающих его полноту для конкретной информационной системы. Входит в состав пакета IBM Rational Suite
	IBM Rational Team Unifying Platform	Обеспечивает совместную работу проектной команды по управлению требованиями, изменениями, тестированием, конфигурациями, документированию и отслеживанию состояния проекта в соответствии с методологией процесса разработки программного обеспечения. В состав пакета входят следующие инструментальные средства: IBM Rational RequisitePro IBM Rational ProjectConsole IBM Rational ClearQuest IBM Rational ClearCase LT IBM Rational TestManager IBM Rational SoDA IBM Rational Unified Process Существует как отдельный продукт и входит в состав пакета IBM Rational Suite. На платформе Eclipse интегрирован с инструментальными средствами пакета IBM Rational Professional Bundle.
	IBM Rational RequisitePro	Средство, позволяющее организовать процесс эффективного управления требованиями в организации Существует как отдельный продукт и входит в состав пакета IBM Rational Team Unifying Platform.

Организация совместной работы	IBM Rational ProjectConsole	Средство мониторинга ключевых показателей проекта, которое упрощает контроль за статусом проекта и генерацию объективных метрик проекта с целью улучшения его предсказуемости. Входит в состав пакета IBM Rational Team Unifying Platform.
	IBM Rational ClearQuest	Средство, позволяющее организовать процесс эффективного управления изменениями в организации. Существует как отдельный продукт и входит в состав пакета IBM Rational Team Unifying Platform
	IBM Rational ClearCase	Семейство масштабируемых и наращиваемых средств конфигурационного и версионного управления в проекте, которое при использовании с другими продуктами IBM Rational позволяет организовать гибкие и контролируемые процессы разработки информационных систем.
	IBM Rational TestManager	Средство, позволяющее централизованно контролировать и управлять процессами автоматизированного тестирования информационных систем, как функционального, так и нагрузочного, включая вопросы его планирования и подготовки. Входит в состав пакета IBM Rational Team Unifying Platform
	IBM Rational SoDA	Средство для формирования проектной документации на основе автоматической генерации отчетов и документов для следующих проектных репозиторий: <ul style="list-style-type: none"> • Репозиторий требований Rational RequisitePro, • Репозиторий тестирования Rational TestManager, • Базы данных запросов на изменения Rational ClearQuest, • Версионные объектные базы (VOB) Rational ClearCase, • Общий проектный репозиторий Rational Administrator. Входит в состав пакета IBM Rational Team Unifying Platform
	IBM Rational Unified Process	База знаний по унифицированному процессу разработки программного обеспечения, реализованная в виде Web-сайта и содержащая инструментальные средства адаптации к специфике процессов разработки. Входит в состав пакета IBM Rational Team Unifying Platform

Средства по управлению проектами и портфелями

Управление проектом по разработке программного обеспечения — это своего рода искусство балансирования между конкурирующими целями, рисками, различными ограничениями и обстоятельствами. Основной задачей данного процесса является обеспечение успешной поставки продукта, удовлетворяющего потребностям заказчиков - основных плательщиков по счетам, и потребителей - конечных пользователей. Основные цели управления проектами:

- организация процесса управления проектом, планирование проекта на протяжении всего жизненного цикла и отдельной итерации;
- соблюдение основных принципов планирования, управления персоналом, выполнения работ и мониторинга проекта с помощью соответствующих метрик;
- эффективное управление рисками.

Компании и организации обычно ведут много проектов, которыми необходимо управлять – как традиционных IT-проектов, например, по разработке ПО или внедрению автоматизированной системы, так и проектов в других областях. Управление портфелем проектов - это подход, который позволяет держать под контролем широкий диапазон проектов и ресурсов, обеспечивая необходимый уровень их управляемости. Помимо управления единым финансовым портфелем, управление проектами включает множество различных процессов – управление ресурсами, затратами, рисками, качеством, а также другие связанные с этим процессы, и все они должны выполняться совместно.

В составе новой линейки средств IBM Rational предлагается надежный инструмент для управления проектами и портфелями, который играет ключевую роль в обеспечении процесса разработки программного

обеспечения, управляемого бизнес целями компании - **IBM Rational Portfolio Manager**. Этот инструмент предоставляет командам разработчиков возможности по управлению проектами, значительно превышающие возможности программы MS Project, в то же время позволяя интегрироваться с ней.

IBM Rational Portfolio Manager

IBM Rational Portfolio Manager - инструмент для управления проектами и портфелями - как на уровне предприятия в целом, так и отдельного проекта, а также в разрезе IT. Руководители компании получают инструмент, с помощью которого они имеют возможность анализировать и управлять проектами разработки программного обеспечения, исходя из приоритетов бизнес целей, финансовой эффективности, имеющихся ресурсов и проектных рисков. Менеджерам проектов и членам проектных команд IBM Rational Portfolio Manager обеспечивает проектную среду и необходимые инструменты для подготовки, планирования, выполнения и завершения задач проекта.

IBM Rational Portfolio Manager позволяет автоматизировать управление проектами и портфелями в целом – от определения возможностей и приоритетов до реализации и завершения проекта. Этот продукт позволяет менеджерам проектов, менеджерам портфелей, руководителям и другим лицам, заинтересованным в выполнении проектов, получать в реальном времени информацию о ходе выполнения всех проектов и планов, входящих в их портфели. Это помогает улучшить управление и контроль за ходом выполнения множества проектов, а также автоматизировать поток жизненно важной информации, необходимой для управления одним, несколькими или множеством проектов. Менеджеры проектов могут опуститься на уровень задач и подзадач, чтобы получить возможность управления затратами, распределением ресурсов, календарными планами проекта и ресурсов, а также управлять масштабом, сроком окончания и выпуском.

В первую очередь, IBM Rational Portfolio Manager ориентирован на потребности IT-подразделений, связанных с управлением проектами, позволяя обеспечить управление всеми элементами IT-инфраструктуры как единым портфелем активов. По данным аналитиков, уже сегодня две трети руководителей компаний управляют IT как портфелем или как комплексом инвестиций в IT, а к 2006 году по крайней мере половина организаций IT-отрасли будет использовать специализированные приложения для управления портфелями проектов, предназначенные для управления совместной работой команд, распределения ресурсов и отслеживания усилий и затрат. Это означает, что популярность решений для управления портфелем, к которым относится IBM Rational Portfolio Manager будет расти. При этом IBM Rational – единственное решение, объединяющее возможности интеллектуального управления портфелем проектов с ведущими инструментами и методологией разработки приложений, которое позволяет обеспечить планирование, управление и измерение IT-проектов и активов на протяжении всего жизненного цикла.

IBM Rational Portfolio Manager интегрирован с другими решениями IBM, включая WebSphere, Tivoli и Lotus. В перспективе IBM Rational Portfolio Manager будет также использоваться и для решения более широкого диапазона задач, включая поддержку таких моделей и стандартов как Sarbanes/Oxley, Six Sigma, Application Portfolio Management, Product Lifecycle Management и т.п.

Функциональные возможности. IBM Rational Portfolio Manager позволяет достичь высокого уровня прозрачности и предсказуемости при выполнении IT-проектов, который нужен для принятия стратегических решений менеджерами проектов, менеджерами программ и руководством. Этот продукт подходит для управления проектами любого масштаба, но для проектов разного размера будут использоваться разные возможности. Например, для небольшого проекта могут потребоваться только функции Document (Документы), Issues (Результаты) и Risk Management (Управление рисками), а также функция Process Work Flows (Потоки заданий процесса), которые помогут упростить управление проектом. Для управления крупными проектами могут потребоваться все функции Rational Portfolio Manager, а именно:

- Управление портфелем (Portfolio Management) – управление перечисленными выше процессами на уровне предприятия
- Управление стандартными проектами (Standard Project Management) – аналог Microsoft Project (MSP). Эта функция предназначена для создания иерархической структуры работ (Work Breakdown Structure) и диаграммы Гантта (Gantt Chart), а также для распределения усилий участников команды. Этот инструмент предоставляет возможности по управлению проектами, значительно превышающие возможности программы MS Project, в то же время обеспечиваются необходимые механизмы интеграции средств IBM Rational и Microsoft Project. В частности, интеграция IBM Rational RequisitePro

с Microsoft Project позволяет осуществить экспорт функциональных требований в план проекта и на их базе сформировать задачи, которые надлежит выполнить для успешного завершения проекта. Задачи, сформированные в Microsoft Project, могут быть детализированы, распределены по времени, по ресурсам, а также дать основу для проведения необходимых расчетов бюджета проекта. В общем случае, когда программа MSP используется совместно с Rational Portfolio Manager, она работает как инструмент для составления расписаний. Когда программа MSP используется автономно, она используется также и как инструмент планирования.

- Методы многократного использования (Reusable Methods) – позволяя использовать шаблоны Rational Unified Process и шаблоны документов, эти средства предназначены для построения структуры проекта и распределения задач между его участниками.
- Управление рисками (Risk Management) - эта функция позволяет собирать и отображать информацию о рисках, связанных с проектом или с организацией, а также анализировать возможности сокращения рисков.
- Управление ресурсами (Resource Management) - эта функция позволяет управлять данными о персонале, собирать и отслеживать информацию о времени выполнения работ и затратах, а также составлять график загрузки сотрудников.

Среди других функций можно отметить управление документацией проекта, бухгалтерский учет по проекту и управление совместной работой над проектом, а также возможности для формирования и изменения требований к функционированию и управлению IT-структур на высоком уровне.

Богатые функциональные возможности Rational Portfolio Manager предназначены для поддержки всех сторон, заинтересованных в выполнении проекта:

- Менеджеры проектов могут планировать, отслеживать и контролировать выполнение проектов
- Члены команд проектов могут получать доступ к соответствующим стандартам, процедурам и к другим документам
- Члены команд проектов могут получать задания и докладывать о ходе их выполнения
- Руководители проектов могут поддерживать осведомленность о состоянии всех проектов, за которые они отвечают
- Менеджеры ресурсов могут эффективно управлять ресурсами, за которые они отвечают

Для работы Portfolio Manager необходима база данных DB2 или Oracle, а также Web-сервер WebSphere, WebLogic или Apache. Это ПО не входит в комплект поставки Portfolio Manager.

Rational Portfolio Manager и Team Unifying Platform (TUP). В общем случае, модели совместного использования этих продуктов различаются в зависимости от потребностей конкретного проекта. В большинстве случаев, если для управления проектом разработки ПО пользователю требуется дополнительная детализация, Team Unifying Platform хорошо подходит для решения этих задач. В то же время, для понимания бизнес-целей и контроля можно использовать Rational Portfolio Manager.

Rational Portfolio Manager – мощное серверное приложение с базой данных, с Web-доступом и репозиторием, изначально разработанное специально для улучшения управления проектами, рисками, расходами, ресурсами и активами как на уровне отдельного проекта, так и на уровне портфеля предприятия. В отличие от него, инструментальный Rational Team Unifying Platform (TUP) предназначен для управления полностью структурированными проектами разработки ПО и обеспечивает богатый набор функциональных возможностей для таких областей как управление требованиями, управление изменениями, управление тестированием и конфигурированием ПО и др.

В Rational Portfolio Manager предусмотрены базовые возможности для отслеживания списков приоритетных проблем и требований проекта, а также для управления изменением потока запросов высокого уровня и уровня проекта – эти возможности ориентированы, в первую очередь, на менеджеров проектов и руководителей. В то же время средства Team Unifying Platform, ориентированные на исполнителей, предоставляют гораздо больший уровень детализации, масштабируемости и настраиваемости. Таким образом, для управления проектами, не связанными с разработкой ПО и предполагающими только базовые возможности по управлению требованиями и изменениями, достаточно возможностей Rational Portfolio Manager. Для управления же большинством проектов по разработке ПО больше подходит сочетание Rational Portfolio Manager и TUP которое обеспечивает как процессы управления проектом, так и управления жизненным циклом разработки.

Rational Portfolio Manager имеет интерфейс с небольшими окнами (Portlet), которые позволяют пользователям получать доступ к другим приложениям, оставаясь в основной программе. Пользователь TUP, например, может просматривать такие приложения, как Rational Unified Process, ProjectConsole, RequisiteWeb или ClearQuest Web. Такая «интеграция» делает Portfolio Manager единой точкой входа к управлению проектами, которая позволяет совмещать детализацию проекта, предоставляемую TUP, с аспектами управления проектом, которые поддерживает Portfolio Manager. Обеспечить модель совместной

работы можно запустив в окне Rational Portfolio Manager клиента Rational Web, входящего в состав Team Unifying Platform. В число компонентов Team Unifying Platform, имеющих Web-интерфейс, входят также Rational Unified Process (RUP), Rational ProjectConsole, Rational RequisitePro, Rational ClearQuest и Rational ClearCase LT. Кроме этого, Rational SUMMIT Ascendant, хотя и не входит в состав Team Unifying Platform, позволяет проводить расчет затрат для решения Rational, поэтому эту программу тоже можно рассматривать в качестве кандидата для интеграции. Перечисленные возможности становятся доступны сразу после приобретения продуктов и не предполагают интеграции по данным, которые будут реализованы в следующих версиях.

Основные процессы создания программного обеспечения

Одной из ключевых особенностей новой линейки продуктов IBM Rational является более тесная интеграция предлагаемых средств поддержки жизненного цикла разработки.

Наиболее полно интеграция инструментальных средств основных процессов создания приложений на платформе J2EE - моделирования, разработки и тестирования - реализована в пакете **IBM Rational Professional Bundle**. Наличие единой интегрированной среды разработки на основе Eclipse позволяет легко переходить от моделирования высокоуровневых представлений бизнес процессов к проектированию модулей разрабатываемой программной системы с последующей их реализацией и тестированием.

Интегрированное решение по проектированию и разработке приложений – IBM Rational Software Architect (RSA), которое входит в IBM Rational Professional Bundle, содержит средства моделирования IBM Rational Software Modeler (RSM) и разработки приложений IBM Rational Application Developer (RAD), дополненное инструментом разработки сайтов IBM Rational Web Developer (RWD). Эти продукты позволяют создавать средства преобразования моделей, обеспечивающие автоматизированный переход от высокоуровневых моделей к низкоуровневым моделям реализации, вплоть до графического представления кода на UML. Средства поддержки процесса тестирования в составе пакета IBM Rational Professional Bundle – это IBM Rational Performance Tester – средство нагрузочного тестирования и IBM Rational Functional Tester - средство функционального тестирования, в состав которого входит также и средство ручного тестирования - IBM Rational Manual Tester.

Для создания приложений на платформах и инструментальных средствах, не поддерживаемых в пакете IBM Rational Professional Bundle, предназначен пакет **IBM Rational Suite**. В этот пакет, помимо средств моделирования, разработки и тестирования, включены также и средства организации совместной работы проектной команды – пакет IBM Rational Team Unifying Platform.

Пакет **IBM Rational Team Unifying Platform**, о котором подробнее речь пойдет в следующем разделе, является объединяющей платформой, которая рекомендуется как основа для организации взаимодействия в рамках коллективов разработчиков при использовании инструментальных средств обоих пакетов - IBM Rational Professional Bundle и IBM Rational Suite.

Более подробно средства моделирования, разработки и тестирования в составе пакетов IBM Rational Professional Bundle и IBM Rational Suite описаны ниже.

Моделирование и проектирование

Моделирование представляет собой один из ключевых процессов создания программного обеспечения, направленный на решение следующих задач:

- снижение сложности понимания предметной области;
- понимание структуры и динамики предметной области, в которой должна быть развернута система (целевой организации);
- понимание текущих проблем целевой организации и определение потенциальных возможностей усовершенствования;
- обеспечение общего понимания целевой предметной области заказчиками, конечными пользователями и разработчиками;
- выявление системных требований, необходимых для автоматизации предметной области.

Область моделирования охватывает следующие дисциплины:

- бизнес-моделирование предметной области, которая рассматривается как потенциальная для внедрения процессов автоматизации.

- функциональное моделирование системы, которая позволяет автоматизировать некоторую часть исследуемой предметной области.
- анализ и проектирование системы, которые позволяют сформировать детальное представление системы на уровне конкретных средств реализации.

Многие компании добились большей эффективности в процессе создания сложных программных систем благодаря средствам моделирования начального уровня, которые на протяжении ряда лет предлагались в составе продуктов IBM Rational и хорошо знакомы сообществу разработчиков. Имея невысокую стоимость в расчете на одного пользователя, они позволили бизнес-аналитикам эффективно описывать процессы и данные. Однако, по мере изменения и усложнения бизнес-требований, появилась потребность в использовании более развитых и совершенных средств моделирования и проектирования, которые отражали бы бизнес-ориентированный подход к процессу создания программных систем.

Для выполнения этих требований, IBM существенно расширило возможности традиционных средств моделирования, таких как Rational Rose и XDE Modeler, новыми инструментами в составе пакета **IBM Rational Software Architect (RSA)**. Одним из преимуществ новых средств моделирования является возможность автоматизированного преобразования моделей, позволяющая быстро переходить от высокоуровневого моделирования к разработке и тестированию приложений. Это позволяет более эффективно использовать шаблоны проектирования (patterns), стандарты и лучшие проектные решения для создания высококачественного кода и повышения общей эффективности проектов разработки программных систем. Базовые возможности новой линейки средств графического моделирования, дизайна и проектирования реализованы в продукте IBM Rational Software Modeler. IBM Rational Software Architect включает всю функциональность Rational Software Modeler, дополняя ее возможностями автоматизированного преобразования моделей и поддержкой C++ в дополнение к Java.

Кроме того, новые средства моделирования в линейке IBM исключают риски использования нестандартной технологии моделирования, жестко привязанной к технологии конкретного поставщика. Решения IBM Rational Software Architect построены на базе Eclipse - платформы с открытым исходным кодом, написанной на Java, что дает возможность быстро наращивать объем функциональных возможностей предлагаемых решений в соответствии с конкретными требованиями проекта. Eclipse поддерживает использование модулей расширения независимых разработчиков (plug-in), что также способствует созданию оптимальной среды моделирования приложений как в среде Windows, так и в среде Linux.

Специализированные средства моделирования позволяют автоматизировать повторяющиеся действия, тем самым повышая не только продуктивность, но и зрелость процесса разработки программного обеспечения в целом. Во многом этому способствует использование стандартизованного языка моделирования Unified Modeling Language (UML). Однако, как показывает опыт, текущей версии UML были присущи определенные ограничения, в частности, в сфере моделирования структуры и поведения сложных систем. Новые средства моделирования IBM поддерживают UML версии 2 (UML 2), в которой предлагаются структурированные классы, усовершенствованные блок-схемы, диаграммы действий и диаграммы конечных автоматов, что позволяет гораздо более точно отображать архитектуру будущих программных систем.

Кроме того, средства моделирования в составе IBM Rational Software Architect поддерживают Model Driven Architecture (MDA) – новую инициативу Object Management Group (OMG) в области описания управления процессами, благодаря которой становится возможным определять несколько уровней моделей, связанных с заданными пользователем преобразованиями между моделями, для более четкого разделения аспектов жизненного цикла. Новые средства проектирования IBM поддерживают как последнюю версию этого языка - UML 2.0, так и более ранние реализации.

Для поддержки инструментальных средств и платформ, не вошедших в пакет IBM Rational Software Architect, используются средства моделирования из пакета IBM Rational Suite. При совместном использовании пакетов IBM Rational Professional Bundle и IBM Rational Suite возможен импорт моделей из IBM Rational Rose в RSA для реализации моделей на платформе J2EE.

Возможности инструментального средства IBM Rational XDE Modeler включены в инструментальные средства нового пакета RSA. Однако, если в организации уже наработано большое количество скриптов для IBM Rational XDE Modeler, то, из-за возможной трудоемкости переноса имеющихся скриптов на новые инструментальные средства, может оказаться целесообразным продолжение использования IBM Rational XDE Modeler наряду с новой линейкой инструментальных средств RSA.

Кроме того, линейка средств моделирования IBM включает также IBM WebSphere® Business Integration Modeler (WBI Modeler) - средство моделирования бизнес-процессов, которое входит в пакет для бизнес-интеграции IBM WebSphere. WBI Modeler наряду с новыми средствами IBM Rational является частью интегрированной платформы разработки на основе Eclipse. По этой причине мы посчитали целесообразным также включить его описание в данный раздел.

IBM Rational Software Architect (RSA)

IBM Rational Software Architect представляет собой новую линейку инструментальных средств проектирования и разработки приложений, порталов и сайтов, ориентированных на платформу J2EE. Это мощный и простой в работе инструмент, который объединяет все аспекты проектирования и разработки программного обеспечения на основе моделей UML. Этот продукт предназначен для анализа, проектирования, сопровождения и развития корпоративных приложений и включает все возможности создания J2EE, Web и web-служб, реализованные в Rational Application Developer for WebSphere Software, возможности которого описаны ниже.

Rational Software Architect построен на базе открытой и расширяемой платформы Eclipse, которая использует ряд открытых промышленных стандартов. Благодаря этому пользователи могут создавать приложения, оптимизированные для работы не только с технологиями IBM, но и других поставщиков.

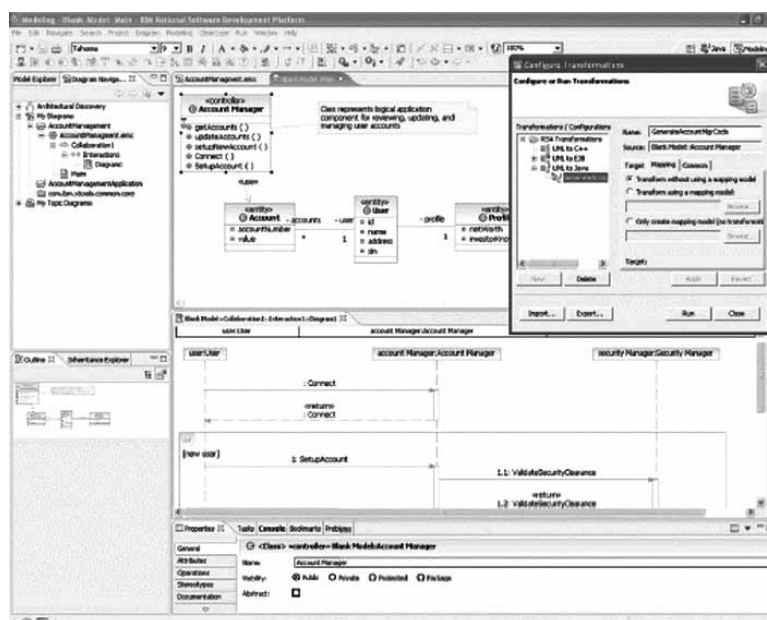
Основные преимущества IBM Rational Software Architect

Особенность	Функция	Преимущества для практика	Преимущества для бизнеса
Включает Rational Application Developer	Полная интегрированная среда разработки пользователя	Возможность спроектировать и внедрить мощную архитектуру IDE	Повышается продуктивность за счет использования всей функциональности
Возможности Rational Software Modeler	Включает такие возможности, как UML 2.0, OCL, упрощенные диаграммы, отчеты, шаблоны, повторное использование компонентов, поддержку команды, скрипты	Практики используют новейшие возможности масштабирования	Более полные и улучшенные модели улучшают архитектуру программного обеспечения и качество
Генерация и внедрение (harvesting) кода	Генерация кода на Java, J2EE и C++ на основе моделей UML. Берется существующий код и «внедряется» (harvest) модель	Повышается продуктивность за счет кодирования маршрутов и повторного использования существующего кода	Улучшение продуктивности команды и качества за счет повторного использования и автоматизации
Transformation Authoring	Полный механизм трансформации для генерации model-to-model и code-to-code в среде разработки на основе моделей	Повышается продуктивность за счет генерации специализированного кода, согласованности, например, с помощью подхода внедрения OMG MDA к разработке программного обеспечения	Сокращается цикл разработки и затраты на разработку
Структурный анализ Java	Анализируется код приложений Java для выявления повторяющихся элементов структуры, на основе которых оценивается возможность сопровождения	Повышается общее качество и согласованность разрабатываемого программного обеспечения	Улучшается сопровождение программного обеспечения, снижаются связанные с ним затраты

Разработка C/C++	Обеспечивается инфраструктура для разработки C/C++ в Software Architect	Знакомая среда для разработки приложений Java и C/C++	Сокращаются затраты /разработчики могут использовать старый опыт
Моделирование Physical ER	Используются обозначения IE и IDEF, в дополнение к UML, для упрощения разработки и понимания приложений без данных	Разработчики базы данных могут работать на более высоком уровне абстрагирования при создании/редактировании баз данных	Повышается продуктивность команд
Визуализация Java method body с помощью диаграмм Sequence	Используются конструкции диаграмм UML для понимания схемы методов Java	Легче понять существующий код	Повышается продуктивность команд
Конфигурация RUP для Software Architect	При работе пользователя с инструментом правила обработки и помощь пользователю обеспечивается динамически	Лучшие руководства по разработке программного обеспечения и best practices всегда под рукой	Лучшие приложения

Новый уровень продуктивности разработки приложений. Rational Software Architect содержит новые функции, которые упрощают освоение продукта и дальнейшую работу с ним, поднимая на качественно новый уровень продуктивность разработки как на основе моделей, так и на основе кода. Большой выбор типов диаграмм оказывает ощутимую помощь в выполнении задач проектирования и документирования. Редакторы web-диаграмм и Page Designer помогают создавать богатые возможностями web-приложения. RSA предоставляет возможности визуализации структуры и поведения приложений. Росту продуктивности моделирования способствует также автоматизация разработки шаблонов и преобразования моделей.

Эти современные возможности моделирования помогают настроить инструмент на конкретные потребности. В сочетании с тесной интеграцией между возможностями проектирования и разработки, а также с другими аспектами управления жизненным циклом разработки, Rational Software Architect повышает продуктивность разработки приложений.



Rational Software Architect

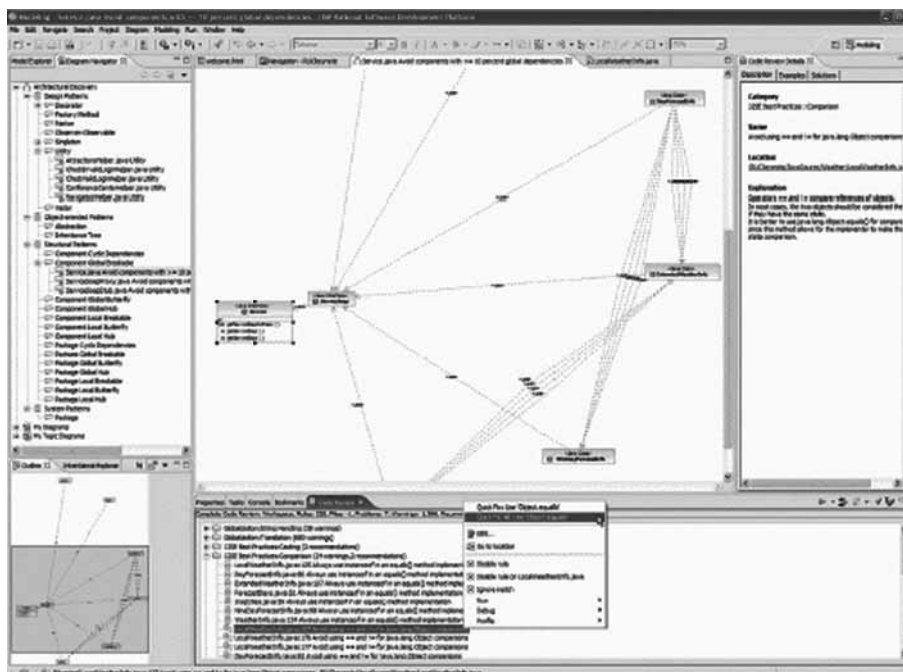
Анализ и контроль структуры приложений Java. Качественное моделирование приложений помогает создать хорошо спроектированный код. Но архитекторы и разработчики часто начинают свою работу с существующего кода. Здесь им требуется быстро проанализировать структуру и поведение приложения, перед тем как приступить к новым разработкам. В существующих приложениях после модификации исходного кода часто обнаруживаются проблемы с производительностью или нежелательные побочные эффекты. Эти проблемы часто возникают вследствие того, что в ходе появляются новые зависимости, и архитектура рушится.

Rational Software Architect содержит функции структурного анализа и контроля, которые прямо направлены на поиск проблем в коде Java. Шаблоны проектирования, реализованные в коде, можно визуализировать, даже если разработчики не знали об их существовании, вследствие чего возникают возможности для повторного использования компонентов. Rational Software Architect также автоматически обнаруживает структурные «антишаблоны» -- нежелательные, но распространенные проблемы программного обеспечения, осложняющие его поддержку и обновление.

Функции анализа и структурного контроля кода в Rational Software Architect позволяют быстро обнаруживать и исправлять потенциальные проблемы с функциональностью, масштабируемостью и возможностью обслуживания, возникшие вследствие нежелательных зависимостей, добавленных в ходе реализации.

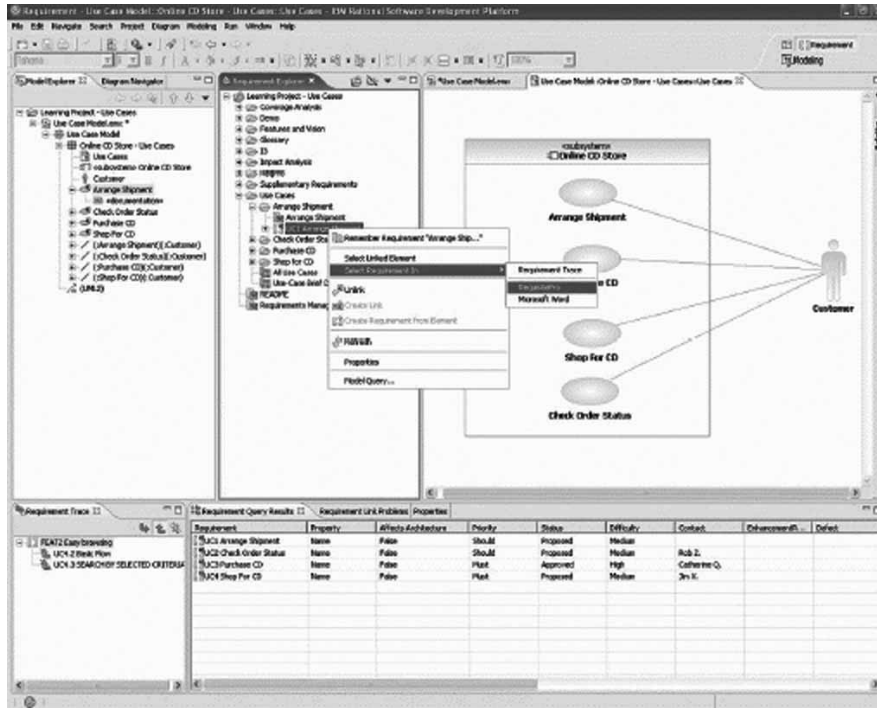
Упрощение инструментария для проектирования и разработки. Каждой организации, занимающейся разработкой программного обеспечения, необходимы инструментальные средства разработки. Для большинства современных проектов по разработке ПО минимальными инвестициями будут капиталовложения в интегрированную среду разработки (IDE). Некоторые IDE поставляются вместе с расширенным набором инструментов помимо обычных редактора, компилятора и отладчика – например, добавляются профилировщики и мастера диаграмм. Однако средства полной поддержки моделирования и разработки на основе моделей, как правило, приобретаются и устанавливаются отдельно от IDE, вследствие чего образуется сложная в настройке среда проектирования и разработки. Кроме того, не всегда эти инструменты интегрированы и поддерживают совместную работу.

Rational Software Architect упрощает среду разработки и проектирования. Поскольку продукт включает в себя все возможности Rational Application Developer for WebSphere Software - среды IDE, вы получаете в свое распоряжение полностью интегрированную среду проектирования и разработки в одном дистрибутиве, устанавливаемую одновременно. Это упрощает оценку, приобретение и интеграцию всех необходимых компонентов среды разработки ПО. Кроме того, Rational Software Architect устанавливается на платформах Windows и Linux, что еще больше упрощает ваш инструментарий если вы работаете с несколькими платформами.



Проектирование в Rational Software Architect

Интеграция с другими аспектами управления жизненным циклом разработки. Интеграция проектирования и разработки повышает уровень контролируемости связей между кодом и наиболее тесно связанными с ним артефактами. Однако для сложных проектов необходима контролируемость в масштабах жизненного цикла. При изменении требований в ТЗ архитекторам нужно знать, какая часть архитектуры затронута этими изменениями. Для таких проектов также требуется управлять изменениями, влияющими и на модель, и на файлы кода. Все эти задачи в комплексе очень трудны даже при условии идеального управления процессом разработки ПО. Неудача в решении этих проблем повышает риск провала проекта в целом.



Связь моделирования с другими аспектами управления жизненным циклом в Rational Software Architect

Rational Software Architect помогает осуществить интеграцию с другими аспектами управления жизненным циклом. Можно обращаться к требованиям технического задания, хранящимся в Rational RequisitePro, выбирать из них те, которые связаны с соответствующими элементами моделирования, и выполнять синхронизацию по выбранным пользователем правилам. Пользователи могут генерировать отчеты, освещающие контролируемость связей между требованиями ТЗ и проектированием. Файлами моделирования можно управлять с помощью Rational ClearCase LT - мощного продукта для управления конфигурацией, который поставляется вместе с Rational Software Architect. В качестве альтернативного варианта пользователи, работающие с инструментом Concurrent Versions System (CVS), могут интегрировать продукт с этой системой. А интеграция с IBM Rational Unified Process (RUP) предоставляет группам разработчиков возможность общего, интерактивного и интегрированного управления процессом.

Rational Software Architect интегрируется с этими и другими компонентами платформы организации групповой работы IBM Rational, обеспечивая управление техническим заданием, контролируемость, контроль исходного кода и другие функции управления работой группы в течение жизненного цикла разработки. Эта интеграция снижает риск, связанный с разработкой программного обеспечения, и повышает предсказуемость разработки.

Rational Software Architect входит в состав пакета IBM Rational Professional Bundle, но может поставляться и как отдельный программный продукт.

IBM Rational Software Modeler (RSM)

IBM® Rational® Software Modeler (RSM) – это визуальный инструмент моделирования и проектирования, ориентированный на создание приложений на платформе J2EE.

Rational Software Modeler поддерживает стандартизованный язык моделирования Unified Modeling Language (UML), включая как его последнюю версию UML 2.0, так и более ранние реализации.

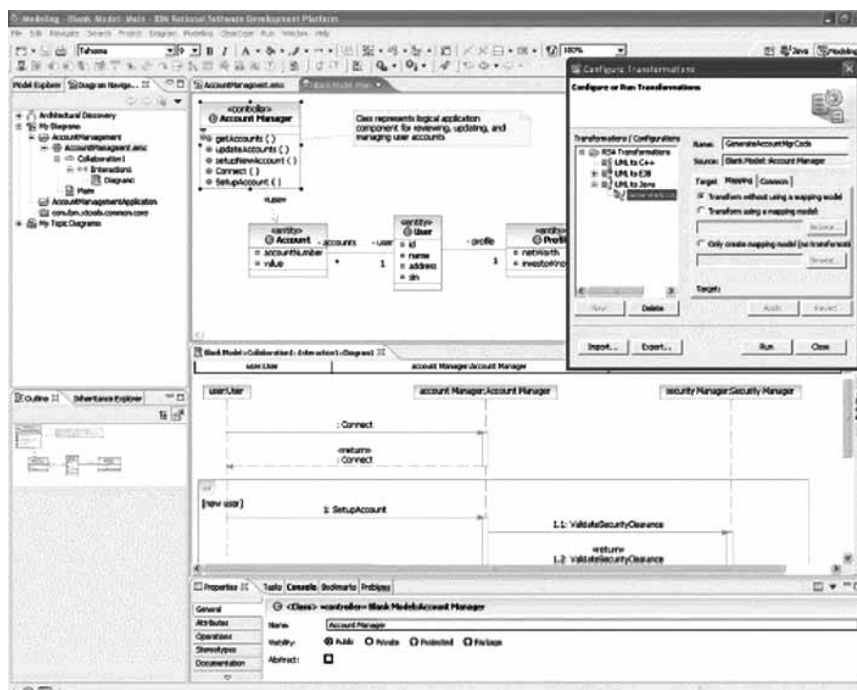
Основанный на расширяемой технологии Eclipse и открытых стандартах, Rational Software Modeler представляет собой открытый, в высшей степени расширяемый и настраиваемый инструмент, поддерживающий моделирование в масштабе предприятия. В тесной интеграции с другими средствами управления жизненным циклом разработки, Rational Software Modeler упрощает анализ и проектирование, способствует упрощению работы и повышению ее производительности.

Основные преимущества IBM Rational Software Modeler

Особенность	Функция	Преимущества для практика	Преимущества для бизнеса
Основан на Eclipse 3.0	Пользователи могут применять открытую и расширяемую платформу Eclipse	Работа в единой хорошо знакомой командной среде разработки	Выше продуктивность за счет использования единой платформы для команды разработчиков программного обеспечения
Моделирование UML 2.0	Поддержка анализа и дизайна с помощью диаграмм Use Case, Class, Sequence, State Machine, Communication, Component и Deployment diagrams	Использование новейших возможностей UML повышает точность дизайна	Более полные усовершенствованные модели улучшают архитектуру и качество программного обеспечения
Поддержка OCL	Определения архитектурных ограничений модели с помощью Object Constraint Language (OCL)	Привильное использование модели и уменьшение ошибок	Выше качество программного обеспечения
Упрощение составления диаграмм	Легкое создание диаграмм с помощью диаграмм свободных форм, тематических диаграмм и диаграмм просмотра	Экономия времени и повышение продуктивности моделирования	Улучшается документирование дизайна и сокращаются затраты рабочего времени на подготовку документации
Генерация отчетов	Генерация отчетов HTML и PDF из файлов моделей UML	Легкий и быстрый доступ к нужной информации	Упрощается доступ к информации модели при оперативной оценке проекта
Интеграция с Rational RequisitePro	Просмотр требований в RequisitePro и создание ссылок на элементы модели	Легкий и быстрый доступ к требованиям для ссылок на модель	Улучшается работа в команде и уменьшается число ошибок проекта из-за неучтенных требований
Применение и создание шаблонов	Законченный функционал для сбора, совместного использования и применения специальных знаний дизайнера	Повторное использование хорошо известных технологий и подходов дизайнера повышает продуктивность и обеспечивает согласованность	Повышается продуктивность команды и дизайнеры лучше согласованы и более надежны

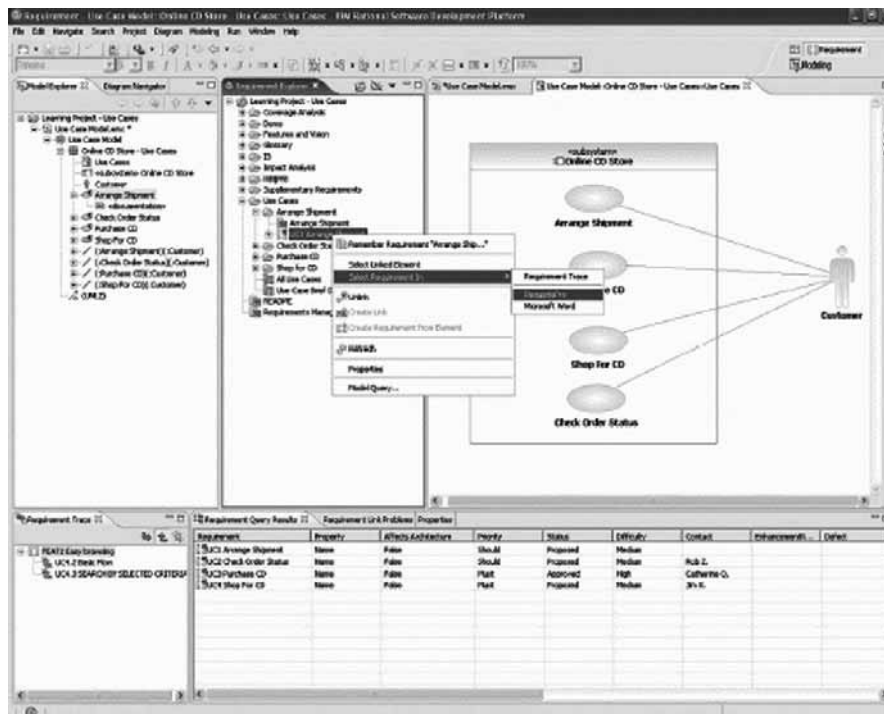
Asset Browser	Легкий доступ к повторно используемым компонентам	Повторное использование повышает продуктивность качества. Применение готовых протестированных компонентов гарантирует, что их ошибки приведут к новым дефектам.	Экономия средств компании за счет повторного использования
Поддержка скриптов	Открытие API для поддержки настройки и расширения среды моделирования	Настройка продукта по мере необходимости	Возможности настройки и расширенное применение продукта улучшают продуктивность и полезность
Поддержка команд	Возможность работы с несколькими моделями, сравнение методов слияния и защита плодов работы с помощью управления конфигурацией	Легче работать и сотрудничать с другими участниками команды	Защита разработок команды и тесное сотрудничество в команде

Новый уровень продуктивности моделирования. Rational Software Modeler содержит целый ряд новых функций, облегчающих работу с продуктом и повышающих производительность моделирования, включая средства автоматизации разработки шаблонов и преобразования моделей, а также большой выбор типов диаграмм для выполнения задач проектирования и документирования.



Работа с IBM Rational Software Modeler

Интеграция с другими аспектами управления жизненным циклом разработки. Для сложных проектов необходимо обеспечение контроля в масштабах всего жизненного цикла. Так, при изменении требований в спецификациях, архитекторам нужно знать, какая часть архитектуры затронута этими изменениями. Комплексное управление изменениями, влияющими на модель и на другие артефакты жизненного цикла, представляет собой довольно трудную задачу даже при условии идеального управления изменениями и в значительной степени определяет успех проекта в целом.



Связь моделирования с другими аспектами управления жизненным циклом в Rational Software Modeler

Rational Software Modeler помогает осуществить интеграцию моделирования с другими аспектами управления жизненным циклом, снижая риски, связанные с разработкой систем, и повышая предсказуемость моделирования. Можно обращаться к требованиям технического задания, хранящимся в Rational RequisitePro, выбирая из них те, которые связаны с соответствующими элементами моделирования, и выполнять синхронизацию по выбранным пользователем правилам. Пользователи могут создавать отчеты, отражающие связи между требованиями и проектированием. Файлами моделирования можно управлять с помощью Rational ClearCase LT – специализированного продукта для управления конфигурацией, который поставляется вместе с Rational Software Modeler (в качестве альтернативы возможно использование инструмента Concurrent Versions System (CVS), также интегрированного с RSM).

Rational Software Modeler входит в состав пакета IBM Rational Professional Bundle.

IBM Rational Rose

IBM Rational Rose — популярное средство визуального моделирования, которое считается стандартом де-факто среди средств визуального проектирования приложений. Этот продукт входит в состав пакета IBM Rational Suite и предназначен для моделирования программных систем с использованием широкого круга инструментальных средств и платформ. Инструментальное средство IBM Rational Rose расширяет возможности моделирования программных систем, выходящих за рамки платформы J2EE и инструментальных средств моделирования в составе IBM Rational Professional Bundle.

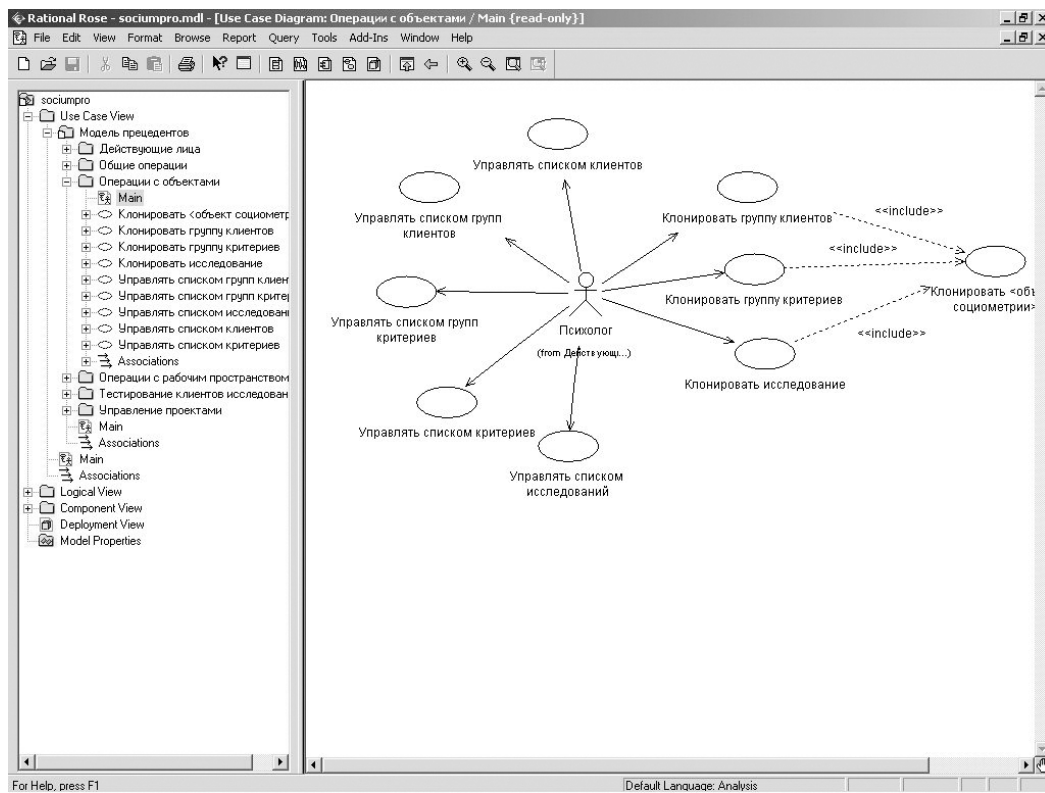
Являясь простым и мощным решением для визуальной разработки информационных систем любого класса, Rational Rose позволяет создавать, изменять и проверять корректность модели.

Rational Rose объединяет команду разработчиков на базе универсального языка моделирования UML, который определяет стандартную графическую символику для описания архитектуры ПО. Любые участники проекта — аналитики, специалисты по моделированию, разработчики и другие — могут использовать модели, построенные в Rational Rose, для большей эффективности создания конечного продукта.

Для бизнес-аналитиков средство Rational Rose дает возможность детально описать и проанализировать бизнес-процессы данной предметной области. Системные аналитики, используя указанные описания, смогут разработать необходимый функционал ИС, который максимально удовлетворит запросы заказчика. Для архитекторов средство Rational Rose будет полезно при создании мощной и гибкой архитектуры системы. Для аналитиков, специализирующихся в области

разработки баз данных, Rational Rose даст возможность визуально проектировать и генерировать базы данных любого размера. Таким образом, можно создавать базы данных Microsoft SQL Server, Oracle, Sybase SQL Anywhere, IBM DB2 и любые другие, которые поддерживают возможность запуска скриптов стандарта ANSI SQL.

Существуют расширения Rational Rose, которые позволяют выполнять скелетную (round-trip) разработку ИС, создаваемых на базе языков C/C++, Java, Smalltalk, Ada, Object Pascal (Borland Delphi) и др. Таким образом, можно сгенерировать каркас программного кода на любом из указанных языков или выполнить процедуру обратного проектирования, что позволяет сформировать модель на базе существующего кода.



Работа с Rational Rose

Возможность публикации модели в Интернете служит основой для объединения работы удаленных команд разработчиков.

Существует вариант Rational Rose RealTime, полезный при разработке систем реального времени. Данное средство включает расширенную нотацию визуального моделирования и отвечает всем требованиям разработчиков указанного направления. Rational Rose RealTime поддерживает визуальную разработку конструкций реального времени, генерацию программного кода и исполнение моделей на протяжении всего жизненного цикла.

Rational Rose предоставляет огромные возможности для проектирования ИС с использованием лучших архитектурных принципов. С помощью инструмента можно визуализировать, анализировать и уточнять требования к создаваемому продукту. Способности описывать графический интерфейс отдельно от бизнес-логики приводят в конечном итоге к лучшим результатам на уровне всего проекта. Всегда лучше и эффективнее лишний раз проанализировать, что должно быть сделано, чем каждый раз заниматься трудоемкими переделками и подгонкой кода. Использование единого инструмента моделирования на протяжении всего жизненного цикла разработки помогает создавать "идеальную" систему.

Rational Rose предлагает плавный процесс разработки ИС. Любые модели, создаваемые с помощью данного средства, являются взаимосвязанными: бизнес-модель, функциональная модель, модель анализа, модель проектирования, модель базы данных, модель компонентов и модель физического развертывания системы.

Возможности по созданию и использованию шаблонов архитектурных решений позволяют эффективно использовать опыт, накопленный в предыдущих проектах.

Rational Rose является ведущим инструментом визуального моделирования в программной индустрии, благодаря полноценной поддержке UML и многоязыковой поддержке командной разработки. Инструмент полностью поддерживает компонентно-ориентированный процесс создания ИС.

Интеграция Rational Rose с Rational RequisitePro позволяет на базе визуальной модели разработать полный набор требований, которые необходимо реализовать при создании конечного продукта.

Интеграция Rational Rose с Rational TestManager позволяет создавать сценарии тестирования на базе визуальной модели.

Интеграция Rational Rose с Rational ClearCase позволяет поставить на версионный контроль модель целиком или по частям.

Интеграция Rational Rose с Rational SoDA позволяет автоматизировать процесс создания документов и отчетов по визуальной модели.

IBM Rational XDE Modeler

IBM Rational XDE Modeler - средство визуального моделирования информационных систем на основе UML v1.4, которое входит в пакет IBM Rational Suite.

Возможности IBM Rational XDE Modeler покрываются средствами нового пакета IBM Rational Software Architect. Однако, если в организации уже наработано большое количество скриптов для IBM Rational XDE Modeler, то, из-за возможной трудоемкости переноса имеющихся скриптов на новые инструментальные средства, может оказаться целесообразным продолжать использования IBM Rational XDE Modeler наряду с новой линейкой инструментальных средств RSA.

IBM Rational XDE Modeler позволяет архитекторам и проектировщикам организовать управляемую моделью разработку с использованием UML. Пользователи могут создавать независимые от платформ модели программной архитектуры, бизнес-требований, повторно используемых информационных ресурсов и коммуникационных средств управленческого уровня. Поддержка UML, ставшего промышленным стандартом, а также мощный механизм работы с шаблонами позволяет пользователям создавать семантически насыщенную прикладную архитектуру, соответствующую бизнес-требованиям и легко понимаемую коллективом разработчиков.

Архитекторы и проектировщики могут использовать поддержку множественных моделей в Rational XDE Modeler для реализации парадигмы управляемой моделью архитектуры (MDA).

Разработчики могут использовать архитектурные модели и шаблоны как основу для реализации, ускорив, таким образом, разработку приложений в строгом соответствии с предоставленной архитектурой. Кроме того, такие возможности, как моделирование в свободной форме, генерация отчетов и публикация в Web-среде позволяют организовать совместную работу пользователей и заинтересованных лиц над архитектурными и проектными решениями, независимо от того используют ли заинтересованные лица Rational XDE.

Rational XDE Modeler включает в себя расширенные возможности работы с шаблонами, что позволяет программным архитекторам с помощью автоматических трансформаций аналитических моделей контролировать проектные решения и их реализацию. Подобная автоматизация также позволяет повысить эффективность архитектуры за счет ее непротиворечивости и воспроизводимости.

Основные возможности Rational XDE Modeler:

- управляемая моделью разработка с поддержкой языка моделирования UML версии 1.4;
- поддержка множественных моделей, позволяющая использовать архитектуру MDA (Model-Driven Architecture — управляемая моделью архитектура);
- возможность использования либо в виде отдельного приложения, либо интегрировано в среде
- IBM WebSphere или Microsoft Visual Studio .NET;
- возможность реализации архитектур как на платформе .NET, так и на платформе J2EE;
- создание нейтральных к платформам моделей и диаграмм на любом уровне абстракции;
- определяемые пользователем шаблоны, предназначенные для создания пользовательской настройки и повышения эффективности шаблонов архитектурного проектирования;
- перекрестные ссылки для разнообразных моделей и ведение различных версий вплоть до уровня класса и диаграммы, позволяющие структурировать любой проект;
- поддержка тесной интеграции между аналитическими моделями и моделями компонентов;
- моделирование в свободной форме;
- публикация и генерация отчетов в Web.

Архитекторы начинают свою работу с использования шаблона модели, который предоставляет структуру для их проекта. Они могут воспользоваться либо стандартными шаблонами, либо быстро и просто создать свои собственные шаблоны. Встроенный в Rational XDE Modeler механизм работы с шаблонами позволяет создавать шаблоны на основе уже существующих моделей, с возможностью произвольных дальнейших модификаций шаблона под текущую задачу.

Автоматизированные шаблоны позволяют архитекторам быстро начать разработку, полностью использовать имеющиеся творческие способности, избежать выполнения повторяющихся рутинных задач, а также проектировать свои собственные шаблоны.

Используя Rational XDE Modeler, архитекторы получают исчерпывающее представление о влиянии изменений на проектные решения. При этом отслеживаются не только уже сделанные изменения и история развития проектных решений, но также и влияния предстоящих изменений.

Кроме того, архитекторы могут быстро оценить "глобальное" влияние предложенных архитектурных улучшений на будущие версии и редакции ПО. Без использования Rational XDE Modeler рассмотрение столь широкой области воздействия было бы затруднительно.

IBM Rational XDE Modeler позволяет пользователям создавать модели и диаграммы на любом уровне абстракции, начиная от диаграмм в свободной форме для управленческого звена и заканчивая подробными проектными моделями, предназначенными для преобразования в программный код.

Rational XDE Modeler интегрируется с Rational RequisitePro, что позволяет просматривать и управлять постоянно изменяющимися требованиями. Также предоставляются возможности публикации в среде web, наряду с другими возможностями генерации отчетов, что позволяет обмениваться информацией с другими сотрудниками, независимо от их участия в проекте.

Rational XDE Modeler интегрируется и с IBM Rational ClearCase, что позволяет контролировать версии приложений непосредственно из IDE-среды.

RUP предоставляет конфигурируемую инфраструктуру процессов создания ПО, которая включает специальные конфигурации RUP для разработчиков на Java и .NET, а также специальный модуль для реализации техники экстремального программирования (XP).

IBM WebSphere Business Integration Modeler (WBI Modeler)

IBM WebSphere® Business Integration Modeler (WBI Modeler) представляет собой средство моделирования бизнес-процессов и включает средства для проектирования, тестирования и совершенствования комплексных бизнес-процессов, а также средства для организации коммуникации между ними. WBI Modeler используется для создания карты существующих бизнес-процессов и их дальнейшего моделирования и оптимизации в целях управления компанией как единым целым.

Решения WBI Modeler построены на базе платформы Eclipse и стали первыми реализациями нового поколения средств моделирования WebSphere, интегрированными с другими инструментами Eclipse, такими как, IBM Rational Application Developer, IBM Rational, IBM WebSphere MQ Workflow (ранее известный как IBM MQSeries Workflow) и программным обеспечением IBM Tivoli.

WBI Modeler имеет в своем составе следующие средства моделирования бизнес-процессов, обеспечивающие высокий уровень их гибкости:

- WBI Modeler Entry Edition – средство моделирования начального уровня, настраиваемое в соответствии с потребностями индивидуальных пользователей и позволяющее растущим компаниям решить задачи формализации бизнес процессов при более низких затратах в расчете на одного пользователя
- WBI Modeler Advanced Edition включает расширенные средства моделирования для предприятий с протяженными бизнес-процессами, а также инструменты для более сложного моделирования и оптимизации, которые помогают анализировать эффективность выполнения бизнес-процессов и потенциальные бизнес-результаты еще до реального выделения ресурсов на разработку и развертывание

Оба варианта WBI Modeler - Advanced и Entry - поддерживают три разных профиля пользователя, а расширенная версия и три операционных режима.

Выбор профиля пользователя. В зависимости от потребностей, пользователи имеют возможность выбрать базовый, промежуточный или продвинутый профиль для моделирования бизнеса, каждый

из которых обеспечивает свой уровень детализации и сложности информации, используемой при моделировании процессов. Пользователи могут легко переключаться между этими профилями, используя преимущества тех или иных средств.

- Базовый профиль моделирования бизнеса (Basic) является входной точкой для таких бизнес-пользователей, как бизнес-аналитики, проектировщики и эксперты и позволяет им легко и быстро фиксировать и моделировать последовательности операций, сложную логику и бизнес-правила.
- Промежуточный профиль моделирования бизнеса (Intermediate) позволяет пользователям с более глубокими техническими знаниями - разработчикам архитектуры бизнеса и бизнес-процессов - определять и просматривать дополнительные детали моделей процессов и данных, включая выражения и связи.
- Расширенный профиль моделирования бизнеса (Advanced) обеспечивает наиболее подробный уровень детализации для моделей процессов и данных и предназначен для разработчиков решений и ИТ-архитектуры, которые сконцентрированы на развертывании моделей в среде выполнения.

Выбор технологического режима. WBI Modeler позволяет пользователям выбрать один из трех разных технологических режимов:

- Операционный – доступен в начальной и расширенной версиях, является заданным по умолчанию режимом работы и обеспечивает наибольший уровень детализации среди трех технологических режимов.
- Business Process Execution Language (BPEL) – оптимизирован для моделей экспорта в файловые форматы BPEL, Web Services Description Language (WSDL) и XML Schema Definition Language (XSD), а также для импорта из IBM WebSphere Studio Application Developer Integration Edition, обеспечивая разработку приложений на основе моделей.
- MQ Workflow Flow Definition Language (FDL) – оптимизирован для моделей экспорта в формат FDL, позволяет компаниям быстро начать работу, используя IBM WebSphere MQ Workflow в качестве решения для создания автоматизированного потока операций.

Оба решения – WBI Modeler Advanced Edition и WBI Modeler Entry Edition – предоставляют репозиторий для централизованного хранения правил, процедур, графиков и другой важной корпоративной информации, что обеспечивает ее непротиворечивость и доступность в рамках всей компании. Объединение этой информации с бизнес-процессами крайне важно для точного представления всех факторов, влияющих на результат каждого из процессов. В результате становится возможным идентифицировать важные корпоративные данные и правила, которые определяют эффективность процессов, что, в свою очередь, помогает оценить степень готовности к автоматизации.

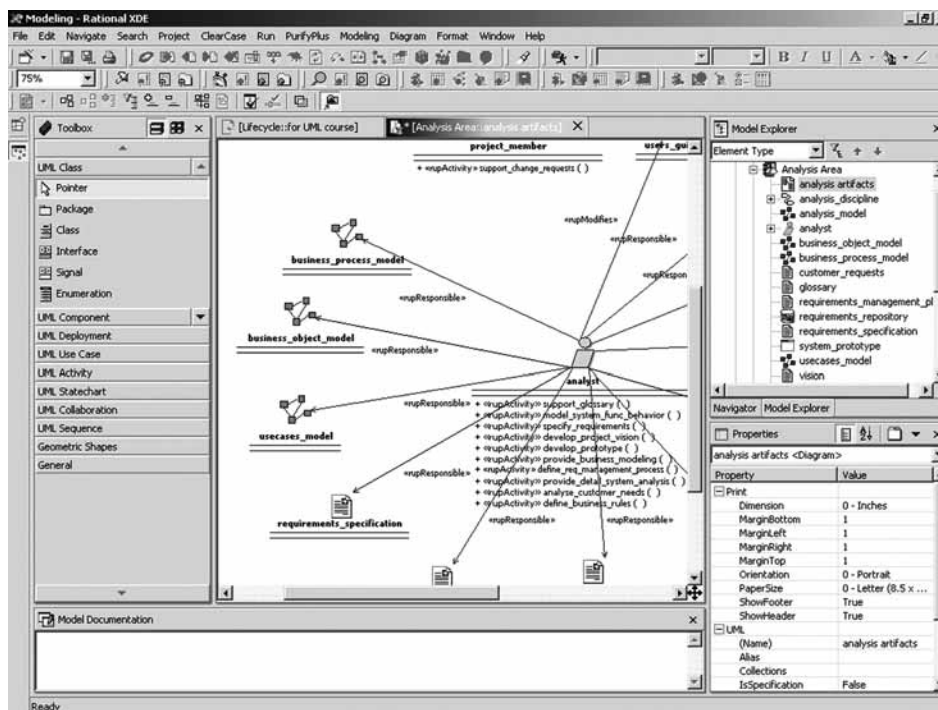
WBI Modeler Advanced Edition и WBI Modeler Entry Edition также имеют в своем составе инструменты для динамического анализа, которые помогают оптимизировать моделирование бизнес-процессов.

Возможности компоновки для просмотра моделей. Решения WBI Modeler Advanced Edition и WBI Modeler Entry Edition позволяют просматривать модели либо в свободной компоновке, либо в виде секций (swimlane). Свободная компоновка предполагает использование всего диапазона средств редактирования, тогда как альтернативный вариант компоновки отображает присвоение операций процесса конкретным ролям, ресурсам или организационным единицам.

После ввода моделей к ним применяются правила проверки достоверности, которые позволяют провести простое тестирование документации и зафиксировать этапы процесса. С помощью WBI Modeler Advanced Edition можно также провести более глубокое тестирование с целью прогнозирования общей продолжительности работ и предполагаемых затрат, ожидаемого отставания и т.д. Эти возможности появляются при моделировании сценария с определенной моделью. Сравнивая ожидаемые расходы на существующие процессы с затратами для оптимизированных процессов, легко подготовить анализ окупаемости инвестиций (ROI).

Экспорт и импорт данных. WBI Modeler Advanced Edition позволяет быстро начать развертывание оптимизированных бизнес-процессов благодаря возможностям экспорта их определений в IBM WebSphere MQ Workflow, V3.5, IBM WebSphere Business Integration Server (для обоих продуктов при помощи экспорта FDL) и WebSphere Business Integration Server Foundation (при помощи экспорта BPEL). Импорт FDL и существующих моделей WebSphere Business Integration Workbench, V4.2.4 также позволяет многократно использовать существующие бизнес-данные, обеспечивая сокращение издержек. Для случаев, когда требуется усовершенствование бизнес-модели для создания приложений, WBI Modeler Advanced Edition позволяет экспортировать язык Unified Modeling Language

(UML) в программное обеспечение IBM Rational® XDE™ и Rational Software Architect. Кроме того, WebSphere Business Integration Modeler Advanced Edition поддерживает моделирование в командной среде с использованием интегрированной среды разработки (IDE) Eclipse для моделей процессов с механизмом выписки и возврата элементов.



Экспорт UML модели в Rational XDE

IBM WebSphere® Business Integration Modeler входит в состав программного пакета для бизнес-интеграции IBM WebSphere, который представляет собой современное связующее ПО, объединяющее приложения, пользователей и бизнес-процессы в единый механизм.

Средства разработки

Основные цели разработки ПО:

- определение структуры кода на основе реализуемых подсистем, организованных по уровням;
- реализация классов и объектов в виде модулей (исходных, двоичных, исполняемых файлов и др.);
- тестирование разработанных модулей;
- интеграция результатов работы отдельных программистов (или групп) в рабочую систему.

Для ведения непосредственной разработки IBM Rational предлагает целый набор инструментария, описанного далее.

IBM Rational Application Developer

IBM Rational Application Developer for WebSphere Software – это средство быстрого проектирования, разработки, анализа, тестирования, профилирования и развертывания Web-приложений, Web-служб, Java, J2EE и порталов в среде IDE на базе Eclipse.

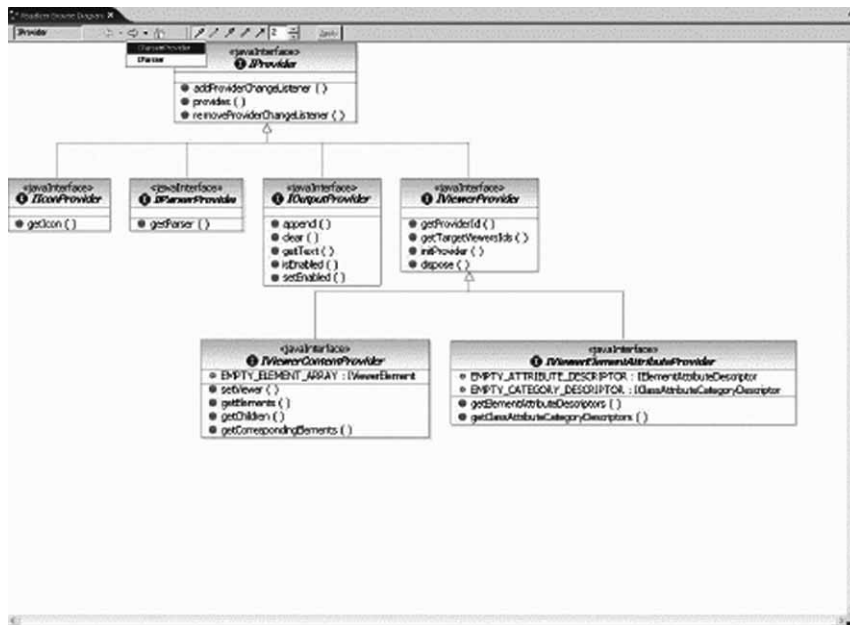
В условиях ведения бизнеса “по требованию” разработчики должны быстро реагировать на непрерывно изменяющиеся запросы. Для этой цели им необходима среда разработки, которая повышала бы производительность труда, сводила бы к минимуму длительность обучения и сокращала бы время разработки и тестирования. IBM Rational Application Developer for WebSphere Software – это всеобъемлющая интегрированная среда разработки, обеспечивающая полную поддержку модели программирования J2EE, включая Web, Java, Web-службы и EJB, которая позволяет значительно ускорить разработку приложений. Rational Application Developer включает в себя интегрированные средства разработки порталов, визуального редактирования UML, анализа кода, автоматизированного тестирования и развертывания – все, что необходимо разработчикам для производительной работы и для того, чтобы полученный код был хорошо спроектирован, масштабируем и готов к промышленной

эксплуатации. Встроенные инструменты контроля версий и поддержки групповой работы позволяют разработчикам сложных проектов и большим командам координировать контроль версий и обеспечить защиту ресурсов группы.

IBM Rational Application Developer for WebSphere Software оптимизирован для программного обеспечения WebSphere и предоставляет возможности разработки для других технологических платформ.

Основные преимущества IBM Rational Application Developer

Особенность	Функция	Преимущества для практика	Преимущества для менеджера проекта	Преимущества для бизнеса
Надежная поддержка EJB	Генерация Sesion bean facade, Unit test EJBs, теги метаданных Xdoclet в классах Java classes для развертывания EJB, легкое отображение EJB в базу данных	Выше продуктивность и качество	Выше продуктивность и качество	Сокращение затрат повышение эффективности
Интеграция UML Vizualization и поддержка редактирования	Создание диаграмм классов UML, диаграммы Sequence для структуры Java/EJB, базы данных и тематических диаграмм	Легкая визуализация сложных приложений и возможность легко понять их структуру	Повышается продуктивность разработчика	Сокращение затрат повышение эффективности
Поддержка J2EE и гибкость	Намного упрощается развертывание J2EE и поддерживается несколько уровней J2EE	Выше продуктивность и качество, улучшается эффективность	Выше продуктивность и качество, улучшается эффективность	Сокращение расходов, повышается эффективность и гибкость
Разработка порталов и портлетов	Утилиты Visual Portal для создания сайта и страниц, разработка и просмотр Visual Portlet и среда разработки Integrated Portal	Значительно облегчается разработка порталов и портлетов	Повышается продуктивность, быстрее вывод на рынок	Сокращение затрат, быстрее выводятся на рынок порталные решения
Поддержка анализа кода	Анализ статичного кода и динамический анализ в реальном времени. Обнаружение утечек памяти Анализ нарушения правил кодирования программ.	Значительно упрощается поиск скрытых проблем.	Повышается качество и производительность приложений	Повышается производительность приложений, меньше простоев, выше удовлетворенность пользователей.
Профилирование приложение	Измерение производительности приложений, в том числе между серверами	Предотвращение проблем с производительностью	Повышается продуктивность и производительность	Повышается производительность приложений, меньше простоев, выше удовлетворенность пользователей



Моделирование структуры приложения в RAD

Полная поддержка модели программирования J2EE. Параллельная поддержка проектов J2EE версий 1.2 , 1.3 и 1.4, включая полную поддержку следующих средств: Enterprise JavaBeans™, EJB Query Language (EJB QL), Container-Managed Persistence (CMP), Web-архивы (WAR) и корпоративные архивы (EAR). Включает в себя среды тестирования модулей для нескольких конфигураций IBM WebSphere Application Server, чтобы обеспечить поддержку проектов с различными тестовыми средами.

Расширение возможностей программирования Java. Application Developer включает в себя большой набор инструментов для создания приложений Java и поддерживает Java Development Kit версии 1.4.2. Visual Editor for Java позволяет разрабатывать GUI для Java методом перетаскивания и вставки, с помощью компонентов AbstractWindow Toolkit (AWT), SWT или Swing, а также позволяет визуально привязывать данные к виджетам UI и немедленно просматривать изменения вида GUI благодаря динамическому механизму обновления для исходного кода и визуального дизайна. Unified Modeling Language (UML) Visual Editor содержит графические редакторы в качестве альтернативного способа визуализации и редактирования кода Java и таблиц данных при помощи стандартных диаграмм UML, а также помогает вам лучше понимать сложный код и управлять им.

Интеграция приложений с помощью web-служб и архитектур, ориентированных на службы. Rational Application Developer содержит инструменты, необходимые для обнаружения, создания, сборки, тестирования, развертывания и публикации web-служб. Вы можете создавать новые web-службы с нуля или добавлять в существующие приложения совместимые с WS-I web-службы, чтобы упростить интеграцию и повторное использование:

- автоматическая генерация файлов WSDL и WSIL, необходимых для описания web-служб и подготовки их к инспекции на основе имеющихся материалов, либо разработка файла WSDL и генерация кода Java;
- простота создания, проверки и обнаружения совместимых web-служб, соответствующих WS-I, с помощью редактора WSDL и браузера web-служб;
- создание Java-прокси для взаимодействия с клиентскими приложениями;
- генерация тестовых клиентов web-служб без кодирования;
- занесение в реестр UDDI версии 2

Упрощение разработки XML. Полный набор функций XML в составе Application Developer позволяет разработчикам создавать, редактировать и преобразовывать документы XML:

- визуальные инструменты для создания и просмотра файлов XML и для автоматической проверки документов на соответствие определениям типа документа (DTD);
- визуальные инструменты для преобразования элементов XML в таблицы реляционной базы данных;
- тесная интеграция с IBM DB2 для создания документов XML из существующих данных DB2 или разбора документов XML в данные DB2;
- инструменты для экономии времени при сборке и отладке сценариев преобразования Extensible Stylesheet Language (XSL) и для преобразования документов XML в другие форматы разметки, в том числе HTML;

Создание приложений на основе баз данных. Rational Application Developer содержит интегрированные инструменты для создания приложений баз данных из запросов, элементов Beans и EJB.

Мастер соединений с базой данных упрощает создание соединения Java Database Connectivity (JDBC) с СУБД, такими как IBM DB2 Universal Database, IBM Informix®, Oracle или Microsoft® SQL Server®.

Мастер запросов SQL и построитель запросов SQL предоставляют визуальный интерфейс для создания и выполнения операторов SQL. Можно создать простой запрос с помощью мастера запросов SQL или использовать построитель запросов SQL, который поддерживает более широкий набор операторов. Встроенный мастер преобразования SQL-XML позволяет создавать документы XML и XSL, спецификации DTD, схемы XSD, файлы HTML и связанные артефакты.

Поддержка SQL for Java (SQLJ) позволяет быстро создавать и отлаживать приложения, использующие хранимые процедуры SQLJ и DB2 SQLJ, а также повышает производительность доступа к данным для статических соединений SQL.

Упрощение объектно-реляционного соответствия. Обеспечение объектно-реляционного соответствия дается легко и просто благодаря поддержке методов сверху вниз, снизу вверх и от середины. Мастера позволяют легко создавать и тестировать компоненты EJB. Вы можете строить приложения, которые используют, извлекают и отображают нужные данные, отформатированные в соответствии с вашими потребностями. Rational Application Developer поддерживает отношения «многие ко многим», что помогает генерировать реализации со множественными связями.

Упрощение web-разработки и управления web-сайтами. Простые в работе мастера и инструменты объединяют практически все аспекты web-разработки (HTML, JSP, JSF и сервлеты) в единый интерфейс и позволяют разработчикам с различным характером технической подготовки и даже тем, кто не знаком с Java, строить многофункциональные приложения для работы с данными:

- визуальное проектирование и конструирование web-приложений с помощью Model-View-Controller и Struts 1.1;
- быстрая разработка многофункциональных пользовательских web-интерфейсов и web-форм с помощью многократно используемых, перетаскиваемых компонентов JavaServer Faces, которые генерируют код обработки событий, проверки пользовательского ввода и привязки данных для web-приложений;
- возможность подключения web-приложений к реляционным базам данных, компонентам EJB и web-службам с помощью простых инструментов, поддерживающих Service Data Objects, новый промышленный стандарт для доступа к гетерогенным данным;
- создание интерактивных web-интерфейсов пользователя с производительностью и характеристиками техобслуживания на уровне тонких клиентов с помощью клиентских компонентов Faces, расширяющих спецификацию JSF;
- быстрая разработка визуальных отчетов по данным с помощью встроенных инструментов Crystal Reports. Проектирование и разработка собственных отчетов или использование готовых шаблонов отчетов Компонент JSF Report Viewing позволяет внедрять отчеты в web-приложения и динамически исполнять их в Интернете;
- простая разработка многофункциональных web-страниц в режиме визуального редактирования или редактирования исходного кода с помощью богатыми возможностями редактора HTML и JSP;
- возможность быстро импортировать web-сайт целиком, изменять стили и применять их в глобальном масштабе, визуально добавлять и удалять страницы из древовидного структурного представления с помощью Web Site Designer;

- простое создание и анимация оригинальной графики с помощью компонентов WebArt Designer и Animated GIF Designer.1.

Кодирование на 4GL и генерация кода на Java. Enterprise Generation Language (EGL) – это язык четвертого поколения, который позволяет разработчикам процедур быстро создавать и отлаживать управляемые данными web-приложения и бизнес-логику с помощью знакомых программных конструкций и без написания кода на Java. EGL содержит конструкции и функциональные возможности Informix 4GL, благодаря чему Application Developer является идеальной средой разработки для Informix. EGL тесно интегрирован с JavaServer Faces, так что вы можете создавать интерактивные, управляемые данными web-приложения с помощью компонентов JSF.

Визуальная разработка портлетов и порталных приложений. Application Developer содержит набор визуальных инструментов разработки порталов и среду тестирования WebSphere Portal, что позволяет разрабатывать и тестировать отдельные портлеты и порталные приложения в целом. Новые портлеты создаются при помощи мастеров, которые генерируют структуру проекта портлета, совместимую с J2EE, и могут создать весь портлет. Мастера могут генерировать портлеты, соответствующие IBM Portlet API или же JSR 168, спецификации промышленного стандарта для агрегации, персонализации, представления и обеспечения безопасности портлетов. Можно спроектировать интерфейс портлета с помощью Page Designer и JavaServer Faces, а затем визуализировать структуру и потоки событий с помощью Web Diagram Editor и схемы Struts, чтобы облегчить поддержку приложений. Кроме того, можно сочетать портлеты с Java Server Faces, чтобы разрабатывать портлеты визуально с помощью компонентов Faces. Portal Designer позволяет визуально создавать и редактирования приложения порталов, а также визуально редактировать их темы и внешний вид.

Автоматизация сборки приложений. При работе с инструментами предыдущих поколений разработчикам приходилось вручную компоновать архивы Java (JAR) и WAR в файл EAR для развертывания. Application Developer автоматизирует сборку EAR и генерирует дескрипторы развертывания для новых приложений, экономя ваше время.

Повышение качества кода. Чем раньше замечены ошибки, тем проще их исправить, поэтому в состав Application Developer входят инструменты, которые помогают разработчикам повысить качество кода. Во-первых, функция автоматизированного анализа кода применяет более 200 стандартов и практических рекомендаций кодирования J2EE к коду приложения, пометая нарушения и подсказывая варианты устранения ошибок – включая вносимые одним щелчком быстрые исправления. В числе имеющихся категорий правил – практические рекомендации программирования на J2SE/J2EE, глобализации и доступности; интерактивный редактор правил позволяет определять новые правила.

Во-вторых, тестирование компонентов Java и web-служб автоматизирует создание тестовых заглушек, наборов и входных данных на основе анализа кода или файла WSDL. При создании и выполнении тестов используются приоритеты, основанные на различных указанных метриках; динамически генерируемые входные и выходные данные можно преобразовать в гибком редакторе данных, который поддерживает создание сложных объектов и вставки пользовательского кода. Поддерживается как локальное, так и удаленное тестирование.

Ускорение отладки. Для ускорения процесса отладки в Application Developer имеется множество инструментов анализа времени выполнения для приложений J2EE, выполняемых как на локальной, так и на удаленной машине. Визуализированный поток выполнения отображается на интерактивной блочной диаграмме на базе UML, которая коррелирует с информацией о потоках, с тем чтобы обнаруживать и диагностировать проблемные состояния. Графики вызовов для графического анализа производительности помогают выявлять узкие места на уровне методов, а отчеты с цветовым кодированием выявляют и отображают невыполненные фрагменты кода на уровне строк. Мощная утилита обнаружения утечки памяти использует технологию, разработанную IBM Research, чтобы сузить множество из десятков тысяч кандидатов на утечку до нескольких подозрительных участков. Кроме того, в выполняемый код можно вставлять датчики для пользовательского анализа, которые позволяют выполнять определенный пользователем код при выполнении операций entry, exit и try/catch над указанными классами и методами.

Оптимизация тестирования модулей и систем. Интегрированные среды тестирования модулей поддерживают тестирование и отладку локального и серверного кода на IBM WebSphere Application Server, WebSphere Application Server Express, WebSphere Portal и Apache Tomcat. Вы можете создавать и настраивать экземпляры серверов с помощью мастеров, выполнять пошаговую трассировку приложений, задавать точки останова и даже модифицировать код во время отладки, не перезапуская сервер тестирования модуля. Можно создавать, тестировать и развертывать приложения J2EE на BEA WebLogic Server, версии 6.1, 7.0 и 8.1, с помощью IBM Rational Deployment Toolkit for WebLogic Server.

Повышение производительности труда группы благодаря эффективному управлению изменениями.

Application Developer тесно интегрирован с решениями IBM Rational ClearCase и IBM Rational ClearQuest для управления конфигурацией. Такая глубокая интеграция позволяет выполнять множество операций по управлению изменениями прямо в Application Developer, а также дает интегрированное представление проектов, которое помогает повысить как качество сотрудничества, так и производительность труда группы. Application Developer включает в себя IBM Rational ClearCase LT, который обеспечивает надежный контроль версий на уровне записей и содержит готовые средства поддержки групповой работы. Также продукт поддерживает систему Concurrent Versions System и системы управления конфигурацией от независимых поставщиков через специализированные модули расширения (plug-in).

Визуализация и графическое редактирование кода. Редактор UML (Unified Modeling Language) Visual Editor позволяет графически визуализировать и редактировать код и объекты данных J2EE с помощью стандартной нотации UML, благодаря чему разработчики, даже малознакомые с UML, могут лучше понимать сложный код и управлять им. UML Visual Editor также содержит мощные графические редакторы, с помощью которых вы можете добавлять диаграммы классов UML непосредственно в проект Java или EJB, который автоматически создаст соответствующий код. Затем этот код можно модифицировать – как напрямую, так и с помощью диаграмм классов UML. А поскольку визуализация выполняется динамически, она всегда синхронизирована с нижележащим кодом, и этот код можно переделывать, перемещая объекты в диаграмме класса.

Для создания нового кода Application Developer позволяет добавлять диаграммы классов UML непосредственно в проект Java или EJB, автоматически генерировать соответствующий код и редактировать его либо непосредственно в диаграммах классов UML, либо в проекте Java или EJB. С помощью редактора диаграмм классов можно визуализировать объекты данных, создавать новые «зерна», создавать и модифицировать связи между «зернами» или исследовать имеющиеся связи между «зернами», классами и интерфейсами. Диаграммы помогают вам обнаружить и выделить связи между фрагментами кода и объектами данных, не столь очевидные в других редакторах, что поможет вам в беседах с архитекторами ПО и другими участниками команды разработчиков.

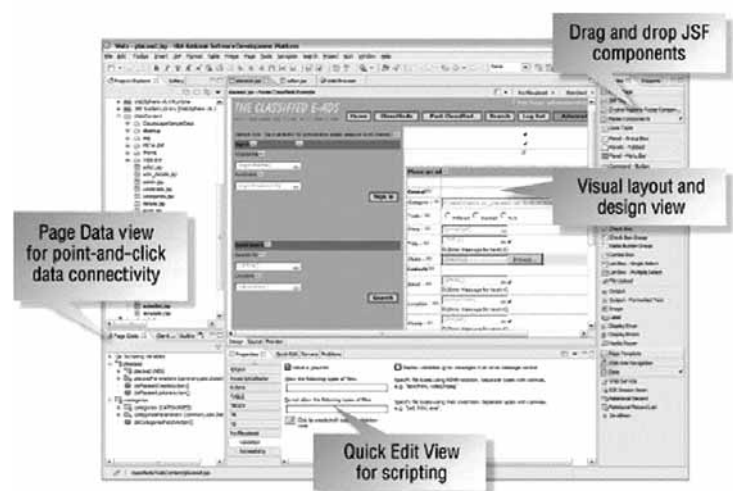
Возможности Rational Application Developer построены на платформе Eclipse с открытым исходным кодом, благодаря чему вы можете адаптировать и расширять свою среду разработки приложений с помощью модулей расширения Eclipse (plug-in), созданными IBM, бизнес-партнерами IBM и сообществом Eclipse.

А при использовании Rational Application Developer в сочетании с платформой IBM Software Development Platform вы можете получить доступ к широкому спектру требований и функций управления изменениями непосредственно из Rational Application Developer и оценить высокий уровень интеграции на протяжении всего цикла разработки.

IBM Rational Web Developer

IBM Rational Web Developer – это инструмент для быстрой разработки, тестирования и развертывания приложений Web, web-служб и приложений Java в среде IDE, простой в обучении и работе. IBM Rational Web Developer полностью соответствует потребностям разработчиков, которым необходимы инструменты, максимально облегчающие работу со сложными задачами и интуитивно понятные в работе.

Web Developer представляет собой интегрированную среду разработки, в которой можно быстро и легко разрабатывать web-приложения, web-службы и приложения Java. Благодаря широкому спектру простых в изучении и работе визуальных инструментов для разработки, тестирования и развертывания Web Developer является превосходным выбором для разработчиков ПО, не имеющих большого опыта работы с Java или не нуждающихся в создании собственных компонентов EJB.



Web Developer основан на платформе Eclipse с открытым исходным кодом, благодаря чему вы можете адаптировать и расширять свою среду разработки приложений в соответствии с вашими конкретными потребностями. При использовании продукта в сочетании с платформой IBM Software Development Platform вы можете получить доступ к широкому спектру требований и функций управления изменениями непосредственно из Web Developer.

IBM Rational Web Developer for WebSphere Software оптимизирован для программного обеспечения WebSphere и содержит возможности разработки для других технологических платформ.

Основные преимущества IBM Rational Web Developer

Особенность	Функция	Преимущества для практика	Преимущества для менеджера проекта	Преимущества для бизнеса
Открытая, расширяемая и настраиваемая платформа разработки	Платформа на основе Eclipse может расширяться с помощью plug-ins и настраиваться	Адаптируемые к новым/растущим требованиям инструменты повышают продуктивность	Гибкость, сокращение затрат, выше продуктивность	Гибкость, сокращение затрат, выше продуктивность
WebSphere Test Environment	Встроенный test server для мгновенного тестирования	Не нужно устанавливать отдельный сервер	Выше продуктивность и качество	Сокращение затрат, повышение эффективности
Поддержка RAD для Web-приложений	Графическое конструирование страниц HTML/JSP, создание drag/drop компонентов JSF/интеракций с данными (SDO), утилиты и мастера Struts Построение бизнес-логики на EGL, упрощенный процедурный язык 4GL для программистов, не пользующихся Java	Выше продуктивность и качество. Быстрее можно приступить к созданию web-приложений	Выше продуктивность и качество. Можно использовать традиционные методы и легче подбирать команду для проекта	Сокращение затрат, повышение эффективности Быстрее выполняются проекты
Поддержка RAD для Web-сайтов	Web-дизайнеры в графическом виде проектируют сайты, шаблоны страниц и style sheets для поддержания единого оформления	Выше продуктивность и качество, упрощается поддержание приложений.	Улучшается продуктивность, разработчика.	Сокращение затрат, повышение эффективности
Поддержка Web-сервисов	Мастера/утилиты для легкой подготовки/использования/тестирования Web-сервисов	Разработчик абстрагируется от более низкого уровня.	Меньше затраты на обучение и выше продуктивность	Продуктивность повышается за счет повторного использования, эффективности бизнеса.

Упрощение web-разработки и управления web-сайтами. Мастера и инструменты быстрой разработки приложений (RAD) объединяют практически все аспекты web-разработки (HTML, JSP, JSF и сервлеты) в единый интерфейс и позволяют разработчикам с различным уровнем подготовки и даже тем, кто не знаком с Java, быстро строить многофункциональные приложения для работы с данными.

- Визуальное проектирование и конструирование web-приложений с помощью Model-View-Controller и схем Struts 1.1 и/или JavaServer Faces
- Быстрая разработка многофункциональных пользовательских web-интерфейсов и web-форм с помощью многократно используемых, перетаскиваемых компонентов JavaServer Faces, которые генерируют код обработки событий, проверки пользовательского ввода и привязки данных для web-приложений
- Возможность подключения web-приложений к реляционным базам данных, компонентам EJB и web-службам с помощью простых инструментов, поддерживающих Service Data Objects, новый промышленный стандарт для доступа к гетерогенным данным
- Создание интерактивных web-интерфейсов пользователя с производительностью и характеристиками техобслуживания на уровне тонких клиентов с помощью клиентских компонентов Faces, расширяющих спецификацию JSF
- Простая разработка многофункциональных web-страниц в режиме визуального редактирования или редактирования исходного кода с помощью богатого возможностями редактора HTML и JSP
- Возможность быстро импортировать web-сайт целиком, изменять стили и применять их в глобальном масштабе, визуально добавлять и удалять страницы из древовидного структурного представления с помощью Web Site Designer
- Простое создание и анимация оригинальной графики с помощью компонентов WebArt Designer и Animated GIF Designer¹

Кодирование на 4GL и генерация кода на Java. Enterprise Generation Language (EGL) – это язык четвертого поколения, который позволяет разработчикам процедур быстро создавать и отлаживать управляемые данными web-приложения и бизнес-логику с помощью знакомых программных конструкций и без написания кода на Java. EGL содержит конструкции и функциональные возможности Informix 4GL, благодаря чему Web Developer является идеальной средой разработки для Informix. EGL тесно интегрирован с JavaServer Faces, что дает возможность создавать высокоинтерактивные, управляемые данными web-приложения с помощью компонентов JSF.

Интеграция приложений с помощью web-служб и архитектур, ориентированных на службы. Web Developer содержит инструменты, необходимые для обнаружения, создания, сборки, тестирования, развертывания и публикации web-служб, совместимых с WS-I. Вы можете создавать новые web-службы с нуля или добавлять web-службы в существующие приложения, чтобы упростить интеграцию и повторное использование.

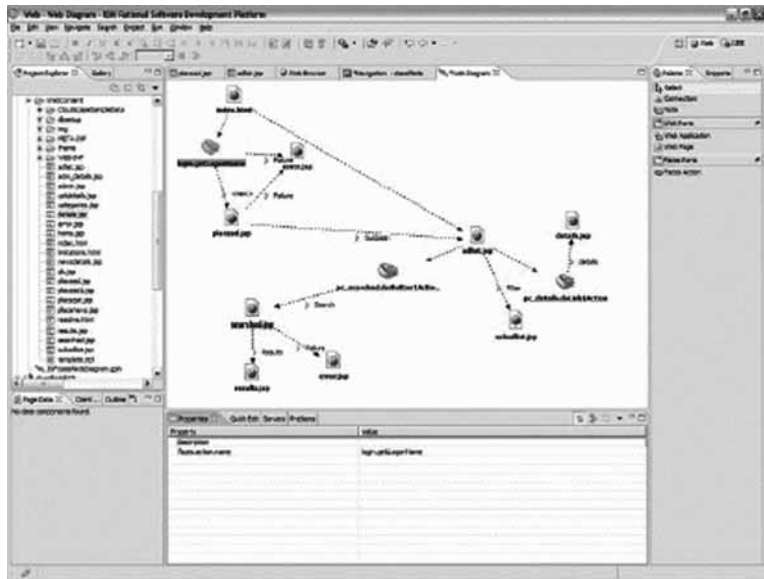
- Автоматическая генерация файлов WSDL и WSIL, необходимых для описания web-служб и подготовки их к инспекции на основе имеющихся материалов, либо разработка файла WSDL и генерация кода Java
- Простота создания, проверки и обнаружения совместимых web-служб, соответствующих WS-I, с помощью редактора WSDL и браузера web-служб
- Кодирование входных и выходных сообщений с помощью SOAP
- Создание Java-прокси для взаимодействия с клиентскими приложениями
- Генерация тестовых клиентов web-служб без кодирования
- Занесение в реестр UDDI версии 2

Упрощение разработки XML. Полный набор функций XML в составе Web Developer позволяет разработчикам создавать, редактировать и преобразовывать документы XML.

- Визуальные инструменты для создания и просмотра файлов XML и для автоматической проверки документов на соответствие определениям типа документа (DTD)
- Визуальные инструменты для преобразования элементов XML в таблицы реляционной базы данных
- Тесная интеграция с IBM DB2 для создания документов XML из существующих данных DB2 или разбора документов XML в данные DB2
- Инструменты для экономии времени при сборке и отладке сценариев преобразования Extensible Stylesheet Language (XSL) и для преобразования документов XML в другие форматы разметки, в том числе HTML

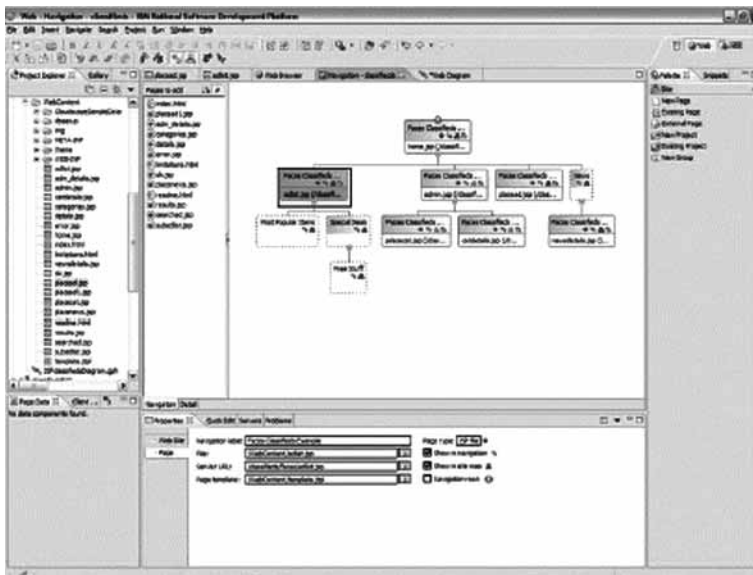
Создание приложений на основе баз данных. Web Developer содержит интегрированные инструменты для создания приложений баз данных из запросов и элементов Beans («зерен»). Мастер соединений с базой данных упрощает создание соединения Java Database Connectivity (JDBC) с СУБД, такими как IBM DB2 Universal Database, IBM Informix, Oracle или Microsoft SQL Server.

Мастер запросов SQL и построитель запросов SQL предоставляют визуальный интерфейс для создания и выполнения операторов SQL. Можно создать простой запрос с помощью мастера запросов SQL или использовать построитель запросов SQL, который поддерживает более широкий набор операторов. Встроенный мастер преобразования SQL-XML позволяет создавать документы XML и XSL, спецификации DTD, схемы XSD, файлы HTML и связанные артефакты. Поддержка SQL for Java (SQLJ) позволяет быстро создавать и отлаживать приложения, использующие хранимые процедуры SQLJ и DB2 SQLJ.



Разработка ссылочной модели в Web Developer

Быстрое создание приложений Java с помощью визуальных инструментов. Web Developer включает в себя большой набор инструментов для создания приложений Java и поддерживает Java Development Kit версии 1.4.2. Visual Editor for Java позволяет разрабатывать GUI для Java методом перетаскивания и вставки, с помощью компонентов Abstract Window Toolkit (AWT), SWT или Swing, а также позволяет визуальным образом привязывать данные к виджетам (widget) UI и немедленно просматривать изменения вида GUI благодаря динамическому механизму обновления для исходного кода и визуального дизайна.



Тестирование и отладка в Web Developer

Оптимизация тестирования приложений. Интегрированные среды тестирования модулей и визуальный отладчик значительно повышают производительность труда разработчиков и снижают общие затраты на разработку, так как можно реализовать полный тестовый сервер на локальной машине. Web Developer предоставляет среду для быстрого тестирования и отладки локального и серверного кода на WebSphere Application Server, WebSphere Portal и Apache Tomcat.

- Встроенные мастера помогают создавать и настраивать экземпляры серверов; кроме того, Web Developer может создавать их автоматически.
- Возможно выполнять пошаговую трассировку приложений, задавать точки останова и даже модифицировать код на лету и возобновлять отладку, не перезапуская приложение.
- Тестирование JavaScript и Active Script на клиентской стороне с помощью Active Script Debugger
- Можно создавать, тестировать и развертывать приложения на BEA WebLogic Server, версии 6.1, 7.0 и 8.1, с помощью IBM Rational Deployment Toolkit for WebLogic Server.

Повышение производительности разработки благодаря эффективному управлению изменениями. Для быстрого и экономичного создания конкурентоспособных web-приложений необходимы скоординированные усилия всей группы разработчиков, включая дизайнеров графики, программистов и бизнес-аналитиков. Web Developer тесно интегрирован с решениями IBM Rational ClearCase и IBM Rational ClearQuest для управления конфигурацией. Такая глубокая интеграция позволяет выполнять множество операций по управлению изменениями прямо в Web Developer, а также дает интегрированное представление проектов, которое помогает повысить как качество сотрудничества, так и производительность труда группы. Web Developer также поддерживает систему Concurrent Versions System и системы управления конфигурацией от независимых поставщиков через специализированные модули расширения (plug-in).

IBM Rational Web Developer входит в состав пакета IBM Rational Software Architect и существует как отдельный продукт.

IBM Rational XDE Developer

Продукты Rational XDE Developer и Rational XDE DeveloperPlus являются дальнейшим развитием возможностей, заложенных в Rational XDE Modeler. Эти инструменты позволяют выполнять разработку информационных систем с использованием Round-Trip подхода, предполагающего автоматизированную генерацию каркаса кода на базе визуальной модели и обратное проектирование – построение модели на основе существующего кода.

Rational XDE DeveloperPlus дополнительно включает продукты Rational Purify, Quantify и PureCoverage. Также в данный инструмент входит Rational Visual Trace, который позволяет создать диаграмму последовательности на основе выполнения приложения. Этот продукт позволяет отображать созданные объекты приложения и последовательность вызовов их методов.

Средства Rational XDE Developer и Rational XDE DeveloperPlus поставляются для конкретных платформ разработки приложений. Таким образом, существуют Rational XDE Developer для платформы Java (Rational XDE DeveloperPlus for Java) и Rational XDE Developer для платформы .NET (Rational XDE DeveloperPlus for .NET).

Тестирование

Тестирование программного обеспечения занимает от 30 до 50 процентов от всей стоимости разработки. Однако многие полагают, что приложение не может быть хорошо протестировано до момента внедрения. Это заблуждение основано на двух фактах. Во-первых, тестирование ПО является чрезвычайно сложным процессом. Выполнение любой программы может иметь неисчислимое количество различных путей. Во-вторых, тестирование часто проводится без четкой методологии и без требуемой автоматизации с помощью соответствующих инструментальных средств. Сложность создаваемого ПО делает невозможным проведение 100-% тестирования, но хорошо продуманная методология и использование современных инструментальных средств, могут значительно улучшить производительность и эффективность тестирования ПО.

Основные цели тестирования:

- проверить взаимодействие между объектами;
- проверить корректную интеграцию всех модулей системы;
- проверить, что все требования были корректно реализованы;
- идентифицировать дефекты и убедиться, что они максимально выявлены еще до развертывания системы.

Хорошо выполненные тесты, запуск которых осуществляется еще на ранней стадии жизненного цикла, могут значительно снизить стоимость завершения проекта и поддержки ПО. Это может также значительно снизить риски или штрафы, связанные с поставкой ПО плохого качества, исключить низкую производительность работы приложений конечных пользователей, неудобство ввода данных, наличие вычислительных ошибок и ошибочное функциональное поведение системы. Для систем "с особыми требованиями к безопасности", когда отказ в работе может причинить вред людям или принести значительные убытки компании - таких, как системы управления воздушными полетами, управления ракетами или медицинскими поставками, финансовые приложения - высокие требования к качеству ПО являются необходимыми для успеха разрабатываемой системы. Для обычной информационной административной системы такие требования не являются настолько критичными, но эффект от всего лишь одного дефекта может быть, тем не менее, достаточно дорогостоящим.

В соответствии с RUP принята следующая базовая классификация видов тестирования:

1. Функциональное тестирование

- непосредственно функциональное тестирование (Function testing);
- тестирование целостности данных (Data integrity testing);
- тестирование на разных платформах (Configuration testing);
- тестирование отказоустойчивости (Failover & recovery testing);
- тестирование доступа (Security testing);
- инсталляционное тестирование (Installation testing);
- тестирование пользовательского интерфейса (User interface testing).

2. Нагрузочное тестирование

- непосредственно нагрузочное тестирование (Load testing);
- профилирование производительности (Performance profiling);
- тестирование цикла работы (Business cycle testing);
- тестирование при большой пользовательской нагрузке (Stress testing);
- тестирование на больших объемах данных (Volume testing).

Для автоматизации процессов тестирования IBM Rational предлагает следующие инструментальные средства.

Средства тестирования в линейке продуктов Atlantic, входящие в пакет IBM Rational Professional Bundle:

- IBM Rational Performance Tester - средство нагрузочного тестирования
- IBM Rational Functional Tester - средство функционального тестирования, включает также
- IBM Rational Manual Tester - средство для организации и проведения ручного тестирования

Эти средства предназначены для функционального и нагрузочного тестирования программного обеспечения, создаваемого на платформе J2EE. Они интегрированы с инструментом IBM Rational Test Manager, входящего в пакет IBM Rational Team Unified Platform и служащего средством планирования и мониторинга процесса тестирования как для инструментов тестирования новой линейки продуктов Atlantic, так и многоплатформенных средств тестирования линейки продуктов IBM Rational Suite.

Средства тестирования, в составе IBM Rational Suite:

- IBM Rational Robot – средство разработки, записи и выполнения скриптов автоматизированного функционального и регрессионного тестирования приложений, предоставляющее полную поддержку тестирования всех средств управления Visual Studio.NET.
- IBM Rational XDE Tester - расширенные средства автоматизированного функционального и регрессионного тестирования Java- и Web-приложений из сред разработки Eclipse IDE, IBM WSAD и Rational XDE.
- IBM Rational Purify - средство выявления ошибок, связанных с обращением к динамической памяти (версии для Windows и UNIX)
- IBM Rational Quantify - средство выявления узких мест в коде, оказывающих влияние на производительность разрабатываемой информационной системы.
- IBM Rational PureCoverage - средство определения полноты тестирования кода.
- IBM Rational TestFactory - средство для полуавтоматического формирования набора тестовых скриптов, предназначенных для проведения функционального тестирования и обеспечивающих его полноту для конкретной информационной системы; способен выполнить анализ графического интерфейса разрабатываемой ИС и сгенерировать для нее комплексный набор тестов, позволяющий провести максимально полное функциональное тестирование.

Средства управления тестированием в составе Team Unifying Platform:

- IBM Test Manager – средство планирования и мониторинга процесса тестирования, входит в объединяющую платформу Team Unifying Platform, а также в пакет IBM Rational Suite.

Основным инструментом для планирования процесса тестирования, описания его сценариев и управления всем ходом данного процесса является IBM Rational TestManager, который входит в состав пакета Team Unifying Platform и подробно описан в соответствующем разделе. На основе требований на тестирование, которые аккумулируются с помощью IBM Rational RequisitePro, специальный механизм интеграции RequisitePro и TestManager позволяет сформировать план тестирования. Элементами плана служат сценарии, каждый из которых позволяет протестировать какой-либо аспект работы создаваемой ИС. Сформированный план тестирования дополняется тестовыми скриптами, которые формируются с помощью IBM Rational Robot и позволяют автоматизировать процесс тестирования.

При проведении распределенного тестирования, в частности, когда требуется одновременно проверить систему на разных программно-аппаратных платформах, используются специальные агенты (Test Agents). Таким образом, можно одновременно протестировать систему в операционных системах Windows, Linux, HP-UX, Solaris и AIX.

Для проведения расширенного функционального и регрессионного тестирования Java- и Web-приложений рекомендуется использовать новый продукт - IBM Rational XDE Tester. Для автоматизации процесса создания необходимых отчетов — плана тестирования, отчетов по результатам тестирования и т. п. — могут быть использованы Seagate Crystal Report и IBM Rational SoDA.

Для анализа работы систем, построенных с помощью языков C/C++, Microsoft Visual Basic, Java, C#.NET, VB.NET и Java .NET, можно использовать средства IBM Rational Purify, Quantify и PureCoverage. Основное назначение Purify при тестировании native-приложений (приложения, работа которых не управляется с помощью дополнительного окружения, например, так называемых сборщиков мусора) — выявить всевозможные утечки памяти и любые иные ошибки работы с ней. Для приложений класса managed applications (такие, как Java- и NET-приложения) Purify позволяет провести эффективное профилирование памяти с целью оптимизации ее использования. С помощью Quantify становится проще определить узкие места производительности системы и провести ее настройку вплоть до отдельных строк программного кода. PureCoverage окажется незаменимым, если требуется оценить полноту тестирования системы.

IBM Rational TestFactory - еще одно специализированное средство автоматизированного тестирования, предназначенное для анализа графического интерфейса пользователя (GUI) разрабатываемой информационной системы и генерации полного набора тестов, позволяющего провести максимально полное функциональное тестирование.

IBM Rational Performance Tester

IBM Rational Performance Tester – это инструмент для создания, выполнения и анализа тестов производительности, предназначенный для проверки надежности и масштабируемости Web-приложений до их развертывания. Этот инструмент объединяет простое в использовании средство для записи тестов с масштабируемым механизмом выполнения, поддерживающим планирование, создание отчетов в реальном времени и

Возможности IBM Rational Performance Tester

- Тестирование работы Web-приложений в многопользовательском режиме
- Визуальный редактор, поддерживающий как детализированный, так и высокоуровневый режим тестирования
- Гибкие средства моделирования и эмуляции групп пользователей
- IBM WebSphere или Microsoft Visual Studio.NET;
- Низкие требования к памяти и мощности процессора, что позволяет проводить масштабное тестирование многопользовательских приложений с использованием ограниченных аппаратных ресурсов
- Средства создания отчетов в реальном времени, позволяющие немедленно обнаруживать снижение производительности
- Автоматическая идентификация и поддержка динамических ответов сервера
- Автоматическое изменение тестовых данных
- Динамическое изменение пользовательской нагрузки в ходе тестирования.
- Сбор и визуализация ресурсных данных сервера
- Использование браузера для просмотра страниц тестирования
- Интеграция пользовательского Java-кода для гибкой настройки тестов
- Выполнение тестов под управлением операционных систем Windows и Linux.
- Полная версия IBM Rational ClearCaseLT в составе пакета

автоматическое варьирование данных. Все это позволяет гарантировать, что приложения смогут справиться с обработкой больших объемов пользовательской нагрузки.

Успешность применения Web-приложений определяется такими факторами как функциональность и производительность. «Функциональность» обозначает доступные пользователю функции, включая выполнение транзакций и доступ к информации. «Производительность» характеризует способность системы быстро и точно выполнять транзакции и предоставлять информацию несмотря на работу в многопользовательском режиме или ограниченные аппаратные ресурсы.

Отказ приложения, связанный с производительностью, можно предотвратить с помощью предварительного тестирования до этапа развертывания. Однако большинство команд сталкиваются с одной или несколькими из следующих проблем, которые могут повлечь снижение доступности, надежности и масштабируемости при развертывании приложения в реальных условиях:

- Ошибки, связанные с пониманием реального поведения пользователей
- Неполное тестирование производительности до развертывания, либо отсутствие такого этапа
- Недостаток опыта создания тестов производительности, которые точно отражают действия пользователей.

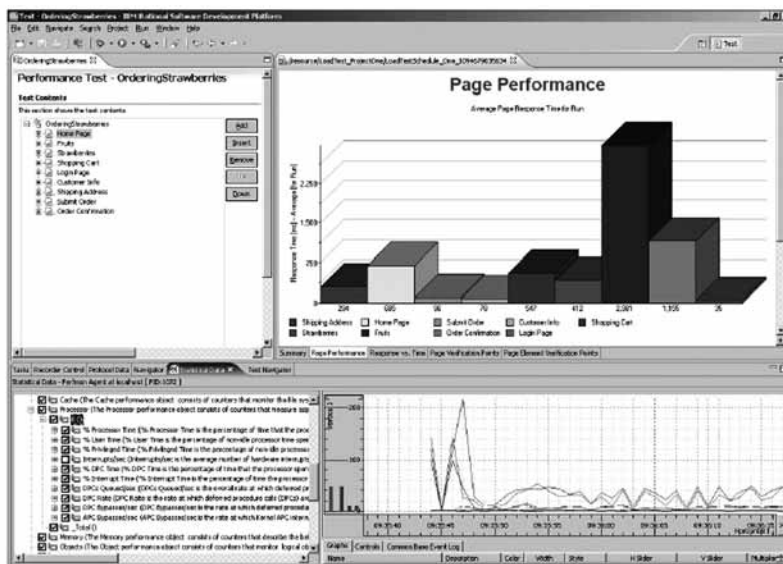
Решение IBM® Rational® Performance Tester было создано для решения этих проблем. Performance Tester уменьшает уровень сложности тестирования рабочих нагрузок и позволяют командам любой квалификации эффективно оценивать производительность приложения до его развертывания.

Простота освоения для новичков. Rational Performance Tester содержит функции, специально созданные для быстрого освоения. Для записи теста необходимо простое взаимодействие с целевым Web-приложением через браузер специалиста по тестированию – Internet Explorer, Netscape или Mozilla, – работающего под управлением операционной системы Windows или Linux. Результаты теста представляются в визуальном редакторе высокого уровня, который поддерживает любой требуемый эксперту уровень детализации. Динамические отклики сервера автоматически идентифицируются и обрабатываются, что помогает новичкам проводить тестирование на основе данных с варьированием входных данных для разных эмулируемых пользователей – не прибегая при этом к ручной модификации теста. С помощью встроенного средства для просмотра Web-страниц каждую загруженную страницу можно просматривать в оффлайн-режиме, анализируя взаимодействие пользователей и транзакции.

Тестовые скрипты можно объединять в различных комбинациях, моделируя поведение различных групп пользователей, входящих в состав предполагаемой аудитории. После этого специалисты по тестированию могут задать ряд моделируемых транзакций пользователей для выполнения эмуляции. Выполнение теста сопровождается созданием удобных для чтения отчетов, которые обновляются в реальном времени в ходе всего тестового прогона. Узкие места идентифицируются по таким показателям, как производительность при работе «по замкнутому циклу», скорость выполнения транзакций и диагностика системы.

Расширенные функции для экспертов. Помимо средств, которые помогают ускорить освоение инструмента новичками, Rational Performance Tester включает также расширенные функции для экспертов, включая членов команды, опытных специалистов по тестированию и ИТ-администраторов.

Время отклика для отдельной страницы может быть «разложено» на аналогичные показатели для отдельных элементов страницы (например, JPG, JSP, ASP), что помогает идентифицировать элементы, ответственные за медленный отклик. В дополнении к средствам автоматической генерации



IBM Rational Performance Tester

и корреляции данных, разработчики могут вставлять собственный программный код Java, который может быть исполнен в любой момент в ходе тестирования. Это позволяет реализовать усовершенствованные методы диагностики и манипулирования данными.

В ходе проведения теста, информация о системных ресурсах, например, коэффициент использования центрального процессора (ЦП) и памяти, может собираться с удаленного сервера и корректироваться с учетом времени отклика и данных о пропускной способности. Собранные данные об использовании ресурсов имеют критически важное значение для определения удаленной системы – маршрутизатора, Web-сервера, сервера приложений, сервера базы данных и т.д. – которая отвечает за обнаруженные простои, а также для идентификации компонента системы (ЦП, оперативная память, жесткий диск и т.д.), который стал причиной узкого места.

Уменьшение расходов на тестирование производительности. Rational Performance Tester характеризуется низкими требованиями к ресурсам процессора и памяти при эмулировании многопользовательского режима. В результате, высокий уровень масштабируемости достигается даже в том случае, если команда не имеет доступа к мощным компьютерам. Кроме того, тесты могут выполняться под управлением операционных систем Windows и Linux, что оптимизирует использование существующих аппаратных ресурсов.

Поддержка открытых стандартов. Rational Performance Tester создан на базе архитектурной оболочки Eclipse и ее расширения Hyades, которое предлагает расширенные функции тестирования и мониторинга. Eclipse и Hyades являются проектами на основе открытого исходного кода, которые поддерживают совместно используемые службы (на основе открытого исходного кода) для разных инструментальных сред, предназначенных для мониторинга, разработки, тестирования и развертывания приложений. Использование этой инфраструктуры приносит ряд дополнительных преимуществ – от «ухода» от патентованных форматов хранения данных до поддержки настройки для внутренних и сторонних систем. Инвестирование средств в инструменты на базе Eclipse и Hyades помогает избежать «заикливания» на одном поставщике и поощряет внедрение инноваций.

IBM Rational Performance Tester является составной частью платформы IBM Software Development Platform. Интеграция с платформой Software Development Platform обеспечивает совместный доступ пользователей к различным средствам поддержки команд разработчиков. С точки зрения тестирования производительности, это позволяет разработчикам и специалистам по тестированию создавать, совместно использовать и анализировать тесты производительности с использованием того же пользовательского интерфейса, с помощью которого они осуществляют мониторинг, строят, тестируют и развертывают свои приложения в рамках Software Development Platform с применением и других инструментов.

IBM Rational Performance Tester входит в состав пакета IBM Rational Professional Bundle, который включает все инструменты, необходимые для проектирования, создания и тестирования приложений J2EE/порталов/служб.

IBM Rational Functional Tester

IBM Rational Functional Tester представляет собой усовершенствованный инструмент для автоматического функционального и регрессионного тестирования, предназначенный для разработчиков интерфейсов GUI и специалистов по тестированию, которым необходим полный контроль за тестированием программных приложений. Это инструмент упрощает сложные задачи для новичков и, в то же время, предоставляет в распоряжение экспертов средства для работы со скриптовыми языками тестирования – Java и VB.NET, обеспечивая неограниченные возможности для адаптации.

Обычно функциональное тестирование входит в сферу ответственности отделов контроля качества, которые контролируют успешную реализацию требований с точки зрения функциональности, удобства и простоты использования и т.д. Функциональное тестирование позволяет оценить, какая доля требований к приложению уже выполнена, какой процент из них успешно реализован и насколько команда отклонилась от плановых показателей проекта.

Хотя функциональное тестирование может быть выполнено и с использованием ручного подхода, автоматизация позволяет получить значительные выгоды:

- Позволяет проводить регрессионное тестирование – по отличительным признакам различных версий – без «связывания» тестовых ресурсов

- Освобождает персонал контроля качества от выполнения базового тестирования, помогая им создавать дополнительные, более полные тесты
- Автоматизирует вспомогательные операции, такие, как подготовка стендового оборудования и конфигурирование баз данных
- Уменьшает количество ошибок, связанных с человеческим фактором, в ходе выполнения таких операций, как прогон шагов теста и запись результатов тестирования.

Решение IBM® Rational® Functional Tester позволяет автоматизировать функциональное и регрессионное тестирование. Созданный на основе технологий Java, Web и VS.NET WinForm, Functional Tester объединяет надежное средство для записи действий пользователя с многочисленными опциями настройки и интеллектуальными функциями для работы со скриптами, что позволяет создавать и выполнять тесты с необходимым уровнем гибкости, соответствующим постоянному изменению приложений.

Снижение затрат на функциональное тестирование. Преимущества автоматического тестирования быстро сводятся на нет, если команде приходится тратить много времени на ручное обновление скриптов тестирования на протяжении всего проекта. Functional Tester использует усовершенствованную технологию ScriptAssure для того, чтобы «изучить» контрольные характеристики пользовательского интерфейса, что позволяет идентифицировать те же самые средства управления в новой версии, несмотря на внесенные изменения. Эти характеристики сохраняются в объектной карте, совместный доступ к которой могут получить различные скрипты и участники проекта. Благодаря этой карте изменения, внесенные в характеристики распознавания объекта, будут отражены во всех скриптах тестирования, что существенно упрощает обслуживание. А новые, расширенные средства фильтрации и поиска по объектным картам сделали обслуживание еще более простой задачей.

В текущую версию Functional Tester включена обновленная версия функции по автоматическому соотношению данных, которая позволяет командам реализовывать скрипты для тестирования на основе данных – меняя входные данные в ходе теста – без ручного изменения программного кода. Новый редактор пулов данных использует интерфейс на основе электронных таблиц для моделирования и сохранения ценных данных тестирования, которые будут использоваться в ходе выполнения теста.

Контрольные точки в составе программного кода теста, которые используются для сравнения фактических данных с ожидаемыми, поддерживают применение обычных выражений для согласования колебаний данных, например, при сравнении фрагментов сложных последовательностей. Все это позволяет дополнительно уменьшить объем операций по обслуживанию тестов при работе с приложениями, в которые вносились изменения.

Неограниченные возможности адаптации и настройки. Модификация и совершенствование скриптов тестирования неизбежны. Но абсолютно все – от простых циклических структур до расширенного управления файлами и взаимодействия с операционными системами – может быть в полной мере реализовано с помощью инструмента для автоматического тестирования. Functional Tester поддерживает популярные языки программирования, для работы с которыми можно использовать надежные и эффективные редакторы и отладчики.

Functional Tester поддерживает создание скриптов тестирования на языке Java – с использованием редактора Eclipse Java Developer Toolkit, или VB.NET – с использованием редактора VS.NET. Любой специалист, знакомый с этими языками программирования, сможет быстро модифицировать существующий код, а также создать пользовательские библиотеки, которые могут быть вызваны другими

Основные возможности IBM Rational Functional Tester

- Эффективная поддержка тестирования приложений на базе Java®, Web и Microsoft® Visual Studio® .NET WinForm
- Возможность выбора языка – Java или Visual Basic .NET® – для адаптации скриптов тестирования
- Встроенный редактор и отладчик Java и VB.NET для расширенного тестирования.
- Технология ScriptAssure для адаптации пользовательского интерфейса
- Автоматическое тестирование на основе данных и с учетом взаимозависимости данных, позволяющее избежать ручного написания программного кода
- Многочисленные точки проверки с поддержкой согласования систематических комбинаций
- Расширенные средства для ведения карт объектов
- Поддержка выполнения тестов и их редактирования под управлением Linux
- Контроль версий
- Дополнительная поддержка для тестирования приложений на основе терминалов 3270/5250
- IBM Rational Manual Tester и IBM Rational ClearCaseLT входят в состав пакета

скриптами тестирования. Написание программного кода значительно упрощается благодаря использованию редакторов Eclipse и VS.NET, которые предоставляют в распоряжение пользователя расширенные средства редактирования, например, автоматическое завершение кода и улучшенные опции отладки. Возможности адаптации и настройки практически ничем не ограничены, а передача скриптов в рамках команды в значительной степени упрощена. Фактически, разработчики интерфейсов GUI и другие члены команды могут получить доступ к Functional Tester для создания собственных тестов из своей среды IDE (интегрированная среда разработки) на базе Eclipse или VS.NET. В качестве примеров среды IDE на базе Eclipse можно привести IBM Rational Software Architect и IBM Rational Application Developer for WebSphere Software.

Поддержка распределенных команд разработчиков. В ряде случаев, важное значение приобретает возможность сохранения нескольких версий одного тестового скрипта, что особенно актуально для поддержки работы распределенных команд. Для поддержки контроля версий команды разработчиков могут использовать продукт IBM Rational ClearCaseLT, который поставляется в одном пакете с Functional Tester, а географически распределенные команды могут расширить его до полной версии IBM Rational ClearCase.

Поддержка открытых стандартов. Средства Rational Functional Tester, предназначенные для создания скриптов на языке Java, созданы на базе архитектурной оболочки Eclipse и ее расширения Nyades, которое предлагает расширенные функции тестирования и мониторинга. Eclipse и Nyades являются проектами на основе открытого исходного кода, которые поддерживают совместно используемые службы (на основе открытого исходного кода) для разных инструментальных сред, предназначенных для мониторинга, разработки, тестирования и развертывания приложений. Использование этой инфраструктуры приносит дополнительные преимущества – от «ухода» от патентованных форматов хранения данных до поддержки настройки для внутренних и сторонних систем. Инвестирование средств в инструменты на базе Eclipse и Nyades помогает избежать «зацикливания» на одном поставщике и поощряет внедрение инноваций.

IBM Rational Functional Tester является составной частью платформы IBM Software Development Platform. Интеграция с платформой Software Development Platform обеспечивает доступ к различным средствам по управлению жизненным циклом разработки ПО, предназначенным, например, для определения требований к приложениям, и поддерживает совместную работу членов команды разработчиков. С точки зрения функционального тестирования, это позволяет разработчикам и специалистам по тестированию создавать, совместно использовать и анализировать функциональные тесты с использованием того же самого пользовательского интерфейса, с помощью которого они осуществляют мониторинг, строят, тестируют и развертывают свои приложения в рамках платформы Software Development Platform (с применением и других инструментов).

IBM Rational Functional Tester входит в состав пакета IBM Rational Professional Bundle, который включает все инструменты, необходимые предприятию для проектирования, создания и тестирования приложений J2EE/порталов/служб. Для всего пакета предлагается единый контракт на обслуживание всего входящего в него ПО.

IBM Rational Manual Tester

IBM Rational Manual Tester – это инструмент для ручного создания и выполнения тестов, который дает возможность повторно использовать шаги теста и тем самым сократить трудоемкость работы тестировщиков и бизнес-аналитиков при внесении изменений в программное обеспечение. Этот продукт упорядочивает процесс ручного тестирования, повышая эффективность и скорость работы по оценке качества приложения.

Лишь немногие организации полностью перешли на автоматизированные тесты. Поэтому ручное тестирование остается важной частью процесса тестирования – опыт показывает, что более 75% всех работ по тестированию выполняется вручную. Популярность ручного тестирования объясняется следующими причинами:

- возможен прием на работу в качестве тестировщиков экспертов по соответствующему направлению, не имеющих опыта тестирования;
- недостаточный опыт работы с автоматизацией тестирования;
- недостаток инструментов автоматизации тестирования;
- плохая проработка или полное отсутствие в тестируемых приложениях интерфейсов, которые поддерживали бы автоматизацию



Работа с Manual Tester

IBM Rational Manual Tester предназначен для команд, в которых хотя бы часть тестирования выполняется вручную. Он позволяет преодолеть привязанный к конкретике характер ручных методов проверки, с тем чтобы гарантировать оптимальное тестирование и обнаружение дефектов в сфере деятельности, которая иначе была бы связана с высокими затратами и давала малую отдачу.

Ручное тестирование – сложнее, чем вы думаете. Ручное тестирование на первый взгляд выглядит обманчиво простым, но при неграмотной реализации превращается в настоящее испытание. Проектные команды быстро осознают всю сложность успешного управления несколькими электронными таблицами и другими средствами для хранения тестов и их результатов. Скоро становится ясно, какой объем повторной работы сопряжен с отсутствием возможности повторного использования и модульности среди множества задействованных тестовых сценариев. И все это время они борются с человеческими ошибками и непоследовательностью исполнителей. Эти недостатки зачастую сводят на нет сам смысл ручного тестирования – оценку качества приложения и обнаружение его дефектов.

Rational Manual Tester помогает группам преодолеть эти сложности на пути к успеху проекта, предоставляя средства для повышения производительности ручного тестирования. Результатом становится то, что большая часть времени и усилий направляется на выполнение сложных, информативных ручных тестов, составление лучшего представления о качестве приложения и возможность проверить больше возможностей приложения до его выпуска.

Расширенные возможности создания и поддержки тестов. Rational Manual Tester способствует использованию ссылочного контента – возможности определить те или иные шаги теста как многократно используемые компоненты, которые можно применять в нескольких тестах. Преимущество здесь не только в возможности повторного использования, которая в любом случае экономит время и деньги. Преимущество еще и в том, что оптимизируется контроль качества и единообразия тестовых сценариев. При внесении изменений в ссылочный контент они автоматически применяются ко всем тестовым сценариям, использующим этот контент, что гарантирует точное применение изменений во всей группе тестирования.

Редактор тестов поддерживает настраиваемые шрифты, создание скриншотов и вложение файлов, что позволяет тестерам четко и ясно вести обмен информацией по поводу шагов тестов. Manual Tester позволяет значительно снизить высокий уровень неопределенности, который часто является бичом других методов определения и описания шагов теста.

Автоматизация – это простота. Rational Manual Tester снижает вероятность человеческой ошибки в областях, наиболее чреватых такими ошибками – вводе и верификации данных. Во-первых, Manual Tester хранит данные в специальном поле тестового сценария, что позволяет тестировщику вставлять в него данные, не набирая их напрямую с клавиатуры. Во-вторых, данные, скопированные из пользовательского интерфейса приложения, автоматически сравниваются с ожидаемыми значениями, что автоматизирует сравнение и сохранение результатов в консолидированном тестовом отчете.

В ручные тесты также можно добавлять пользовательские точки принятия решений. Для записи успеха или неудачи конкретных событий используются точки верификации, которые позволяют запросить у тестировщика ввод дополнительных данных о выполняемой задаче. Вся эта информация содержится в отчетах, которые могут экспортироваться в документы формата CSV для хранения и анализа с помощью любого инструмента, включая популярные электронные таблицы.

Применение для различных целей. Rational Manual Tester позволяет настраивать тесты так, чтобы они отражали принятые в команде тестировщиков метрики процесса и значения. Можно определять пользовательские поля и сопоставлять их отдельным шагам тестов, чтобы охватить больший спектр тестирования, а также хранить дополнительную информацию, собранную в ходе выполнения тестов. Тестировщики также могут прилагать файлы к любым шагам в ходе выполнения теста – простой способ приложить скриншоты или другую нетекстовую информацию, которая может оказаться полезной при анализе результатов тестирования. И папку Reuse, в которой хранятся многократно используемые блоки шагов тестов, и собственные хранилища данных можно поместить в сетевую папку общего доступа или подчинить механизму контроля изменений. Manual Tester может использоваться на любых PC под управлением Windows®, поддерживая работу распределенных групп и мобильных сотрудников. Кроме того, этот программный продукт хранит тесты и результаты тестов в централизованном расположении, независимо от географического распределения и состава группы, что гарантирует простой доступ к тестовым ресурсам и составление отчетов по ним.

На базе открытых стандартов. Rational Manual Tester основан на базе архитектурной схемы Eclipse и ее расширения Nyades для тестирования, трассировки и мониторинга. Eclipse и Nyades – это два проекта с открытым исходным кодом, которые предоставляют совместно используемые службы с открытым кодом в инструментальных средах, предназначенных для разработки, тестирования, развертывания и мониторинга приложений. Преимущества этой инфраструктуры очень многочисленны и разнообразны – от устранения нестандартных форматов хранилищ данных до поддержки как внутренней, так и независимой доработки и расширения возможностей приложений. Капиталовложения в инструменты на базе Eclipse и Nyades позволяют избежать жесткой привязки к поставщику и способствуют развитию новаторства в будущем.

IBM Rational Manual Tester является неотъемлемым компонентом платформы IBM Software Development Platform.

IBM Rational Manual Tester входит в состав пакета IBM Rational Professional Bundle, который состоит из всех настольных инструментов, необходимых предприятию для разработки, конструирования и тестирования приложений, ориентированных на J2EE/порталы/службы, в едином наборе и по единому контракту на техобслуживание

IBM Rational XDE Tester

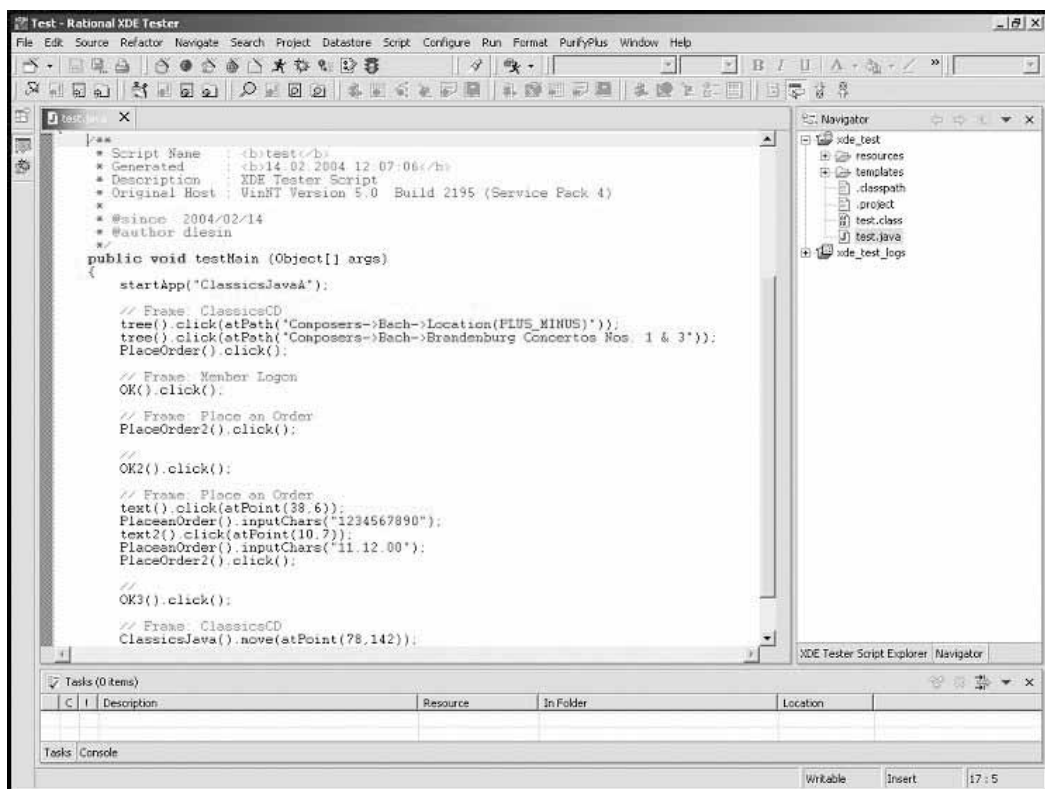
IBM Rational XDE Tester является эффективным инструментом функционального тестирования для приложений на Java и для Web-приложений и предоставляет следующие возможности:

- Java в качестве языка тестовых сценариев, тестирование приложений на Java (J2EE, J2SE, SWT, средства управления AWT/JFC);
- технологию ScriptAssure, которая практически полностью устраняет трудности с сопровождением сценариев, наряду с предоставлением бесшовной проверки достоверности динамических данных;
- мощную среду тестирования в IDE-оболочке Eclipse

IBM Rational XDE Tester интегрируется с инструментами IBM Rational, предназначенными для управления тестированием, управления требованиями, обнаружения дефектов и управления версиями сценариев тестирования. Эти инструменты и возможности интеграции упорядочивают процесс тестирования, предоставляя удобные возможности взаимодействия между бизнес-аналитиками, тестировщиками и разработчиками.

IBM Rational XDE Tester обладает четко выверенной комбинацией возможностей, предназначенной для экономии времени тестировщиков Java- и Web-приложений, вместе с тем позволяя проводить более тщательное тестирование приложений.

IBM Rational XDE Tester предоставляет в ваше распоряжение всю мощь Java для создания тестовых сценариев. Язык сценариев Java облегчает тестирование заказных пользовательских интерфейсов и средств управления Java.



Работа в Rational XDE Tester

Вследствие популярности Java становятся легко доступными образцы кода, эффективные приемы и техники. Кроме того, использование общего языка может привести к росту взаимопонимания между тестировщиками и разработчиками, повысив в итоге эффективность реализации всего проекта.

Технология ScriptAssure имеет набор уникальных возможностей, позволяющих:

- устанавливать чувствительность тестовых сценариев в соответствии с изменениями пользовательского интерфейса, при этом устраняя необходимость обновления тестовых сценариев при создании новых версий приложения;
- запускать тестирование функциональности пользовательского интерфейса, прежде чем интерфейс будет окончательно готов;
- проводить проверку достоверности информационного содержимого динамических приложений путем анализа диапазона допустимых откликов.

Технология ScriptAssure освобождает тестировщиков от рутинной работы, позволяя уделить больше времени самому тестированию.

IBM Rational XDE Tester использует мощные возможности Eclipse и IBM WebSphere Studio в качестве среды тестовой разработки. Это предоставляет тестировщикам следующие возможности:

- профессиональную открытую среду разработки;
- современный редактор с автоматическим дополнением кода, предназначенный для помощи в создании и изменении сценариев тестирования;
- полнофункциональный отладчик для простой идентификации и разрешения проблем со сценариями тестирования;
- расширенные возможности прямого обмена информацией с разработчиками.

При функциональном и регрессионном тестировании "толстых клиентов" Java- и Web-приложений выбор IBM Rational XDE Tester наиболее предпочтителен.

IBM Rational XDE Tester интегрирован с IBM Rational Team Unifying Platform — мощным набором средств управления тестированием, обнаружения дефектов, управления версиями сценариев тестирования и управления требованиями. Инструментальные средства, входящие в состав этой платформы, ускоряют разработку приложений, значительно облегчая координацию и коммуникацию внутри коллектива разработчиков.

С помощью инструментов, имеющихся в IBM Rational Team Unifying Platform, весь коллектив разработчиков будет обладать четкой информацией о результатах тестирования. В частности, разработчики смогут легко получить доступ к информации об обнаруженных тестировщиками дефектах, что поможет легко воспроизвести найденные ошибки. Кроме того, управленческому звену будет предоставлена информация, необходимая для оценки степени рисков по каждому этапу проекта.

IBM Rational Team Unifying Platform освобождает тестировщиков от обязанности вручную отслеживать изменения в требованиях, что достигается путем автоматической пометки тестовых сценариев, ссылающихся на измененные требования.

Использование IBM Rational XDE Tester вместе с IBM Rational Team Unifying Platform упорядочивает и рационализирует весь процесс тестирования.

IBM Rational XDE Tester поддерживается со стороны всемирной сервисной организации, объединяющей онлайн-ресурсы с персонализированным обучением, консультированием и технической поддержкой. IBM Rational Developer Network предоставляет доступ к статьям, технической документации, обучающим курсам, а также к онлайн-сообществу профессионалов-разработчиков, использующих наилучшие практики и инструменты IBM Rational.

Опытные консультанты, различные обучающие ресурсы и надежная техническая поддержка расширят возможности вашего коллектива разработчиков. Rational Services ускоряют достижение успеха, являясь основой для непрерывного совершенствования процессов разработки ПО.

IBM Rational Robot

IBM Rational Robot — средство, которое позволяет создавать, изменять и выполнять автоматизированные тесты для интернет-приложений, ERP- и клиент-серверных приложений. Rational Robot обеспечивает объектную поддержку для тестирования приложений, созданных с использованием технологий, спецификаций и инструментария Java, HTML, DHTML, Visual Basic, PowerBuilder, Oracle Developer/2000, Microsoft Visual Studio и т. д. Rational Robot является решением для автоматизированного тестирования, которое позволяет, единожды написав тест, вторично использовать его на любых платформах.

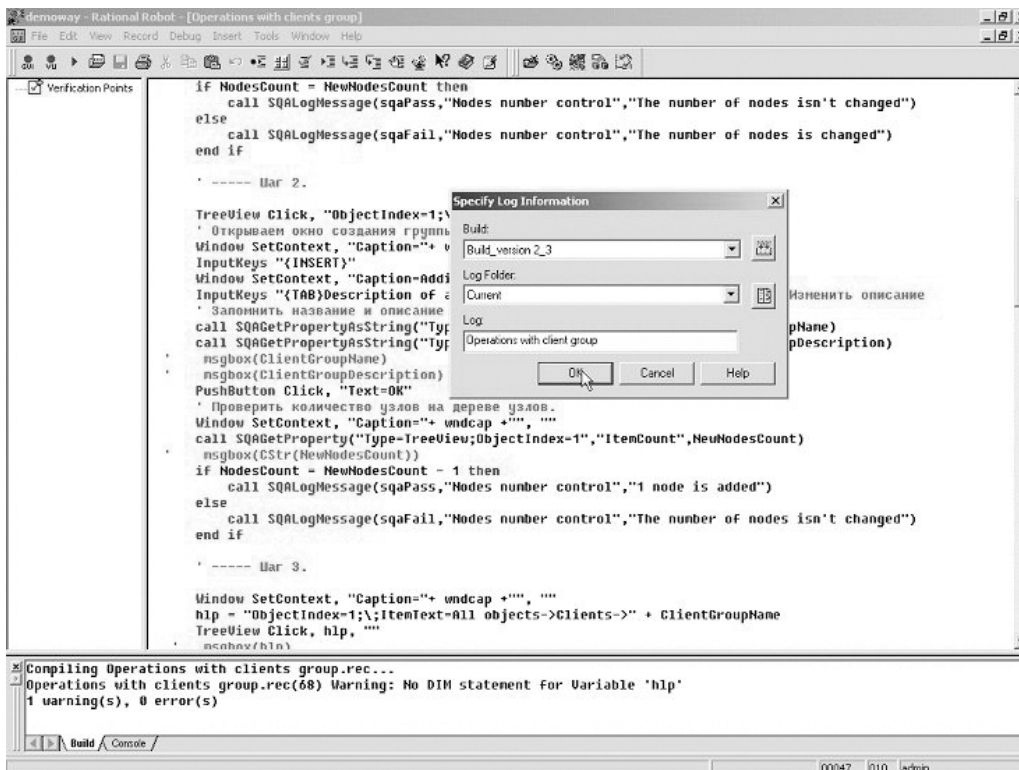
Технология Object Testing, используемая в Rational Robot, позволяет всесторонне тестировать приложения, что особенно важно в современных средах разработки, где производительность разработки ПО с использованием встроенных компонентов резко возросла. Эти различные библиотеки классов Java, средств управления ActiveX Control, OLE Control (OCX), Visual Basic Control (VBX), объекты

Visual Basic, Win32 Control и т. д. Тестирования всего лишь графических интерфейсов этих объектов недостаточно. Также важно протестировать невидимые свойства, например вложенные SQL-запросы и свойства, контролирующее их поведение.

Rational Robot генерирует сценарии функциональных тестов на языке SQABasic, который синтаксически подобен обычному Visual Basic. С помощью SQABasic можно просматривать и редактировать сценарии тестов прямо во время записи. Rational Robot включает в себя встроенный редактор и отладчик с режимом анимационного воспроизведения и онлайн-проверкой синтаксиса скрипта. В любое время можно дополнить сценарии тестов какими-либо процедурами и логическими условиями. При этом доступен вызов любой функции из DLL или из API-интерфейса Windows.

Для всеохватывающего тестирования Web-, ERP- и клиент-серверных приложений необходимо протестировать все компоненты приложения при различных условиях. Rational Robot предоставляет тестовые сценарии для меню, списков, буквенно-числовых символов и многих других объектов. Но можно и самостоятельно определить тестовые сценарии, вызывающие внешние DLL или файлы исполнения. Rational Robot также предоставляет специализированные тестовые сценарии для специфических объектов, например Java Control, ActiveX Control, OCX, VBX, PowerBuilder DataWindow, специальных объектов Oracle Form, объектов Visual Basic и т. д. Можно даже "научить" Rational Robot понимать неизвестные объекты и настраивать его таким образом, что он будет уверенно распознавать их в процессе тестирования.

Rational Robot позволяет определять многочисленные таймеры для дополнительной оценки производительности приложения.



Работа в Rational Robot

Rational Robot автоматически вносит информацию с результатами всех тестов в масштабируемый встроенный репозиторий Rational Repository. Инструмент Test Log Viewer служит для облегчения визуального анализа результатов и отображает их результаты тестирования различными цветами. Простой выбор данных позволит перейти непосредственно на соответствующую исполняемую строку в тестовом скрипте, ускоряя при этом анализ результатов.

Rational Robot обеспечивает возможность тестирования приложений, созданных в различных интегрированных средах разработки:

- HTML и DHTML;
- Java-среды разработки (Sun JDK, Symantec Visual Cafe, Microsoft J++);
- Oracle Developer/2000;
- SAP R/3;
- среды разработки VisualBasic-приложений;
 - PeopleTools;
 - PowerBuilder;
 - Microsoft Visual Studio .NET.

IBM Rational Purify

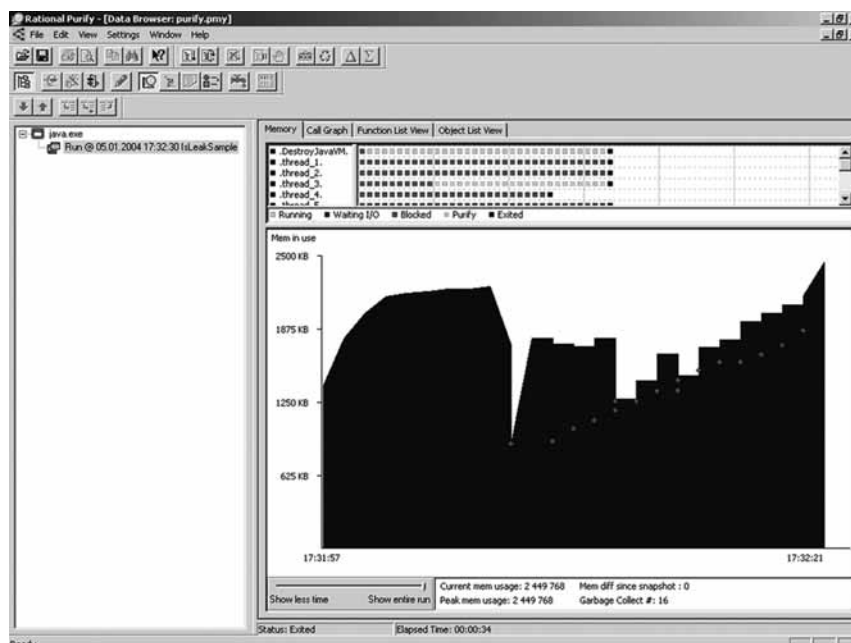
Rational Purify — средство автоматического выявления ошибок, связанных с памятью. Rational Purify помогает командам разработчиков выявлять источники ошибок с самого начала процесса разработки и в конечном счете создавать качественный продукт.

Rational Purify предлагает быстрый и эффективный способ отладки приложений в реальном времени. С использованием данного средства можно основное внимание сосредоточить на непосредственном написании кода и значительно сократить время на его отладку. При использовании Purify с начальной стадии проекта и регулярно в течение всего процесса разработки заметно повышается гарантия необходимого уровня качества к тому моменту, как программный код будет передан тестировщикам. Purify помогает обнаружить фатальные ошибки доступа к памяти и избежать "аврала" в последние часы сдачи проекта.

Компонентные технологии являются составной частью современных программных приложений, и ошибки в таких компонентах могут оказывать на приложения катастрофическое воздействие.

Обнаружение ошибок программирования, таких как использование неинициализированной памяти, утечек памяти и повторной инициализации уже используемой памяти, составляют особую трудность при разработке ПО с применением компонентов. Необходимо проверять доступ к памяти вне и внутри компонентов; в противном случае можно потратить огромное количество времени на исправление программной ошибки, которая находится не в написанном вами коде, а скрыта в самом компоненте.

Rational Purify способен автоматически выявить источник ошибки в исходном тексте. Если доступен исходный текст, то непосредственно из Rational Purify может быть запущен редактор, с помощью которого можно быстро исправить обнаруженные ошибки.



Работа с Rational Purify

Запатентованная технология Object Code Insertion позволяет Purify выявлять фатальные и разрушающие ошибки доступа к памяти не только в исходном тексте, но и в библиотеках. При этом необязательно наличие их исходного текста. Purify проверяет компоненты, включая элементы управления ActiveX, COM/DCOM-компоненты, ODBC-компоненты, DLL-библиотеки, компоненты сторонних разработчиков, а также любые компоненты C++ или Java в приложениях, поддерживающих OLE. С помощью Rational Purify можно добиться высокого качества, тщательно тестируя приложение и выявляя дефекты в работе компонентов.

Java-программисты и тестировщики могут использовать Rational Purify совместно с виртуальными машинами Java, повышая эффективность использования памяти в Java-апплетах и Java-приложениях.

Rational Purify может выполнять апплеты и целые Java-приложения. Аналогичным образом Rational Purify оказывается полезным при анализе использования памяти в приложениях, создаваемых по технологии Microsoft .NET.

Использовать Rational Purify исключительно легко: он не отвлекает разработчика от непосредственной работы по созданию кода. Поскольку он легко интегрируется в популярные среды разработки приложений (Microsoft Visual Studio, IBM WSAD, Eclipse IDE), то можно в автоматическом режиме использовать Purify и редактировать исходный код при возникновении ошибок прямо в среде. Таким образом, нарушение привычного процесса разработки сводится к минимуму, и практически нет никакой потери в скорости программирования. При этом достигается значительный выигрыш в общей эффективности работы. Активное тестирование системы и подсистем в реальном времени — мощное дополнение к функциональному и нагрузочному тестированию.

Вообще говоря, тестирование качества приложений имеет несколько измерений: функциональность, надежность, производительность и др. Достаточно часто организации, проверяющие качество, обнаруживают проблемы с надежностью только тогда, когда они случайно наталкиваются на них в ходе функционального тестирования. Программные ошибки, связанные с доступом к памяти и ответственные за сбой приложения, могут проявляться не на каждом компьютере, на котором исполняется такое приложение. В ходе разработки и тестирования эти ошибки могут оставаться скрытыми, а симптомы проявятся только после того, как пройдет некоторое время (иногда через несколько месяцев). Когда указанные ошибки проявятся, для решения подобных проблем придется создавать заплатку за заплаткой. Разумеется, лучше предотвратить появление таких ошибок.

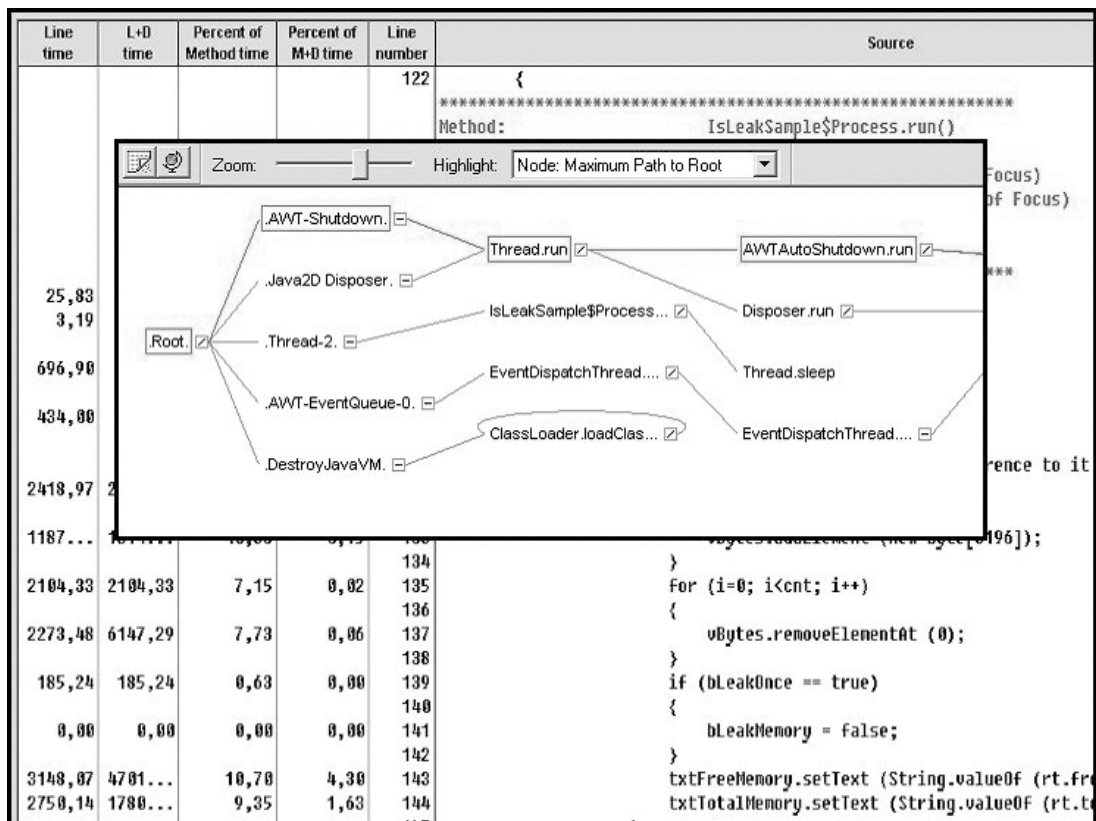
Помимо нахождения критических ошибок Rational Purify также может сообщить, насколько глубоко проведено тестирование программного кода в режиме реального времени. Если установлен Rational PureCoverage, то Purify может автоматически собирать данные о степени тестирования кода при каждом запуске. Это повысит уверенность в том, что осуществлен полный поиск ошибок.

Интеграция Purify с Rational ClearQuest позволяет сохранить информацию об обнаруженной ошибке в общем репозитории запросов на изменения.

Интеграция Purify с Rational TestManager позволяет выполнить анализ в реальном времени в процессе общего тестирования системы. Таким образом, можно провести анализ утечек памяти (C/C++ и др. приложения, при работе которых не задействованы “сборщики мусора” – Garbage Collectors) или профайлинг памяти (Java и .NET приложения).

IBM Rational Quantify

Rational Quantify позволяет точно определить узкие места в производительности приложения. Rational Quantify предлагает графическое отображение данных производительности, с помощью которого легко оценить, какие участки кода являются наиболее критичными с точки зрения производительности.



Работа с Rational Quantify

Проблемы с производительностью приложения возникают по множеству различных причин. Среди них — проблемы в проектировании, проблемы в кодировании, чрезмерные или ненужные вызовы процедур. Обнаружение таких медленных функций и их коррекция — от самых медленных к самым быстрым — обычно не является эффективным способом улучшения производительности приложения.

Rational Quantify помогает быстро и эффективно добраться до реального источника проблем, связанных с производительностью. С помощью утилиты Call Graph можно увидеть целостную картину с графическим интуитивно понятным представлением последовательности вызовов функций. Жирные линии на графическом представлении покажут, какая последовательность вызовов функций потребовала больше всего времени при выполнении программы, а также важность конкретной функции с точки зрения производительности системы.

При разработке ПО с использованием компонентов обнаружение “узких” мест производительности особенно затруднено. При этом важно выявить, где именно находятся эти “узкие” места. Без Rational Quantify можно потратить недели в попытках решить проблему производительности, которая находится вовсе не в написанном коде, а скрыта где-то в стороннем компоненте.

Rational Quantify позволяет непосредственно перейти к тому коду приложения, где обнаружена проблема. С помощью инструмента легко выполнить анализ либо всего приложения, либо его отдельных участков за счет настройки дополнительных фильтров. Фильтры Rational Quantify дают

полный контроль над представлением и организацией данных о производительности, что помогает выборочно отображать те части приложения, которые особенно выиграют от настройки производительности.

Rational Quantify дает возможность установить "узкие" места производительности вплоть до отдельной строки кода.

Thread Analyzer проверяет каждый исполняемый поток и показывает их состояния. Иначе достаточно трудно определить, что каждый поток делает в конкретный момент времени. Rational Quantify дает ответ на этот вопрос в наглядном виде.

Rational Quantify предлагает мощные функции анализа полученной информации. Анализ различий между различными прогонами дает возможность визуально сравнить время выполнения между двумя прогонами, оценить насколько эффективной оказались сделанные в коде изменения по результатам предыдущего тестирования.

Интеграция Quantify с Rational ClearQuest позволяет сохранить информацию об обнаруженной ошибке в общем репозитории запросов на изменения.

Интеграция Quantify с Rational TestManager позволяет выполнить анализ в реальном времени в процессе общего тестирования системы. Таким образом, можно выявить узкие места производительности системы.

Rational PureCoverage

Главная причина раннего завершения многих проектов — обнаруживаемые слишком поздно проблемы с надежностью. В результате стандарты качества часто начинают приноситься в жертву по мере того, как релиз проекта выходит за рамки сроков и бюджета. Можно уменьшить этот риск, пытаясь обнаружить проблемы с надежностью на более ранних этапах цикла разработки.

Rational PureCoverage — средство автоматического определения полноты тестирования кода в приложениях и компонентах, разработанных на Visual C/C++, Java и Visual Basic, C# .NET, Java .NET, Visual Basic .NET. Rational PureCoverage позволяет разработчикам обеспечивать качество и определять эффективность своих усилий в области тестирования. Таким образом, команды разработчиков могут быстрее выпускать качественное ПО, т.е. процесс разработки становится эффективнее.

Независимо от того, производится ли неформальное специальное тестирование или формальное тестирование модуля как части цикла разработки, всеобъемлющее тестирование и быстрая идентификация проблемных областей в процессе разработки являются необходимыми для выпуска качественного ПО. Но без дополнительного инструмента, помогающего определить, какая часть кода

Line Coverage	Line Number	Source
	110	class Process extends Thread
	111	{
	112	public Vector vBytes;
	113	public Process()

Coverage Item	Calls	Methods Missed	Methods Hit	% Methods Hit	Lines Missed	Lines Hit	% Lines Hit
Run @ 05.01.2004 17:36:36 IsLeakSample	67	0	8	100,00	13	73	84,98
(Unknown Directory)	67	0	8	100,00	13	73	84,98
IsLeakSample.java	8	0	7	100,00	13	73	84,98
IsLeakSample\$Action.IsLeakSample\$Action(IsLeakSample)	1		hit		0	1	100,00
IsLeakSample\$Action.actionPerformed(java.awt.event.ActionEvent..)	2		hit		0	4	100,00
IsLeakSample\$Process.IsLeakSample\$Process(IsLeakSample, i..)	1		hit		0	5	100,00
IsLeakSample.IsLeakSample(java.lang.String)	1		hit		0	43	100,00
IsLeakSample.main(java.lang.String [])	1		hit		0	2	100,00
IsLeakSample\$Process.run()	1		hit		3	12	80,00
IsLeakSample.btnStart_Clicked(java.awt.event.ActionEvent)	1		hit		10	6	37,50
(Unknown File)	59	0	1	100,00			

3 876	131	for (i=0; i<=cnt; i++)
	132	{
3 553	133	vBytes.addElement (new byte[8196]);
	134	}
3 553	135	for (i=0; i<cnt; i++)
	136	{
3 230	137	vBytes.removeElementAt (0);
	138	}
323	139	if (bLeakOnce == true)
	140	{
0	141	bLeakMemory = false;
	142	}
323	143	txtFreeMemory.setText (String.valueOf (rt.freeMemory ());
391	144	txtTotalMemory.setText (String.valueOf (rt.totalMemory ());

Работа в Rational PureCoverage

была проверена, невозможно эффективно протестировать приложение. Есть вероятность пропуска кода в процессе тестирования, а значит, вероятность пропуска критических ошибок.

Rational PureCoverage устраняет случайный характер тестирования и помогает более полно тестировать код. Являясь важным инструментом разработки приложений и компонентов, Rational PureCoverage позволяет производить тестирование до или во время важных этапов разработки, например проверки кода, тестирования модуля, в процессе интеграции и определять неработающий код и упущения в тестировании.

Используя PureCoverage с начальных этапов процесса разработки для нахождения и устранения упущений в формальном и неформальном тестировании, можно проверить выполнение всего кода и получить возможность обнаружить ошибки.

Rational PureCoverage позволяет объединять и сравнивать данные о покрытии после многочисленных запусков одного и того же исполняемого кода для получения агрегированной информации о полноте тестирования. Это дает возможность оценить параметры тестирования и обеспечить полноту тестирования всего кода после многочисленных его запусков.

Интеграция PureCoverage с Rational ClearQuest позволяет сохранить информацию об обнаруженной ошибке в общем репозитории запросов на изменения.

Интеграция PureCoverage с Rational TestManager позволяет выполнить анализ в реальном времени в процессе общего тестирования системы. Таким образом, можно определить степень покрытия тестами разрабатываемого ПО.

TestFactory

Rational TestFactory — инструмент для автоматической генерации полного набора тестов для разрабатываемой информационной системы.

Оценить качество ПО можно по трем различным критериям:

- надежность — работает ли приложение без сбоев, зависания или вызова исключений;
- функциональность — делает ли приложение то, что от него требуется;
- производительность — работает ли приложение с приемлемой скоростью при доступе к нему многих пользователей.

Первым и основным критерием в этом списке является надежность. Нельзя успешно протестировать функциональность или производительность приложения, которое регулярно "зависает", и разумеется, оно не может быть поставлено конечным пользователям.

Rational TestFactory шаг за шагом проводит через процесс устранения дефектов и повышения надежности. Тестирование надежности включает в себя проверку всех вариантов исполнения приложения. Поскольку в приложении может потенциально существовать огромное число путей выполнения, то проблема состоит в том, чтобы создать тесты, которые проверяют все части приложения за минимальное количество шагов. Новаторство TestFactory заключается в автоматическом создании таких "наилучших сценариев". TestFactory строит карту приложения, а затем Rational® Robot автоматически генерирует тестовые скрипты. После этого TestFactory использует метрики покрытия кода для определения того, какая часть системы была протестирована.

Поскольку с помощью TestFactory тестовые скрипты генерируются автоматически, то тестировщики могут протестировать каждую сборку, проверяя ее на надежность, независимо от стадии разработки.

Rational TestFactory является наиболее эффективным инструментом для устранения дефектов и повышения надежности, но в полной мере его достоинство становится очевидным при использовании как части всеобъемлющего решения по контролю за качеством ПО.

Организация совместной работы на основе IBM Team Unifying Platform

Инструментарий IBM Rational Team Unifying Platform предназначен для организации совместной работы в составе многофункциональных коллективов с целью повышения производительности как команды в целом, так и отдельных специалистов. Использование средств коллективной разработки IBM Rational помогает синхронизировать работу разнопрофильных специалистов, вовремя предупреждая всех участников проекта об изменениях.

Довольно часто члены команды проекта взаимодействуют недостаточно эффективно, что замедляет обмен информацией и служит причиной потери многих часов рабочего времени. Благодаря полному набору средств для ведения коллективной разработки, тесно интегрированных друг с другом, все члены проектной команды получают общее видение проекта. Специалисты более не изолированы и не ограничены рамками выполняемой функции. Инструментарий IBM Rational предупреждает специалистов по тестированию о том, что им необходимо изменить сценарий тестирования, разработчиков — об изменении приоритетов требований, а руководителя проекта — о превышении установленного предела серьезных ошибок. Такой обмен информацией крайне важен для повышения продуктивности в работе над проектом.

Набор средств IBM Rational под названием Team Unifying Platform (Объединяющая платформа) включает в себя следующие средства совместного управления проектами и создания коллективной инфраструктуры:

- IBM Rational Unifying Process — методология создания информационных систем, обеспечивает всеобщее понимание и следование итеративному подходу;
- IBM Rational RequisitePro — средство управление требованиями, которое позволяет коллективу справляться с постоянно меняющимися требованиями;
- IBM Rational ClearQuest — средство управления запросами на изменения, предоставляет возможности для отслеживания дефектов и изменений, которое позволяет каждому понимать текущий статус качества ПО;
- IBM Rational ClearCase — средство конфигурационного управления, надежный инструмент для управления версиями, который служит в качестве общего репозитория для всех объектов разработки ИС;
- IBM Rational TestManager — средство организации процессов тестирования, которое объединяет в себе планирование тестов, их выполнение и анализ, включая частные и унаследованные тестовые объекты;
- IBM Rational Project Console - средство мониторинга ключевых показателей проекта, которое упрощает контроль за статусом проекта и генерирование объективных метрик проекта с целью улучшения его предсказуемости.
- IBM Rational SoDA — средство документирования, автоматизирует создание и сопровождение всеобъемлющей проектной документации и отчетов

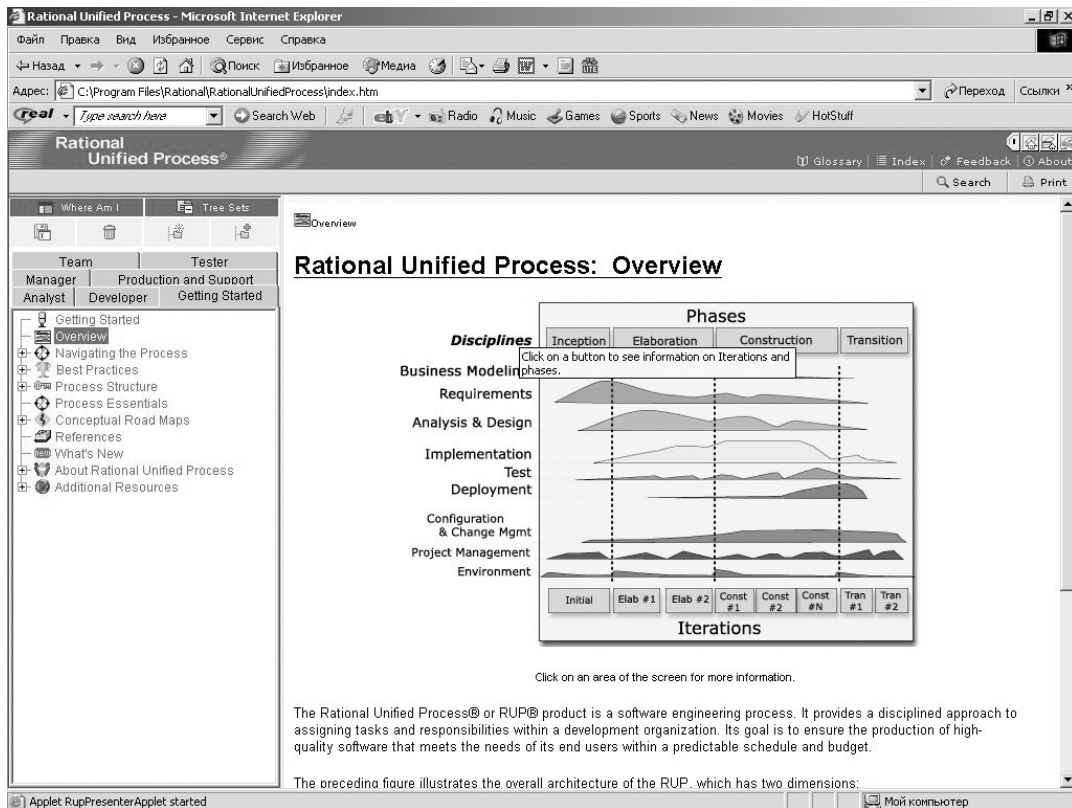
В основе работы инструментария лежит подход Unifying Change Management (UCM), который позволяет организовать индивидуальные рабочие пространства для каждого участника проекта на базе единого проектного репозитория. В зависимости от специфики деятельности конкретного участника проекта он использует необходимые инструментальные средства. Интеграция указанных средств позволяет объединить деятельность всей команды в единый и четкий процесс построения эффективной ИС.

Управление процессом разработки. IBM Rational Unified Process.

В каждом проекте разработчики стремятся к общей цели — в утвержденные сроки создавать качественное и надежное ПО, отвечающее текущим требованиям и масштабируемое для будущих задач. Достичь этой цели практически невозможно без хорошо организованного процесса разработки. IBM Rational Unified Process (RUP) предоставляет группам разработчиков возможность общего, интерактивного и интегрированного управления процессом.

RUP — обширный справочник в виде портала, который организует работу проектной команды. RUP содержит инструкции, шаблоны и примеры для большинства критических задач, которые могут возникнуть в ходе проекта.

RUP — практическое руководство, помогающее разработчикам выполнять их повседневные задачи. Реализованный с использованием HTML и Java для простого и независимого от платформы доступа с локального компьютера и через корпоративную сеть, продукт RUP имеет мощные графические средства навигации, позволяющие быстро находить необходимую информацию по разработке ПО, а также шаблоны для документов, создаваемых в ходе проекта.

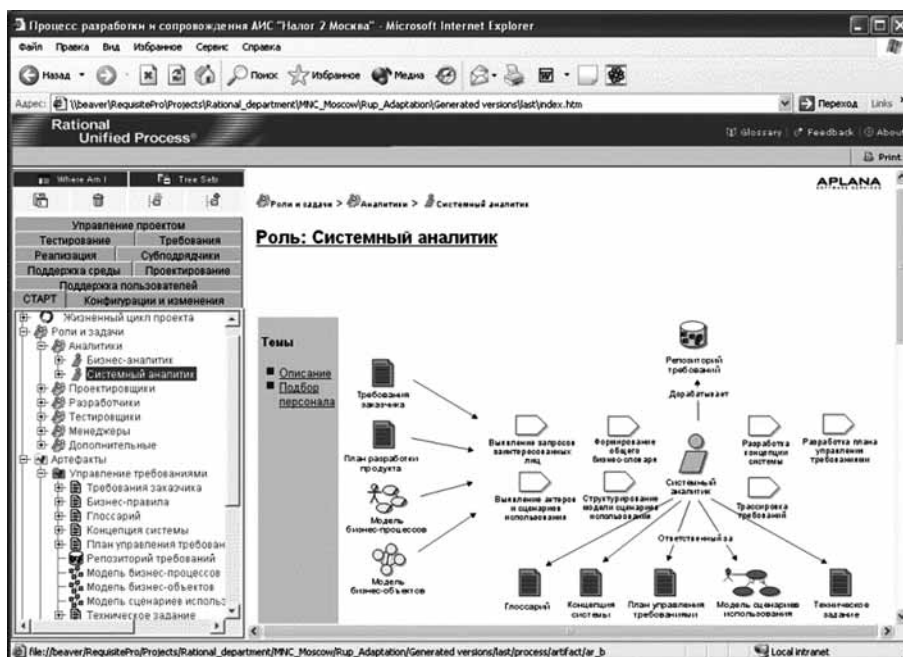


RUP – база знаний

RUP позволяет интегрировать функции инструментальных средств IBM Rational:

- руководства по инструментальным средствам описывают, как эффективно использовать эти средства для реализации конкретных задач в различных фазах разработки;
- расширенная помощь облегчает поиск инструкций для реализации текущей задачи.

RUP может быть адаптирован как для малых, так и для больших проектов и использоваться при разработке приложений всех типов, включая электронную коммерцию (e-business) и Web-приложения, приложения реального времени и встроенное ПО.



Пример адаптированного Web-сайта RUP

Существует два разных подхода к внедрению RUP: целиком или по шагам. Наиболее эффективно внедрять весь процесс целиком, когда начинается новый проект, участники проекта хорошо владеют методологией RUP или некоторый проект ведется настолько хаотично, что любая его реорганизация заведомо принесет пользу. Предпочесть внедрение RUP по шагам стоит, если речь идет о внесении изменений в уже существующий проект, в котором есть налаженные механизмы командного взаимодействия или участники проекта плохо знакомы с положениями RUP.

В каждом конкретном случае целесообразно подобрать свои эффективные пути внедрения новых методов работы. В этом могут помочь специализированные консалтинговые компании, имеющие обширный опыт внедрения средств и методологии RUP – это поможет избежать лишних затрат на определение собственного пути.

Для адаптации методологии RUP в организации IBM Rational предлагает средства, поддерживающие задачи моделирования и описания процессов разработки. Эти средства представляют собой набор инструментов для настройки и публикации Web-сайтов на основе RUP, которые позволяют вносить изменения в RUP и описывать собственные стандарты ведения проектов:

- IBM RUP Builder - средство автоматической генерации Web-сайта модифицированного RUP, которое входит в стандартную поставку IBM Rational Suite Enterprise. При разработке сайта можно использовать готовые модули расширения (plug-in), которые могут встраиваться в текущее описание процесса, добавляя в него некоторую специфику. Готовые модули расширения можно загрузить с сайта IBM Rational по ссылке: <http://www-130.ibm.com/developerworks/rational/library/5823.html>
- IBM Rational XDE Modeler - визуальное средство создания взаимосвязи процессов на основе метамодели RUP, повышающее уровень абстракции и облегчающее понимание и изменение процессов (возможно также использование IBM Rational XDE Developer for Java или IBM Rational XDE DeveloperPlus for Java)
- IBM Rational Process Engineering - средство формирования контента сайта, может быть бесплатно получен с сайта IBM Rational.

Адаптация RUP проводится инженерами-технологами организации или внешними консультантами и включает этапы определения, описания и представления процессов. В качестве основы для определения процесса используется модель RUP, затем выполняется изменение и расширение базовой модели. Визуализация связей между элементами процесса показывает, например, какие артефакты задействованы в процессе и какие роли отвечают за их создание. Библиотека элементов процесса содержит текстовую информацию о каждом элементе в модели процесса. Библиотека содержит также все текстовые страницы RUP, а также необходимые шаблоны для создания новых страниц описания. На последнем этапе — этапе представления процесса, генерируется описание процессов, включающее текст и графику в виде Web-сайта и соединяющее модели процессов и библиотеку описаний в единое целое.

Внедрение RUP может быть проведено

- во всей организации с распространением на все проекты;
- в отдельном проекте;
- в отдельных подразделениях, участвующих в одном или нескольких проектах организации;
- в подразделениях, задействованных в какой-либо части крупных проектов, ведущихся в других компаниях (аутсорсинг).

Начинать проект по адаптации RUP рекомендуется с внедрения части RUP в одном или нескольких подразделениях, а в дальнейшем распространить успешный опыт на организацию в целом. Таким образом, можно организовать на основе RUP эффективные процессы визуального моделирования и управления требованиями, создания прототипов графического интерфейса, планирования работ и определения необходимых ресурсов, автоматизированного тестирования, управления изменениями (дефектами, запросами на улучшение, задачами и т. д.), управления версиями и конфигурацией.

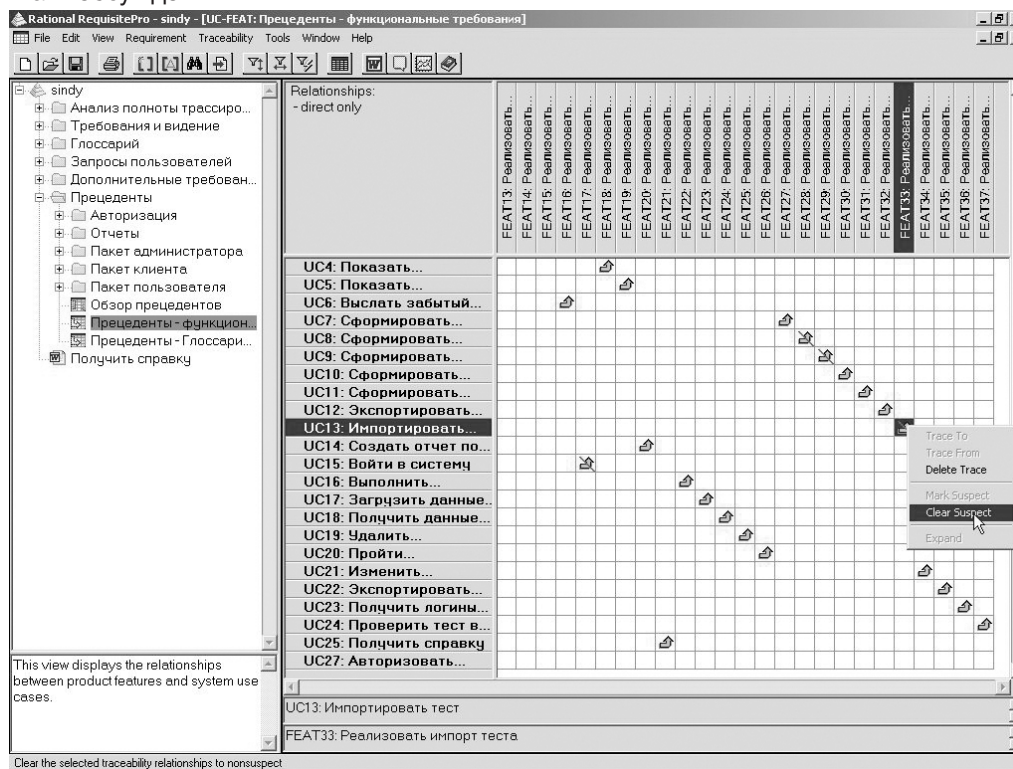
Управление требованиями. Rational RequisitePro

Успешный опыт разработки показывает, что эффективное управление требованиями является ключевым фактором всего процесса разработки ПО. Требования определяют то, что должна делать система. Поэтому в течение всего жизненного цикла проекта нужно организовать эффективную работу с ними. Первым шагом в этом направлении служит организация хранения всех выявленных требований.

Основные цели процесса управления требованиями:

- понять структуру и динамику предметной области, в которой должна быть развернута создаваемая информационная система;
- понять текущие проблемы предметной области и определить потенциальные возможности ее усовершенствования;
- обеспечить общее понимание предметной области заказчиками, конечными пользователями и разработчиками;
- выявить системные требования, необходимые для поддержки автоматизации предметной области;
- установить и поддержать соглашение с клиентами и другими заинтересованными лицами на том, что система должна делать;
- обеспечить разработчиков системы лучшим пониманием требований к ее созданию;
- определить функциональные границы создаваемой информационной системы;
- обеспечить базис для планирования технического содержания фаз разработки;
- обеспечить базис для оценки стоимости и времени на разработку информационной системы;
- определить графические интерфейсы пользователей с учетом их потребностей и целей.

Основным инструментом для организации работы с требованиями в проекте является IBM Rational RequisitePro. Он позволяет команде работать с требованиями, отслеживать возможные изменения в них и организовывать обсуждения.



Работа в Rational RequisitePro

Rational RequisitePro дает возможность организовать контролируемый доступ к требованиям, позволяет всем членам команды проекта быть в курсе текущих требований, их контекста и связей и совместно трудиться над созданием "правильного" продукта, создавать его быстро и качественно.

Благодаря удаленному доступу и привычным способам навигации Интернет стал идеальной средой доступа к данным для совместно работающих команд. Особенно это касается команд, работающих в распределенной или многоплатформенной среде разработки. Rational RequisitePro предоставляет возможность работы как в режиме стандартного интерфейса Windows, так и Web-интерфейса (Rational RequisiteWeb).

Rational RequisitePro через Web-интерфейс открывает доступ к требованиям всем участникам команды, работающим удаленно или на разных платформах. Rational RequisiteWeb позволяет организовать доступ к требованиям, запросы необходимых требований, их добавление и коррекцию через обычный Web-браузер.

Кроме того, члены группы могут участвовать в онлайн-обсуждениях. Любая дискуссия может быть ассоциирована с определенным требованием, набором требований или проектом в целом. Участие в дискуссии может осуществляться через Web- или Windows-клиента, а также по электронной почте, благодаря чему состав участников может быть расширен за счет представителей заказчика, не использующих RequisitePro. Дискуссии как способ выработки предложений и изменений представляют собой открытый и контролируемый механизм выяснения вопросов, обсуждения предложений, совместной работы над идеями и уведомления об изменениях.

Документы являются естественной средой, в которой можно создавать требования, указывая для них контекстную и дополнительную информацию в привычном редакторе Microsoft Word. При этом все требования хранятся в отдельной базе требований. И когда требования создаются в Microsoft Word, в действительности просто образуется их связь с требованиями в базе требований.

При этом с требованиями в документах можно работать таким же образом, как и с обычными требованиями в базе, устанавливая для них дополнительную информацию, определяя их приоритеты реализации и любые иные атрибуты. Такая уникальная и надежная архитектура объединяет возможности базы данных и простоту использования Microsoft Word для оптимизации процесса управления требованиями.

Архитектура Rational RequisitePro является крайне гибкой, что позволяет отделять документы от базы требований без ущерба для дальнейшей работы или доступа других членов команды. Авторы требований, подрядчики или иные лица, внешние по отношению к организации, могут составлять, просматривать, редактировать и работать иным способом с этими документами без необходимости доступа к базе требований. Эти документы автоматически синхронизируются с проектом при их возвращении в проект.

Rational RequisitePro позволяет создавать базы требований с использованием СУБД Oracle, Microsoft SQL Server, SQL Sybase Anywhere Server и Microsoft Access.

RequisitePro дает возможность определить необходимые типы требований и стандартные атрибуты для них в случае нового проекта. Дополнительные атрибуты (приоритет, сложность, состояние, ответственный за реализацию и др.) помогают управлять требованиями намного эффективнее. Это было бы невозможно с помощью ведения одних только документов требований.

RequisitePro предусматривает механизмы для визуального контроля за изменениями в требованиях. Это позволяет точно определить влияние любых изменений в требованиях или в документах на остальные артефакты проекта. В Rational RequisitePro автоматически ведется история изменений требований, что помогает отслеживать их эволюцию в ходе проекта. В результате всегда можно получить информацию о том, кто, когда, зачем и какие конкретно изменения вносил в проектные требования.

Значительно облегчается процесс анализа требований для выявления тенденций их изменения, получения отчетов о степени готовности и оценок состояния проекта. Последнее возможно через определение фильтров, создание запросов и отчетов по требованиям и последующим экспортом полученных данных в Microsoft Excel, где строятся необходимые аналитические диаграммы и таблицы.

Объединяя мощь базы данных с удобством Word, инструментальный продукт Rational RequisitePro предлагает надежное и легкое в использовании решение в области управления требованиями. Интеграция с другими инструментальными средствами IBM Rational предоставляет мощные средства по управлению ходом проекта.

Web-интерфейс RequisitePro (RequisiteWeb), позволяет организовать эффективную удаленную работу с требованиями.

Интеграция Rational RequisitePro с Rational Software связывает требования с визуальными моделями. Участники проекта могут легко формировать требования верхнего уровня на основании существующих прецедентов и детализировать эти требования с помощью анализа сценариев указанных прецедентов.

Интеграция Rational RequisitePro с Rational Rose связывает требования с визуальными моделями. Участники проекта могут легко формировать требования верхнего уровня на основании существующих прецедентов и детализировать эти требования с помощью анализа сценариев указанных прецедентов.

Интеграция Rational RequisitePro и Rational ClearQuest облегчает внесение изменений в проект, связывая запросы на изменения с требованиями, определенными на базе первых. При этом легко получить набор требований, связанных с конкретным запросом на изменение, и наоборот, набор запросов на изменения, связанных с конкретным требованием.

Интеграция Rational RequisitePro с Rational ClearCase позволяет организовать эффективный версионный контроль проектных требований.

Интеграция Rational RequisitePro с Rational TestManager гарантирует, что на основе существующих требований тестировщиками или специалистами по контролю качества будут созданы необходимые сценарии тестирования.

Интеграция Rational RequisitePro с Rational SoDA позволяет автоматизировать создание проектных отчетов по требованиям, используя шаблоны оформления документов, принятые в организации.

Интеграция Rational RequisitePro с Microsoft Project позволяет наладить планирование в проекте в кратчайшие сроки. Требования — это то, что должно быть реализовано, а значит, на их основе можно определить задачи для участников проекта и спланировать их выполнение.

Управление конфигурациями и изменениями. IBM Rational ClearCase, IBM Rational ClearQuest

Перефразируя модель зрелости процессов Института программной инженерии (SEI CMM), можно сказать, что конфигурационное управление и управление изменениями обеспечивает контроль за изменениями и обеспечивает взаимосвязь артефактов проекта. Методы, процессы и инструментальные средства, используемые для обеспечения конфигурационного управления и управления изменениями в организации, могут рассматриваться как единая система конфигурационного управления.

Конфигурационное управление и управление изменениями включают:

- идентификацию объектов конфигурационного управления;
- ограничение возможности изменения этих объектов;
- аудит изменений, произведенных с объектами конфигурационного управления;
- определение конфигураций ОКУ и управление этими конфигурациями.

Система конфигурационного управления является необходимой и неотъемлемой частью всего процесса разработки и содержит ключевую информацию о процессах разработки продуктов, их развитии, развертывании и внедрении. Она сохраняет для повторного использования артефакты, получаемые в ходе выполнения проекта.

Инструментом, позволяющим организовать эффективное управление версиями и конфигурациями является ClearCase, а управления изменениями – ClearQuest.

Rational ClearCase

Rational ClearCase является инструментом, который значительно упрощает ведение процесса управления версиями и конфигурациями. Он помогает наладить эффективный контроль за любыми артефактами проекта: документами, исходниками, моделями, целыми репозиториями проекта, дополнительными файлами и т.д. Все, что создается и меняется в процессе разработки, может быть поставлено на версионный и конфигурационный контроль. Таким образом, в значительной мере повышается надежность и качество работы проектной команды.

Задача управления различными конфигурациями, релизами и версиями возникает в проектах любого уровня и масштаба. Все усложняется, когда двое или более разработчиков в одно и то же время работают над одним и тем же артефактом. Такая проблема особенно актуальна, если эти разработчики территориально удалены друг от друга и не могут оперативно согласовывать свою деятельность.

Очень важно при этом наладить четкий контроль за работой участников проекта, которые работают с его репозиториями. Это могут быть, например, менеджеры проекта, аналитики, разработчики, тестировщики и многие другие. Очень важно четко представлять, кто и что делает в проекте, кто и что изменил в тот или иной момент времени. Не менее важно, чтобы в проекте всегда была возможность откатиться до некоторого предыдущего состояния, если разработка пошла по ошибочному пути. Это может относиться как к проекту в целом, так и отдельным его частям.

С помощью ClearCase можно организовать версионный контроль не только отдельных файлов, но и целых каталогов. Таким образом, можно создавать новые версии целых рабочих пространств и окружений.

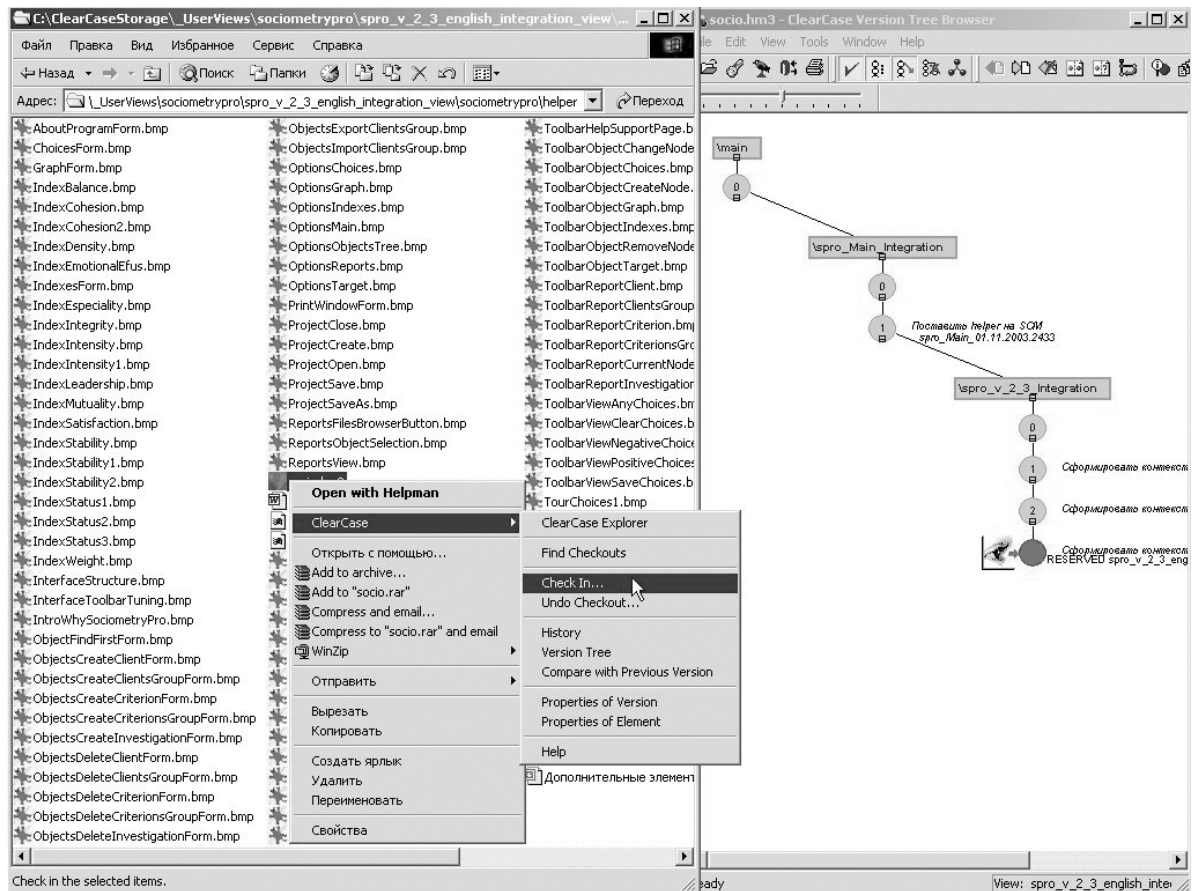


Рис. 43 Работа с Rational ClearCase

Для каждого артефакта, поставленного на контроль, ведется история изменений. К любой версии можно вернуться в любой момент. Для любых двух разных версий одного и того же артефакта можно посмотреть отличия, если, конечно, эти артефакты принадлежат стандартным типам документов, таким как текстовые файлы, документы Microsoft Word и модели Rational Rose.

Мощные возможности ClearCase в области конфигурационного управления позволят организовать параллельную работу над отдельными конфигурациями и версиями одного и того же продукта или его части.

ClearCase значительно ускоряет командную разработку, обеспечивая поддержку параллельной работы. В этом значительно помогут мощные возможности в организации ветвления версий. Всегда можно провести интеграцию нескольких ветвей в основную. Для этого существуют удобный графический инструмент, автоматически запускающийся, если требуется осуществить подобный процесс интеграции. В ходе интеграции участник проекта может в автоматическом или ручном режиме указать, что следует включить в конечную версию того или иного артефакта.

Данные, поставленные на версионный и конфигурационный контроль, под управлением ClearCase хранятся в базах данных специального формата, т.н. Version Object Bases (VOB). Участники проекта имеют доступ к любой из этих VOB через «представления» (Views), каждое из которых отображает некоторый срез проектных данных из VOB. Таким срезом может быть, например, набор артефактов, относящихся к некоторой версии или конфигурации разрабатываемого продукта.

Представления бывают двух типов: динамические и статические. Динамическое представление (dynamic view) отображает артефакты, актуальность которых гарантируется в любой момент времени. Здесь имеется в виду, что если один из участников проекта изменил один из артефактов, то любой другой участник проекта будет работать всегда с последним вариантом этого артефакта. Но для работы с динамическими представлениями требуется постоянное подключение к серверной части ClearCase. Статическое представление (snapshot view) представляет собой своеобразный снимок набора артефактов и позволяет работать с ними без постоянного подключения к серверу.

Существует вариант Rational ClearCaseLT, предназначенный для небольших рабочих групп. Его основные отличия от полнофункционального ClearCase:

- Отсутствие динамических видов.
- Работа в однодоменном окружении.
- Отсутствие поддержки ClearCase Multisite.
- Отсутствие возможности создания мультипроектной среды.

Когда у организации появляется необходимость в конфигурационном управлении на уровне предприятия, например, в расширенном управлении процессами сборки, использовании распределенных версионных хранилищ, использовании удаленной репликации данных, то легко провести модернизацию и перейти на полную версию ClearCase. При этом не потребуется изменять определенные процессы работы, какие-либо данные или подходы.

ClearCase включает мощный Web-интерфейс, с помощью которого легко организовать удаленную работу с артефактами проекта.

Для географически распределенных рабочих групп, работающих совместно в одних и тех же проектах, будет крайне полезен Rational ClearCase MultiSite, являющийся стандартом для организации удаленной совместной работы и обеспечивающий непрерывный параллельный процесс разработки. Технология расширенной репликации позволяет распределенным командам работать одновременно в проектах с локальным доступом к актуальным рабочим материалам. ClearCase плавно интегрируется в следующие рабочие среды: Microsoft Visual Studio, IBM VisualAge for Java, IBM WebSphere Studio, Sybase PowerBuilder, Microsoft Word и др. Специальные настройки графического интерфейса позволяют разработчикам в большей степени концентрироваться на решении конкретных задач, не отвлекаясь на примитивные рутинные процедуры.

Объединение процессов конфигурационного управления и управления изменениями с помощью механизмов Unifying Change Management (UCM) значительно повышает возможности проекта в области контроля изменений в отдельных артефактах. Являясь функциональной надстройкой над ClearCase, UCM автоматизирует многие операции, обеспечивая параллельную работу с артефактами и их наборами.

Интеграция ClearCase и Rational ClearQuest гарантирует, что никто не внесет изменения в артефакты проекта, пока для него не определена соответствующая задача.

Интеграция ClearCase с Rational Rose позволяет ставить на версионный контроль целые модели или их части прямо из среды визуального моделирования.

Интеграция ClearCase с Rational RequisitePro позволяет организовать процесс управления версиями и конфигурациями для всего репозитория требований. В любой момент времени могут быть восстановлены наборы документов и требований для любой версии продукта в необходимой конфигурации.

Интеграция ClearCase с Rational TestManager позволяет аналогичным образом наладить версионный и конфигурационный контроль над всем репозиторием тестирования. В любой момент времени можно восстановить сценарии и тестовые скрипты для проведения процесса тестирования конкретной версии в необходимой конфигурации.

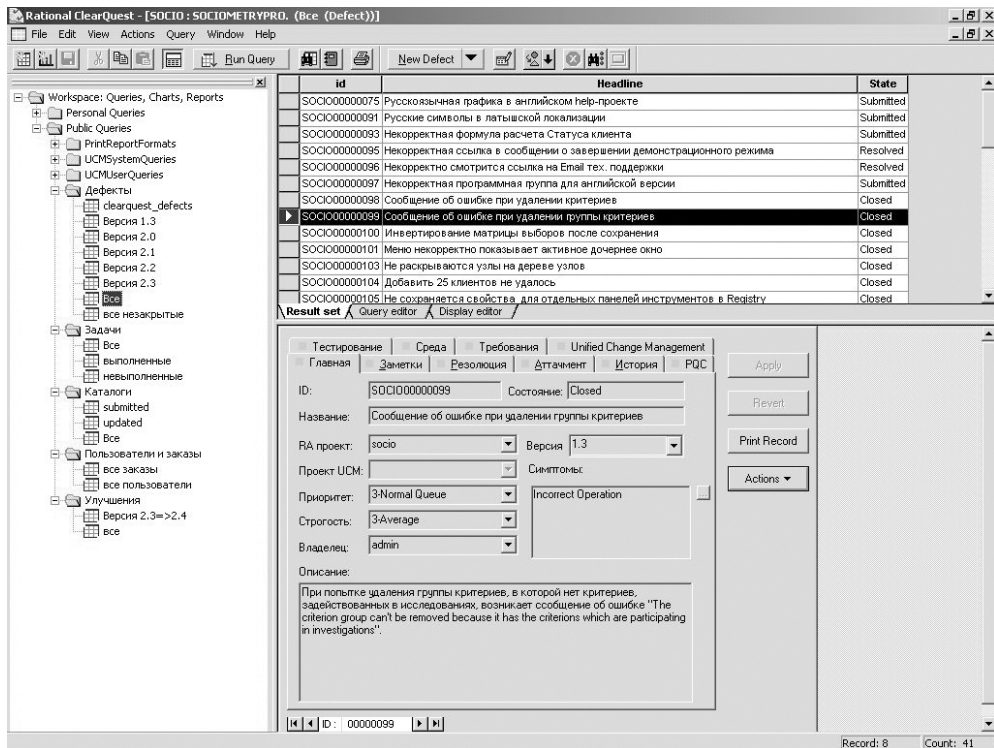
Интеграция ClearCase с Rational SODA позволяет автоматизировать генерацию документов по артефактам проекта и изменениям, связанными с ними.

Интеграция ClearCase с Eclipse IDE и продуктами, базирующимися на Eclipse предоставляет доступ к возможностям использования версионного и конфигурационного контроля для широкого круга дочерних продуктов, таких как XDE Modeler, XDE Developer for Java, IBM Websphere Studio Application Developer, средства Atlantic и т.д.

Rational ClearQuest

Rational ClearQuest представляет собой мощный инструмент для отслеживания всех типов запросов на изменения для любого проекта. Возможность создавать новые типы запросов на изменения, обладающих собственными атрибутами и имеющих нестандартный жизненный цикл, позволяет адаптировать ClearQuest в любой организации.

Как показывает практика, чтобы обеспечить высокое качество создаваемого продукта, недостаточно просто наладить поиск ошибок. Для этого необходимо обеспечить эффективный процесс управления всеми типами запросов на изменения, возникающих в процессе разработки. К таким запросам относятся, в первую очередь, запросы на улучшение самого продукта или процесса по его созданию,



Работа с Rational ClearQuest

описания ошибок, выявленных в продукте, и любые поручения к участникам проекта. Поскольку все запросы на изменение фиксируются в едином репозитории, то появляется возможность наладить четкий контроль за работой коллектива и осуществлять непрерывный мониторинг проекта.

Размер проекта по разработке информационных систем при этом неограничен, т.к. репозиторий запросов на изменения может быть создан на базе известных СУБД, таких как Microsoft SQL Server, Oracle Server, IBM DB2 Server, Sybase SQL Anywhere Server. Для небольших команд вполне подойдет возможность создания репозитория на базе Microsoft Access.

Web-интерфейс ClearQuest, известный как ClearQuestWeb, позволяет взаимодействовать участникам проекта удаленно практически с любой программно-аппаратной платформы. Кроме того, наличие клиентов для Linux, Sun Solaris, AIX, HP/UX и др. предоставляет мощные возможности для организации процесса запроса на изменения в проекте, в котором происходит создание распределенной многоплатформенной ИС.

Для распределенных команд предлагается расширение в виде Rational ClearQuest Multisite, который на базе реплицирования репозитория запросов на изменения предоставит этим командам полный набор возможностей, предлагаемых инструментарием, что недоступно при использовании Web-интерфейса.

ClearQuest предоставляет полностью документированный интерфейс API на базе COM. Это дает возможность интегрировать ClearQuest, например, с центром приема и обработки телефонных звонков или иными системами, участвующими в бизнес-процессе организации.

Интеграция ClearQuest с Rational RequisitePro позволяет связать требования к разрабатываемой ИС с конкретными запросами на изменение. Таким образом, если появились новые требования или изменились существующие, то легко определить, какие запросы на улучшение или обнаруженные ошибки в системе послужили источником для этого.

Интеграция ClearQuest с Rational ClearCase предоставляет возможность наладить контроль не только за запросами на изменения, но и непосредственно за любыми изменениями в артефакты проекта. В этом случае никто не сможет изменить ни один артефакт до тех пор, пока для него не определена соответствующая задача. Таким образом, значительно повышается контроль за ходом процесса разработки.

Интеграция ClearQuest со средствами тестирования позволяет вносить описания обнаруженных ошибок прямо по его результатам. Таким образом, можно создать описание ошибки прямо из лога тестирования, сформированного в Rational TestManager, или из отчета по результатам анализа, выполненного в реальном времени с помощью Rational Purify, Rational Quantify, Rational PureCoverage.

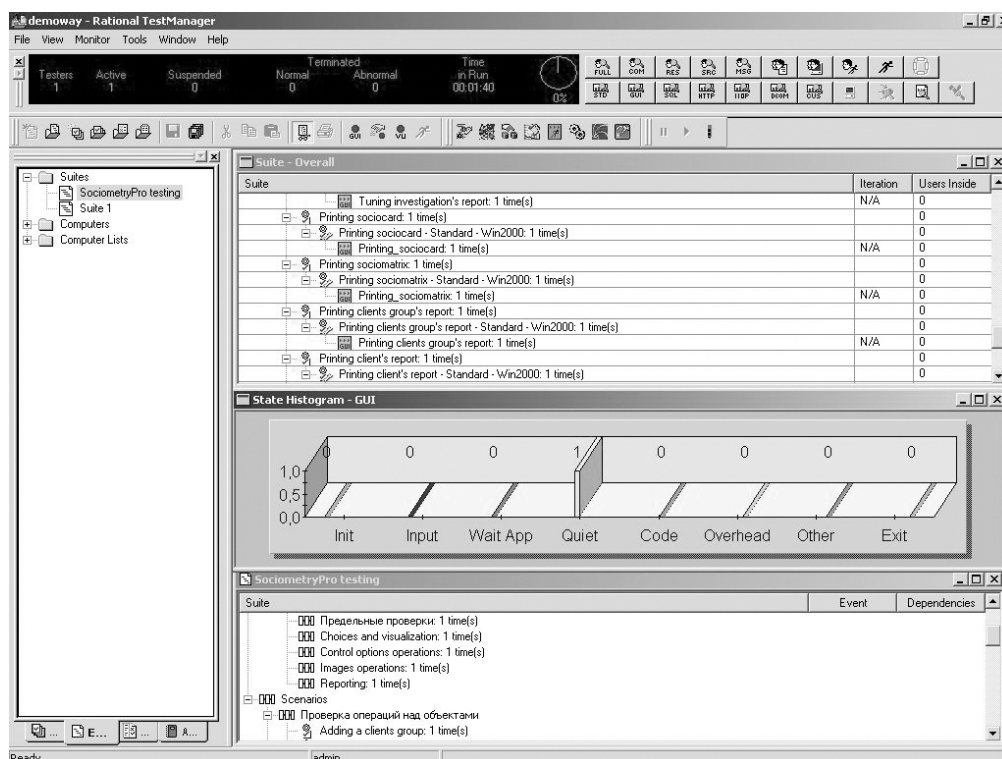
Интеграция ClearQuest с Rational SODA позволяет автоматически формировать отчеты в необходимом виде по любым запросам на изменения.

Интеграция ClearQuest с Microsoft Project позволит быстро сформировать список задач для участников проекта, детализировать этот список до конкретных поручений и реально отслеживать, не выходит ли проект за определенные в плане проекта сроки, ресурсы и бюджет.

Управление тестированием. IBM Rational TestManager

Специализированный инструмент для организации тестирования в составе Team Unifying Platform - Rational TestManager - значительно повышает эффективность процесса тестирования, предоставляя в распоряжение команды общие средства планирования, проектирования, исполнения и анализа результатов тестов. Объединение операций тестирования в единый процесс дает возможность проектной команде наладить эффективный процесс контроля качества создаваемого продукта.

В Rational TestManager ведется план тестирования. Источником для конкретных сценариев тестирования могут служить элементы визуальных моделей и требования. С помощью Rational TestManager организуется единое рабочее пространство, объединяющее элементы плана тестирования, сценарии и тестовые скрипты, отчеты по результатам тестирования и другие необходимые данные.



Работа с Rational TestManager

Rational TestManager связывает наборы тестов с конкретной итерацией процесса разработки и конфигурацией разрабатываемой информационной системы.

Интеграция TestManager с Robot позволяет эффективно автоматизировать процесс тестирования благодаря объединению возможностей планирования и управления распределенным тестированием с мощными возможностями записи и воспроизведения автоматизированных тестов для ИС, построенных практически по любой технологии.

Интеграция TestManager с Eclipse IDE предоставляет богатые возможности для автоматизации процесса тестирования с продуктами, являющимися производными по отношению к Eclipse (XDE Tester, средства тестирования в составе инструментов Atlantic и т.д.).

Интеграция TestManager с Rational Rose позволяет использовать функциональные элементы визуальной модели в качестве источника сценариев тестирования.

Интеграция TestManager с Rational RequisitePro позволяет использовать требования репозитория требований для определения других сценариев тестирования.

Интеграция TestManager с Rational ClearQuest дает возможность прямо заносить описания ошибок в репозиторий запросов на изменения проекта.

Интеграция TestManager с Rational SoDA позволяет наладить процесс автоматической генерации документов по репозиторию тестирования.

Управление выполнением проекта и документированием. IBM Rational ProjectConsole, IBM Rational SoDA

IBM Rational Project Console

IBM Rational ProjectConsole автоматизирует процесс создания и мониторинга отчетов о состоянии проекта благодаря созданию динамического Web-сайта, который отображает панель основных показателей проекта. Это экономит время, необходимое для создания и поддержания Web-сайта команды проекта, а также позволяет избежать сбора подобных данных вручную.

Для руководителей проектов IBM Rational ProjectConsole предоставляет объективную картину достигнутых результатов на протяжении всего жизненного цикла проекта. Rational ProjectConsole собирает фактические данные о состоянии разработки из платформы Rational Suite® и продуктов других поставщиков, представляя эти результаты в графическом виде, что позволяет легко и быстро оценить ход и качество проекта.

Rational ProjectConsole помогает ответить на следующие вопросы:

- Где следует сконцентрировать имеющиеся ограниченные ресурсы, чтобы обеспечить выполнение плана проекта
- Какие тенденции могут негативно повлиять на затраты и график работ
- Насколько стабильной является используемая архитектура.
- Сколько строк программного кода требует внесения изменений/добавления и как это соотносится с графиком выпуска релиза

Интеграция с существующими приложениями Rational позволяет получать и просматривать информацию из IBM Rational Rose®, IBM Rational RequisitePro®, IBM Rational ClearQuest®, IBM Rational ClearCase®, IBM Rational XDE® Developer и IBM Rational Test Manager®.

Точная и доступная информация позволяет снизить риски проекта и принимать своевременные решения с целью их недопущения. В результате IBM Rational ProjectConsole повышает предсказуемость проектов и позволяет соблюдать показатели, заложенные в графике и бюджете проекта.

Упрощение доступа к информации по проекту. Rational ProjectConsole предоставляет в распоряжение команды единую точку доступа к точной информации по проекту. Вам больше не придется тратить время на ожидание отчетов о состоянии, отправку многочисленных запросов об обновлении, «копание» в различных приложениях, базах данных, каталогах и документах с целью получения требуемой информации. Вы больше не будете принимать решения на базе устаревшей информации. Благодаря ProjectConsole вы экономите время, уменьшаете число ошибок, связанных с ручным вводом информации, и получаете более полный контроль за проектами по разработке ПО.

Настраиваемые шаблоны. Rational ProjectConsole предлагает полный набор типовых шаблонов для создания отчетов, а также возможность создавать собственные шаблоны, что позволяет представлять информацию, нужную членам команды, в удобном формате. Каждый отчет автоматически извлекает информацию из платформы разработки Rational Suite и инструментов других компаний согласно заданному расписанию или по запросу. После этого данные публикуются на Web-сайте, которые сразу доступны всем членам команды. Упрощенный доступ к информации по проекту позволяет уменьшить время на поиск информации и уделять больше внимания самому проекту.

Возможности IBM Rational ProjectConsole:

- Обеспечивает доступ всех членов команды к полной информации по проекту через единый Web-сайт
- Собирает и автоматически публикует информацию, полученную из IBM® Rational Suite® и инструментов других поставщиков, по запросу или согласно расписанию
- Объективно измеряет качество и ход выполнения проекта.
- Анализирует данные и графически представляет результаты измерений.
- Предлагает индикаторы и средства для анализа трендов.
- Обеспечивает контроль доступа к отчетам и инструментальным панелям с помощью встроенных средств защиты

Измерение степени завершенности и качества проекта. Продукт Rational ProjectConsole позволяет автоматически получать количественную оценку текущего состояния проекта и анализировать тенденции. Измеряемые показатели автоматически формируются средствами Rational Suite и инструментами других поставщиков с последующим их сохранением в хранилище данных.

Результаты измерений, полученные из Rational RequisitePro, Rational ClearQuest и Rational ClearCase, могут быть графически отображены в Rational ProjectConsole.

Rational ProjectConsole позволяет с помощью гиперссылок связать все объекты и отображать результаты анализа в виде диаграмм, индикаторов и таблиц, публикуя их на автоматически создаваемом и обновляемом Web-сайте.

Объективное состояние проекта. Графическая панель позволяет всем членам команды быстро определять реальное состояние и качество проекта. Rational ProjectConsole позволяет анализировать подробную информацию по отдельным операциям, такие как моделирование, создание программного кода, тестирование. Кроме того, можно визуальнo сравнить плановые показатели с фактическими, проследить тенденции в хронологических данных и просмотреть междисциплинарные измерения, получив полное представление о состоянии проекта в целом. Все это позволяет команде разработчиков ПО:

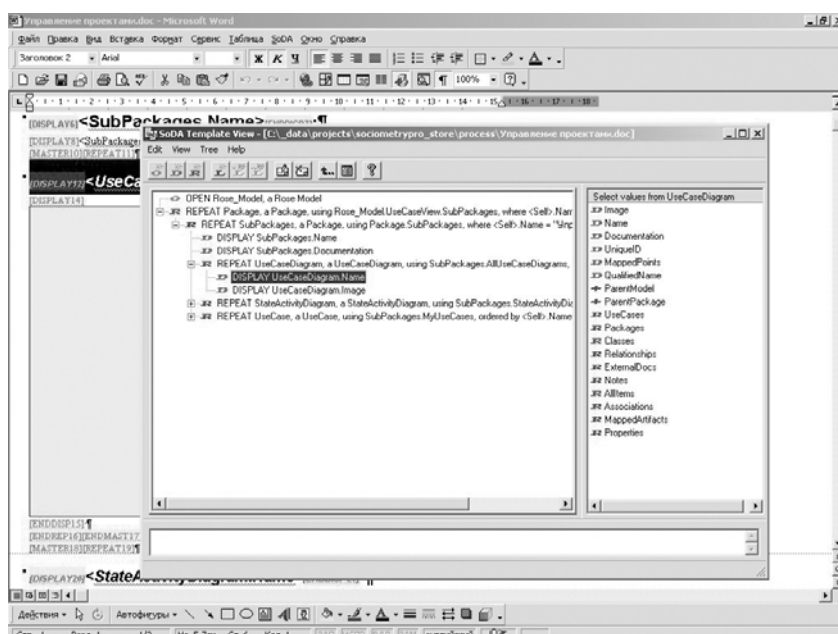
- принимать своевременные корректирующие меры;
- анализировать причины задержек;
- иметь реалистичные ожидания в отношении проекта;
- лучше прогнозировать этапные события проекта;
- обеспечивать объективное и точное измерение состояния и качества проекта

Rational ProjectConsole входит в состав платформы Rational Suite Team Unifying Platform.

IBM Rational SoDA

Формирование проектной документации обычно требует много времени и усилий. Нередко составление документов и отчетов в течение жизненного цикла проектов осуществляется случайным образом, а то и вовсе игнорируется из-за нехватки времени и ресурсов, которые требуются на их разработку и сопровождение.

Специализированное средство IBM Rational SoDA помогает успешно решить эти проблемы, позволяя автоматизировать создание и сопровождение необходимой проектной документации. SoDA значительно облегчает процесс документирования в проекте. Инструмент позволяет более согласованно и оперативно генерировать необходимые документы и отчеты, автоматически извлекая исходную информацию из репозитариев и артефактов проекта, созданных в других инструментальных средствах. Таким образом, SoDA — мощный инструмент для организации обмена жизненно важной информацией проекта между его непосредственными и внешними участниками (заказчиками, конечными пользователями и т. д.).



Работа в Rational SoDA

Использовать Rational SoDA крайне просто. Сначала создаются шаблоны документов, а затем на их основе генерируются сами документы. При этом участники проекта акцентируют внимание только на создании продукта и сопутствующих артефактов. Можно быть абсолютно уверенным, что сгенерированный документ будет всегда включать актуальную информацию, полученную из проектных репозитариев и артефактов. Rational SoDA изначально включает несколько десятков шаблонов.

Основу Rational SoDA составляет Microsoft Word. Любой внешний вид документа, который можно создать в Word, может быть представлен в виде шаблона SoDA. Таким образом, Rational SoDA поддерживает возможность стандартизации типов документов в рамках отдельного проекта или всей организации в целом. Эта стандартизация может обеспечить соответствие документов таким стандартам, как ISO, SEI CMM/CMMI и IEEE, повышая качество проектной документации и облегчая взаимодействие занятых в проекте сотрудников.

Шаблоны SoDA содержат информацию о форматировании документа, его структуре и стилях. В них фиксируется расположение источников информации, из которых извлекаются необходимые

данные. По желанию пользователя Rational SoDA может автоматически генерировать документы и отчеты в формате HTML. Это значительно упрощает публикацию документов в Интернет. Данный функционал особенно полезен для распределенных проектных команд.

Rational SoDA генерирует документы, извлекая информацию из следующих проектных репозитариев:

- репозитарий требований Rational RequisitePro;
- репозитарий тестирования Rational TestManager;
- базы данных запросов на изменения Rational ClearQuest;
- версионные объектные базы (VOB) Rational ClearCase;
- общий проектный репозитарий Rational Administrator.

Кроме того, Rational SoDA может извлекать данные из следующих отдельных артефактов проекта:

- модели Rational Rose и Rational Rose RealTime;
- планы Microsoft Project;
- документы Microsoft Word

Документация программного обеспечения для крупных проектов может составлять сотни и тысячи страниц. При этом документы должны быть постоянно актуальными, чтобы отражать текущее состояние дел в проекте. Rational SoDA позволяет обновлять лишь отдельные части документов, что значительно упрощает ведение процесса документирования.

Примеры внедрения проектов на базе технологий IBM Rational

Swedbank повышает эффективность создания ИТ-приложений, ориентированных на потребности клиентов и цели бизнеса, с помощью разработки на основе моделей

Заказчик	Swedbank (F_reningsSparbanken AB)
Задача	Организация единого процесса разработки ПО в рамках комплексной, многоплатформенной среды с целью устранения разрыва между потребностями бизнеса и проектами по разработке ПО, повышения эффективности разработки и ускорения выхода на рынок с новыми продуктами.
Технологии	IBM Rational Unified Process, Unified Modeling Language, IBM Rational Rose XDE Developer, IBM Rational Application Developer for WebSphere Software

Банк Swedbank – или F_reningsSparbanken AB – одна из крупнейших банковских групп Скандинавии - обслуживает 4,3 миллиона частных клиентов в Швеции и 4 миллиона в государствах Балтии. Для обслуживания клиентов Swedbank использует обширную сеть филиалов, крупнейшую в Швеции

систему управления банковскими счетами по телефону и крупнейшую в Швеции систему Интернет-банк. Основу стратегии банка составляет уникальная комбинация локального присутствия и использования глобальных ресурсов.

ИТ-подразделение банка, в штате которого работает более 800 разработчиков, помогает успешно реализовывать эту стратегию. Эта команда создавала качественное программное обеспечение, но процесс разработки не всегда был последовательным. Обнаружив возможности для совершенствования работы ИТ-подразделения, Исполнительный комитет банка принял решение о развертывании в рамках всего предприятия единого и унифицированного процесса разработки, выбрав в качестве основы методологию IBM Rational Unified Process (RUP).

Маргарета Баунге (Margareta Baunge), руководитель проекта банка Swedbank, вспоминает: «Мы использовали несколько разных процессов: бизнес-специалисты работали с одним процессом, а разработчики использовали другой. Мы решили, что должны придерживаться единой последовательной методологии в рамках всего банка и всех платформ – как унаследованных, так и разрабатываемых систем. Целью этого решения было повышение производительности и эффективности работ в рамках наших проектов».

Выбор в пользу RUP и преимущества разработки на базе моделей

Выбор Rational Unified Process – адаптируемой платформы разработки программного обеспечения, созданной на основе лучших методов организации работ – обеспечил последовательное и непротиворечивое управление процессами в рамках всего ИТ-подразделения, а также возможность совместной работы с консультантами, заказчиками и партнерами, которые также используют это решение. «Платформа RUP является стандартом, который используется многими нашими заказчиками и бизнес-партнерами по всему миру. Важно использование общей терминологии, и мы выбрали платформу RUP, с которой знакомы все», - отмечает Баунге.

После стандартизации процессов с использованием RUP, SwedBank начал переходить на подход к разработке на основе моделей, реализуя такие преимущества как многократное использование компонентов, упрощенное сопровождение и ускоренная разработка. Rational Rose XDE Developer помог улучшить коммуникацию внутри команды благодаря использованию вариантов использования (Use Cases), моделей анализа и моделей разработки. В частности, это позволило успешно завершить создание комплексной банковской, многоплатформенной ИТ-среды, которая включает Windows®, UNIX®, Java®, серверы IBM AS/400®, базы данных IBM DB2® и сервер приложений IBM WebSphere Application Server.

«С помощью IBM Rational Rose XDE Developer мы смогли ликвидировать разрыв между бизнесом и ИТ. Бизнес-специалисты используют этот продукт для моделирования бизнеса, они могут теперь обсуждать и анализировать модели», - говорит Баунге. «Используя IBM Rational Rose XDE Developer для автоматической генерации программного кода на основе моделей, команда Swedbank сэкономила значительный объем рабочего времени разработчиков, а также смогла быстрее адаптироваться к изменениям в требованиях и архитектуре. По оценке одного из наших разработчиков приложений, IBM Rational Rose XDE Developer сгенерировал примерно 40% программного кода для конкретного приложения. При компонентной разработке этот показатель возрастает до 50%. При таком подходе к разработке нам легче вносить изменения, поскольку требуется лишь внести изменения в модели и, затем, еще раз выполнить генерацию программного кода. И вы всегда знаете, что вся документация соответствует программному коду. Раньше, до того, как мы перешли на RUP и Rational Rose, отдельно существовали модель разработки и программный код. Изменения вносились в программный код, однако модель разработки не обновлялась. Теперь мы перешли на модельную разработку, в рамках которой модель и код всегда синхронизированы между собой, а документация всегда обновляется» - поясняет Баунге.

Разработка ПО в рамках полнофункциональной среды IDE

Используя сгенерированный программный код, разработчики Swedbank завершают разработку кода с использованием IBM Rational Application Developer for WebSphere Software в рамках полнофункциональной интегрированной среды разработки (IDE), предназначенной для визуального проектирования, конструирования, тестирования и развертывания Web-служб, порталов и приложений Java™ 2 Enterprise Edition (J2EE). Rational Application Developer стал еще одним средством, обеспечивающим быстрый отклик на потребности заказчиков в новых продуктах при одновременном снижении затрат на разработку ИТ-приложений.

«Продукт Rational Application Developer написан на языке Java на базе открытого исходного кода с использованием модульной архитектуры. Это означает, что мы можем создавать собственные инструменты для решения специфических проблем и интегрировать их в состав среды IDE. Другие поставщики также разрабатывают свои продукты в виде подключаемых модулей, что позволяет нам получать доступ ко всем программам из единой среды» - сообщает Баунге.

Программное обеспечение IBM Rational Application Developer for WebSphere Software упростило интеграцию компонентов и приложений на пяти различных платформах, используемых банком Swedbank. В сочетании с возможностями многократного использования компонентов и функций это позволило ускорить выход на рынок новых продуктов.

Результаты

Помимо улучшения коммуникации и коллективной работы в рамках команды, продукты IBM Rational Unified Process и IBM Rational Rose XDE Developer позволили разработчикам Swedbank – начиная с самого первого проекта – сэкономить время при одновременном повышении качества разрабатываемого программного обеспечения. «В ходе реализации пилотного проекта мы установили, что производительность труда разработчиков действительно повысилась примерно на 30%», - отмечает Баунге.

Кроме того, использование RUP и инструментов IBM для разработки ПО на базе моделей помогли Swedbank повысить эффективность процесса разработки, сформировать непротиворечивую терминологию и методы работы, а также ликвидировать разрыв между бизнес- и ИТ-подразделениями. Учитывая успех этих инициатив, Swedbank решил распространить пример на все свои текущие проекты – а их более двухсот, – а также на все будущие инициативы в области разработки ПО. В будущем Swedbank также планирует оценить возможность использования инструментов IBM Rational для управления изменениями и управления другими потоками операций по разработке ПО.

«Применение RUP, IBM Rational Rose XDE Developer и IBM Rational Application Developer for WebSphere Software позволило улучшить нашу способность создавать и предоставлять заказчикам оптимальные решения. Кроме того, мы усовершенствовали процесс разработки систем и компонентов, которые соответствуют требованиям наших клиентов, ускорив при этом выход на рынок новых продуктов. Без помощи RUP и данных инструментов этого было бы невозможно достичь», - подводит итог Баунге.

Внедрение системы управления конфигурациями и изменениями повышает эффективность сопровождения банковских систем

Заказчик	крупный российский банк
Бизнес-партнер	компания Аплана, группа компаний АйТи
Задача	создание эффективной системы управления конфигурациями и изменениями в ходе сопровождения банковских систем, интегрированной с уже используемыми в банке системами поддержки разработки.
Технологии	методологии и инструментальные средства IBM Rational (Rational ClearQuest, Rational ClearCase, Rational RequisitePro, Rational Test Manager, Rational Robot)

Бурное развитие российской банковской системы в последние годы сопровождалось значительными изменениями в законодательстве, изменениями в требованиях к банковской отчетности. Одновременно увеличивались масштабы и сложность проводимых финансовых операций; широкое распространение получили новые виды услуг, связанные с использованием возможностей сети Интернет.

В результате практически все банки, которые использовали информационные системы или отдельные модули собственной разработки, столкнулись с необходимостью динамического развития банковских систем и оперативного внесения изменений. В то же время эти изменения не должны были повлиять на качество и надежность ИС, а обновления должны были появляться практически одновременно с появлением новых требований.

С такими же проблемами столкнулись и в отделе информатизации одного из крупнейших российских банков, перед которым стояли задачи создания системы управления конфигурациями и изменениями программных систем, интегрированной с уже используемыми в банке системами поддержки разработки. Внедрение системы должно было обеспечить повышение качества разрабатываемого программного обеспечения и эффективности работы сотрудников подразделения, занимающегося разработкой и сопровождением банковских систем.

Принимая во внимание требования, связанные с необходимостью тесной интеграции инструментальных средств, выбор был сделан в пользу методологий и инструментальных средств IBM Rational.

Для выполнения поставленных целей необходимо было разработать и внедрить следующие технологии поддержки разработки:

- систему управления запросами на изменения;
- систему конфигурационного и версионного контроля и управления автоматической сборкой систем и отдельных модулей;
- систему тестирования и контроля качества разрабатываемого ПО.

Решение

Для обеспечения эффективной совместной работы членов проектной команды в банке были внедрены системы, предназначенные для эффективного управления запросами на изменения, управления версиями и конфигурациями, управления требованиями, а также средства автоматизированного тестирования IBM Rational.

С помощью средства конфигурационного управления Rational ClearCase был организован эффективный конфигурационный и версионный контроль, автоматизирован процесс сборки версий и обеспечено структурированное хранение проектных материалов, таких как готовые версии, отдельные модули систем, объекты и библиотеки, тестовые спецификации и скрипты, проектная документация. Такой подход к организации проектных материалов позволил минимизировать влияние человеческого фактора на этапе сборки версий, в частности, снизить зависимость успеха проекта от его конкретных участников.

На базе средства управления изменениями Rational ClearQuest был организован эффективный процесс отслеживания за исправлением ошибок (дефектов) в разрабатываемом программном обеспечении. Все запросы на изменения регистрируются в системе, при этом в этом процессе могут участвовать не только члены проектной команды, но и конечные пользователи, которые могут направлять описания обнаруженных ошибок и предложения по улучшению непосредственно в систему управления запросами на изменения, где указанная информация будет исследована соответствующими специалистами банка. Устранение обнаруженных ошибок поручается конкретным исполнителям с возможностью дальнейшего отслеживания статуса исполнения работы. При этом ClearQuest позволяет планировать изменения, увязывая выполнение отдельных задач с выпуском конкретных версий.

Для повышения эффективности процедур тестирования новых версий в банке были внедрены средства автоматизированного тестирования Rational Test Manager и Rational Robot. С их помощью были созданы тестовые спецификации и сценарии для проведения функционального тестирования.

Внедрение методологии RUP и разработка поддерживающих ее регламентирующих документов позволили эффективно организовать процессы оперативного и стратегического планирования работ, обеспечить более четкое определение ролей и обязанностей участников проектов, а также управление проектными рисками.

Особенности проекта

Среди особенностей проекта можно выделить следующие:

- интеграция внедренной системы конфигурационного управления и управления запросами на изменения на базе ClearCase/ClearQuest с системой поддержки пользователей Microsoft Exchange Helpdesk и корпоративным интернет-порталом с целью обеспечения оперативного информационного обмена;
- интеграция средств управления запросами на изменения и версионного контроля со средствами планирования проектов (Microsoft Project), что позволило детализировать планируемые работы в виде отдельных задач, назначать для них ответственных исполнителей и контролировать их работу;

- интеграция средств конфигурационного управления и управления запросами на изменения (ClearCase/ClearQuest) на базе технологии Unified Change Management (UCM — универсальное управление изменениями)

Результаты

Внедрение системы управления конфигурациями и изменениями (задачами, дефектами, запросами на улучшения) на базе методологий и технологий IBM Rational позволило обеспечить реализацию единой цепочки действий, связанных с внесением изменений в систему, включая регистрацию запроса на изменение (в том числе описания дефектов), планирование и мониторинг выполнения этой задачи с привязкой к конкретной версии систем, отражение изменения в материалах тестирования и документации. В результате процесс планирования и выпуска новых версий стал более прогнозируемым и управляемым, сократились сроки выпуска обновлений, заметно повысилось их качество.

Внедренная система обеспечила руководителей проектов удобными инструментами планирования работы проектной команды, мониторинга выполнения задач по конкретным исполнителям или групп

пам, а также средствами подготовки отчетов по текущему состоянию работ и по загрузке проектных ресурсов. При этом данные, получаемые при помощи средств учета ресурсов, которые потрачены на выполнение отдельных задач и на проект в целом, могут быть использованы для оценки трудозатрат в новых проектах.

Интеграция процессов конфигурационного управления и управления запросами на изменения на базе принципов UCM позволила поднять на качественно новый уровень процессы планирования и управления работами по выпуску новых версий. При этом внесение изменений в систему возможно, только если конкретному исполнителю поставлена соответствующая задача.

Использование средств автоматизированного тестирования IBM Rational на этапе разработки и ввода в эксплуатацию новых версий разрабатываемых в банке информационных систем позволило существенно снизить трудозатраты и сроки проведения тестирования, а также сократить до минимума влияние человеческого фактора на этапе тестирования. В результате удалось снизить затраты на тестирование и значительно повысить качество и надежность разрабатываемых ИС.

В ближайшей перспективе развития системы управления проектами банка — внедрение и настр-ройка средств автоматической сборки версий, а также дальнейшее совершенствование системы управления требованиями.

Внедрение Unified Change Management в компании Protek поднимает управления проектами на качественно новый уровень

Заказчик	Protek
Бизнес-партнер	компания Аплана, группа компаний АйТи (поставка инструментальных средств)
Задача	интеграция процесса управления изменениями с процессом версионного и конфигурационного управления, максимальная автоматизация всех основных процессов разработки, повышение прозрачности и управляемости проектов
Технологии	Unified Change Management

Компания Protek осуществляет разработку, продажу и техническое сопровождение программных продуктов, предназначенных для конвергентных операторов связи и провайдеров услуг связи нового поколения. Имея опыт свыше 70 успешных внедрений в 35 странах мира, Protek представляет решения в области биллинга и обслуживания клиентов, учета сетевых ресурсов, управления сетями, а также сбора и обработки информации от сетевых устройств (mediation).

Московский центр разработки Protek является частью глобальной команды разработчиков компании, имеющей штаб-квартиру в Великобритании и офисы в СНГ, Европе, Америке, Южной Азии, Южной Африке и на Ближнем Востоке.

С 1998 г. проекты в московском центре разработки Protek ведутся в соответствии с методологией ведения проектов IBM Rational Unified Process. В частности, в ходе разработки и сопровождения

программных систем активно используются средства конфигурационного управления, управления изменениями, управления требованиями, а также средства тестирования от IBM Rational. В то же время из-за роста объемов и сложности разрабатываемых в Protek программных систем компания столкнулась с необходимостью дальнейшего повышения эффективности ведения проектов на основе максимальной автоматизации основных процессов. Первоочередными задачами, которые предстояло решить московскому офису Protek, стало следующее:

- обеспечение всех участников проекта единым инструментом эффективного информационного обмена в рамках проекта;
- интеграция процесса управления изменениями с процессом управления версиями и конфигурациями, автоматизация процесса сбора запросов на изменения;
- предоставление руководителю проекта эффективного инструмента для управления проектом, контроля за ним и формирования необходимой отчетности.

В качестве средства решения указанных задач была выбрана методология Unified Change Management (UCM), созданная IBM Rational на базе анализа наиболее удачного опыта в этой области — от постановки требований до выпуска релиза и его дальнейшей эксплуатации.

Решение

Внедрение механизма UCM, который представляет собой функциональную надстройку над Rational ClearCase, должно было обеспечить решение поставленных задач за счет объединения процессов конфигурационного управления и управления запросами на изменения.

Внедрение принципов UCM затрагивает все аспекты изменений в ходе проекта. Характерной чертой UCM является возможность управления изменениями на уровне задач вместо отслеживания изменений в отдельных файлах или директориях, как это принято в стандартном варианте ClearCase.

При таком подходе каждая задача ассоциируется с набором проделанных или планируемых изменений. Таким образом, автоматизируя многие рутинные операции, UCM позволяет сфокусировать внимание коллектива разработчиков на задачах более высокого уровня, на их непосредственной работе. Не менее важно, что при включении в команду проекта новых разработчиков, UCM позволяет им сразу активно включиться с работу, выполняя конкретные четко поставленные задачи.

Результаты

Внедрение UCM в проект “Nexoss” позволило наладить эффективный версионный контроль и управление конфигурациями, а также поднять на качественно новый уровень управление проектом в целом.

Использование принципов универсального управления изменениями позволило организовать независимое рабочее пространство для каждого разработчика, команды в целом и всего проекта. Результатом стало значительное упрощение сборки новых версий программных систем и автоматизации этой процедуры. Введение высокого уровня организации задач позволило снизить число объектов версионного контроля с нескольких тысяч файлов до нескольких десятков активностей (activities) для нескольких направлений проекта (baselines).

С точки зрения менеджеров проекта, выполняемые работы стали более наглядными и прозрачными, появилась возможность получить высокий уровень детализации при оценке состояния проекта.

Оперативное планирование проекта теперь осуществляется «сверху вниз», поэтапно, с возможностью автоматизации подготовки отчетов. Более гибким и оперативным стало перераспределение ресурсов. Возможности интеграции UCM с Microsoft Project и возможности создания актуального и полного перечня задач (To_Do_List) для каждого члена команды избавило коллектив от дублирования усилий и выполнения рутинных операций.

В целом, внедрение технологии UCM позволило повысить качество разработки программных систем. Таким образом, появились возможности для эффективного мониторинга актуального состояния проекта и своевременного выявления «узких мест» производительности. Автоматическая рассылка уведомлений дала возможность оперативно доводить до каждого участника проекта необходимую и достаточную для внесения изменений информацию, что отражается в лучшую сторону на сроках проекта. Кроме того, высокий уровень автоматизации UCM снижает степень влияния человеческого фактора, уменьшает количество ошибок и дисциплинирует команду.

В ближайших планах Protek – запуск пилотного проекта для мультисайтной реализации UCM в центрах разработки, расположенных в Москве, Англии и Норвегии.

Business Objects повышает эффективность разработок с помощью IBM Rational ClearQuest

Заказчик	Business Objects
Задача	Автоматизация процесса управления изменениями в распределенном режиме в целях повышения качества создания программных продуктов, производительности и эффективности команды разработки, которая непрерывно расширяется численно и географически.
Технологии	IBM Rational ClearQuest

Компания Business Objects, продуктами которой пользуются порядка 30 000 заказчиков в 80 странах мира и офисы в Сан-Хосе, Калифорния, и Париже, Франция, является одним из лидеров на рынке систем бизнес-анализа, помогая крупнейшим компаниям управлять своим бизнесом.

После того, как команда разработчиков Business Objects заметила снижение эффективности и производительности процесса разработки ПО, эта проблема была признана угрозой эффективности на уровне всей компании. Используемый инструмент для отслеживания и управления дефектами в ходе разработки ПО не располагал средствами поддержки командной работы, и специалисты полагались на ручную проверку при контроле качества.

Поиск альтернатив

Джеймс Андерсон (James Anderson), менеджер по контролю качества компании Business Objects, вспоминает: «Мы располагали процессом, с которым все были знакомы, но который не был зафиксирован “на бумаге” Проблема заключалась в том, что используемый нами продукт для контроля дефектов не поддерживал переходов из одного состояния в другое и способов формализации процесса. Кроме того, команда разработчиков тогда насчитывала 250 человек в разных офисах от Ванкувера до Великобритании, а структура этого инструмента не была масштабируемой. В результате, ежедневные простои системы продолжительностью от 30 до 45 минут приводили к реальному снижению производительности и замедлению разработки”.

В ходе поиска нового средства для контроля дефектов на первый план вышли такие характеристики, как масштабируемость и возможность автоматизации процессов. После оценки нескольких продуктов, компания Business Objects выбрала решение IBM Rational ClearQuest – гибкую систему для контроля изменений и дефектов с высоким уровнем масштабируемости, которая управляет всеми типами запросов на изменение на протяжении жизненного цикла разработки. Дополнительным аргументом в пользу ClearQuest стал успешный опыт использования инструментов тестирования IBM Rational. Кроме того, IBM Rational ClearQuest тесно интегрирован с собственными продуктами Business Objects – Crystal Enterprise™ и Crystal Reports®, – что позволяло пользователям Rational ClearQuest генерировать отчеты в любое время и в любом месте сразу же после установки..

Компания Business Objects смогла быстро развернуть решение IBM Rational ClearQuest, затратив на это на треть меньше времени, чем предполагалось. Ускоренное внедрение стало возможным, в частности, благодаря гибкости Rational ClearQuest и его матрице переходов между состояниями, которая позволила команде автоматизировать и формализовать свой процесс. «Поддержка переходов между состояниями была тем фундаментальным блоком для формализации процесса, который у нас отсутствовал. Благодаря Rational ClearQuest, мы гораздо быстрее выпустили первый продукт, добившись при этом и экономии расходов» - говорит Андерсон. Помощь консультантов IBM Rational, которые помогли формализовать используемые процессы, и услуги по обучению также способствовали экономии времени.

Программные блоки и другие возможности

Команда Business Objects использовала несколько ключевых функций IBM Rational ClearQuest для адаптации своих схем и реализации своего процесса, включая такие возможности как:

- создание настраиваемых формуляров и упорядочивание информации с учетом вкладок;
- разработка формуляров для Windows® и Web-клиентов;

- реализация «готовых» средств защиты и контроля доступа для групп пользователей;
- определение требуемых полей и связывание их с переходами между состояниями.

Команда адаптирует Rational ClearQuest к новым потребностям, используя для этого специальные программные блоки (hook) – программный код, исполняемый в определенное время или при определенных обстоятельствах с целью инициирования связанной операции. Эти программные блоки помогают отдельным специалистам понять, что и каким образом им нужно делать в контексте используемой модели разработки. Они также используются для ведения подробных контрольных журналов и автоматизации многих других шагов.

В частности, программные блоки сыграли ключевую роль в эффективной реализации механизма ветвления, используемого командами разработчиков. После того, как кто-нибудь вводит дефект, программный блок автоматически создает начальную ветвь.

Поддержка многоплатформенной, распределенной разработки

Помимо инструментов разработки под управлением Windows, многие разработчики и специалисты по тестированию Business Objects используют AIX, Linux, Solaris и другие платформы. Кроме того, команда разработчиков Business Objects выросла и теперь включает почти 500 специалистов, работающих по всему миру, включая офисы в США, Канаде, Франции, Китае, Индии и Великобритании. В этих условиях ключевое значение приобретает доступ к системе управления дефектами компании через Web. Web-клиент Rational ClearQuest позволяет команде разработчиков Business Objects отслеживать дефекты и управлять ими в рамках многочисленных платформ, практически из любой точки компании. Используя Web-браузер, члены команды представляют информацию о дефектах, выполняют запросы и получают доступ к детальным схемам и отчетам.

Еще одним важным преимуществом Web-клиента Rational ClearQuest является удобство администрирования, поскольку он позволяет избежать установки и обновления клиентского программного обеспечения. В случае Business Objects это означало экономию на инфраструктуре и ресурсах, необходимых для развертывания 500 клиентов по всему миру.

Управление потоком операций и задачами контроля

С помощью Rational ClearQuest компания Business Objects отслеживает не только дефекты, но и операции. Прямая интеграция Crystal Enterprise в состав Web-интерфейса Rational ClearQuest позволила значительно повысить производительность труда разработчиков и усовершенствовать потоки операций. Пользователи теперь могут просматривать отчеты и вызывать записи редактирования в Rational ClearQuest в одном и том же окне, а параметрические отчеты позволяют увидеть собственные списки задач.

Все это помогает разработчикам и руководителям проектов управлять потоками операций, а также легко присваивать и отслеживать задачи в рамках всего проекта. Руководители проектов используют это решение для отслеживания задач разработки, контроля качества и тестирования ПО, локализации и документирования и имеют общую картину издержек проекта, а также хода реализации проекта.

Бизнес-преимущества

Программное обеспечение Rational ClearQuest помогло компании Business Objects эффективно автоматизировать и реализовать процесс разработки в условиях его непрерывной эволюции, а также внесло большой вклад в повышение производительности, эффективности и качества создания продуктов.

Андерсон отмечает, что Rational ClearQuest вместе с усовершенствованиями, внесенными в процесс тестирования, помог компании Business Objects повысить общее качество разработки. «Я считаю, что последняя версия продуктов нашей линейки Crystal, предназначенных для создания отчетов, стала самой качественной и надежной. Количество вопросов и проблем, с которыми мы столкнулись после выхода версии, было наименьшим за всю нашу историю. Кроме того, эта версия вошла в число наиболее продаваемых наших продуктов».

Интеграция Rational ClearQuest в процесс разработки стала одним из ключевых факторов успеха в условиях, когда компания начала реализацию более сложных проектов с использованием географически распределенных команд. При этом сохраняются возможности адаптации процесса, который может эволюционировать в соответствии с изменениями требований. Теперь команда имеет гарантии, что ни один шаг не будет пропущен, и ни один отдел или руководитель проекта не будет «оставлен за бортом» при принятии важных решений, что обеспечивает своевременное выполнение правильных операций правильными людьми.

VIVEO ToolObject создает технологически независимую бизнес-модель для банковской индустрии на основе UML и средств моделирования IBM Rational

Заказчик	VIVEO ToolObject
Задача	Создание технологически независимой компонентной бизнес-модели для банковской индустрии (ECM, Enterprise Component Model) в целях повышения конкурентоспособности продуктов VIVEO ToolObject
Технологии	RUP, UML, инструменты моделирования IBM Rational Rose XDE Developer

Компания VIVEO ToolObject, входящая в корпорацию VIVEO Group, зарекомендовала себя как поставщик стратегических ИТ-решений для банковской и финансовой сферы. В определенный момент команда VIVEO ToolObject пришла к выводу, что информационные системы в банковском секторе быстро устаревают в условиях быстрого изменения рыночного спроса и технологий. И происходит это, в первую очередь, из-за того, что многие ИТ-проекты начинаются с определения технической платформы, еще до рассмотрения бизнес-архитектуры.

Объединив свою экспертизу в банковской индустрии с накопленным опытом в технологической сфере, компания VIVEO ToolObject создала принципиально новое решение, основанное на адаптируемой модульной бизнес-модели банковской организации и использовании итеративного подхода к разработке. Отделив бизнес-стратегию от базовой технологической платформы, VIVEO ToolObject создала комплексную бизнес-модель банковского бизнеса и финансов, дополнив ее собственной методологией ECM finance.

Стандартизованное, адаптируемое решение

Разработанный на базе управляемой моделями архитектуры (Model-Driven Architecture), продукт ECM Finance представляет собой компонентную модель предприятия (ECM) для финансового сектора, которая основана на проверенных схемах работы в банковской сфере и включает технологически независимую оболочку.

Изначально ECM Finance создавался как адаптируемая модель, легко настраиваемая в соответствии со специфическими бизнес-потребностями и требованиями технологической платформы конкретного заказчика. В связи с этим компания VIVEO ToolObject ориентировалась на использование стандартизованного в рамках отрасли языка моделирования и выбрала для создания продукта Unified Modeling Language (UML), а также поддерживающие его инструменты моделирования IBM Rational Rose XDE Developer, а также RUP (Rational Unified Process) в качестве настраиваемой, адаптируемой платформы для разработки ПО.

Адаптация продукта начинается с анализа всех бизнес-потребностей и создания модели ECM с использованием инструментов Rational и языка UML. Затем на базе этих моделей совместно с заказчиком создаются практические решения. При этом совместная разработка вариантов использования (Use Cases) гарантирует, что бизнес-пользователи смогут понять спецификации и принять участие в процессе разработки эффективного решения.

В ходе работ обеспечивается полный оперативный контроль за процессом создания решения, который гарантирует соответствие разворачиваемой системы ожиданиям заказчика. Решение моделируется с помощью средств RUP, включая процессы бизнес-моделирования, определения требований, анализа и разработки. На базе управляемой моделями архитектуры (MDA, Model Driven Architecture) создается модель реализации, с помощью которой генерируется код и выполняется разработка конечных приложений. Оперативный контроль и следование подходу RUP позволяют полностью отразить в конечном решении потребности пользователей.

Сокращение этапа бизнес анализа и моделирования на 60%

ECM Finance и подход VIVEO ToolObject помогают заказчикам сократить длительность этапа анализа и моделирования на 60 %, поскольку они могут использовать существующую модель, и им не приходится

приходится начинать с нуля. Такой подход позволяет быстрее начинать новые проекты и добавлять новые возможности. Например, во Франции многие финансовые организации используют концепцию городского планирования в качестве аналога ИТ-системам. Данный подход, получивший название «урбанизационного», подразумевает модульное разбиение информации и процессов на «городские кварталы», а связывающие их дороги представляют каналы коммуникации. Команда VIVEO ToolObject использовала этот подход и включила его в состав ECM finance с помощью языка UML. Эта методология определяет бизнес-карту, которая упорядочивает бизнес-знания в рамках систем и кварталов, позволяя таким образом распараллеливать выполнение бизнес-задач.

В результате создается целевая бизнес-модель предприятия, которая точно описывает различные бизнес-процессы и позволяет их эффективно использовать в рамках подхода MDA при создании пользовательских приложений. Продукт ECM Finance содержит описание полных и детальных бизнес-процессов, которые определяют службы и интерфейсы, необходимые для построения пользовательских приложений. В результате, эти процессы можно интегрировать во многие архитектуры реализации, включая сервис-ориентированную архитектуру (SOA, Services-Oriented Architecture).

Оптимизация процесса разработки

В рамках типового клиентского проекта решение ECM finance применяется в составе многоэтапного процесса. Прежде всего, модель ECM Platform Independent Model (PIM) адаптируется и индивидуализируется в соответствии со специфическими бизнес-требованиями клиента. После этого PIM адаптируется с учетом технологических аспектов платформы заказчика, и создается модель для конкретной платформы (PSM, Platform Specific Model) – с использованием инструментов Rose Scripts или технических шаблонов для преобразования из PIM в PSM. На последнем этапе с помощью Rational Rose и на основе PSM генерируется программный код Java™ или C++.

На основе модели проектирования определяются правила автоматизации для создания PSM, которая не будет зависеть от технологической платформы, например, J2EE или .NET. После этого настраиваются скрипты для генерации модели реализации, и на ее основе генерируется программный код. Благодаря использованию преобразований MDA модель ECM служит в качестве единого источника для создания всех новых моделей, включая PIM и PSM, на основе бизнес-требований и технических ограничений заказчика.

Помимо инструментов моделирования IBM Rational, успеху продукта ECM Finance способствовало также и применение методологии IBM Rational Unified Process. Использование итерационного подхода к разработке привело к значительной экономии рабочего времени и мгновенной обратной связи в конце каждой итерации. Кроме того, использование оболочки RUP позволило получать преимущества также и на этапе адаптации продукта к потребностям заказчиков.

Rational XDE и поддержка архитектуры MDA

Инструментарий Model Driven Architecture для IBM Rational XDE представляет собой набор инструментов и документации, который обеспечивает расширенную поддержку VIVEO ToolObject в ходе применения MDA при разработке приложений с использованием моделей IBM Rational XDE и UML. Этот инструментарий основан на двух очень мощных механизмах автоматизации – образцах и шаблонах программного кода, которые существенно облегчают создание и развертывание средств автоматизации MDA.

IBM Rational XDE добавил новое измерение в предложение VIVEO ToolObject, предоставив заказчикам интегрированное решение для анализа спецификаций вплоть до программного кода с помощью IBM Rational Application Developer for WebSphere® Software и платформы Eclipse. Открыв решение для существующих технологических оболочек, поддерживаемых в рамках Rational XDE, VIVEO ToolObject создала дополнительные аргументы в пользу приобретения своего решения. Полная поддержка MDA в рамках Rational XDE также является одним из ключевых преимуществ, поскольку позволяет отделить бизнес-аспекты от технических вопросов и делает решения VIVEO ToolObject более понятными для клиентов.

Гибкость, стабильность и возможность многократного использования

VIVEO ToolObject предлагает заказчикам стабильное, не зависящее от конкретной технологии решение, которое облегчает многократное использование новых и унаследованных компонентов, а также увеличивает степень гибкости всей системы. С помощью инструментов Rational, UML и RUP заказчики получают возможность создать промежуточный уровень между существующими системами

и новыми технологиями, дополняя унаследованные системы новыми возможностями. Интегрированная поддержка архитектуры MDA и платформ разработки J2EE и .Net, а также языков Java и C++, позволяет заказчикам использовать широкий диапазон продуктов, которые помогают воплотить создаваемые модели в реальные бизнес-решения.

ECM Finance используется рядом ведущих финансовых организаций, которые дают положительные отзывы об этом продукте. Успех ECM Finance внес значительный вклад в рост оборотов компании VIVEO ToolObject

Эффективная организация процессов тестирования на базе методологии RUP и средств тестирования IBM Rational

Заказчик	компания Аплана, группа компаний АйТи
Задача	Эффективная организация работы Центра Тестирования в проектной модели
Технологии	RUP, средства тестирования IBM Rational Test Manager, IBM Rational Robot, IBM Rational Clear Quest и др.

Центр Тестирования Аплана отвечает за обеспечение процессов тестирования во всех проектах по разработке заказного программного обеспечения, которые ведутся в компании с 1999 года. На сегодняшний день Центром выполнено более ста проектов различной сложности – от проверки работоспособности небольших приложений до тестирования проектов федерального уровня. Помимо обеспечения внутренних процессов разработки, Центр предоставляет на коммерческой основе услуги функционального и нагрузочного тестирования, а также услуги консалтинга в области организации процессов тестирования.

С ростом числа и сложности выполняемых проектов без существенного увеличения штата специалистов, перед Центром Тестирования встала задача создания регламентированной системы тестирования, которая должна была обеспечить рационализацию использования ресурсов и повышение управляемости процедур тестирования в проектной организации. Еще одной задачей стала подготовка компании к сертификации процессов разработки программных систем на соответствие требованиям ISO 9001:2000.

Решение

В качестве основы для формализации процедур тестирования в компании Аплана была использована методология RUP (Rational Unified Process), а также стандарты IEEE Std 829-1998 «Standard for Software Test Documentation» и Key Practices of the Capability Maturity Model SM, Version 1.1 /CMU/SEI-93-TR-025 ESC-TR-93-178.

Организационные процедуры и методики тестирования RUP были адаптированы к специфике проектной организации компании Аплана. На основе рекомендаций RUP были разработаны внутренние стандарты и описания процедур тестирования, а также соответствующие ролевые инструкции и шаблоны основных документов. В соответствии с разработанным регламентом, в ходе проектов разрабатываются такие обязательные документы как план тестирования (Test Plan) и итоговый отчет тестирования (Test Summary Report), а также, по необходимости, дополнительные документы - тестовые спецификации (Test Case Specifications), описание тестовых процедур (Test-Procedure Specifications), сценарии тестирования (Test Scripts), статус-отчет тестирования (Test Incident Report), программа и методика испытаний (Acceptance Test).

Особенностью адаптации методологии RUP стала возможность выбора для каждого проекта того инструментария тестирования, который наилучшим образом соответствует задачам проекта. В качестве базового инструментария для обеспечения процессов тестирования в компании используются средства IBM Rational. Так, для управления процессом тестирования используется Rational Test Manager, а для проведения функционального и нагрузочного тестирования – Rational Robot. Для документирования и отслеживания сообщений об ошибках, выявленных в ходе тестирования, используются средства IBM Rational ClearQuest.

Результаты

Внедрение адаптированной методологии RUP и использование современного инструментария позволяет Центру Тестирования компании Аплана эффективно и качественно проводить все основные виды тестирования программных систем: модульное, компонентное, функциональное системное,

регрессионное, нагрузочное, стрессовое. Следование регламентированным внутренним стандартам и использование количественных оценок в ходе проектов позволило сделать процедуры тестирования в компании Аплана более управляемыми и эффективными.

Разработанные регламентирующие документы и процедуры, обязательные для исполнения во всех проектах, стали важной составляющей Системы Качества компании Аплана, успешно сертифицированной на соответствие требованиям стандарта ISO900:2000.

Описание проектов иллюстрирует использование одним из заказчиков технологий и сервиса корпорации IBM и/или бизнес-партнеров IBM. Описанные результаты и выгоды складываются из многих факторов. Компания IBM не гарантирует сопоставимые результаты. Вся информация о проектах, предоставлена описываемым заказчиком и/или бизнес-партнером. Компания IBM не удостоверяет ее точность.

Полезные источники информации и сервисы

Услуги IBM Rational

Широкий спектр услуг и ресурсов IBM Rational, дополняют инструменты и практические рекомендации IBM Rational, помогая повысить квалификацию команды, сократить сроки ввода в эксплуатацию нового решения, а также получить максимальную отдачу от инвестиций в инструменты IBM Rational.

- Программы обучения работе с программным обеспечением IBM Rational включают в себя курсы, охватывающие все возможности разработки программного обеспечения. Сочетание обучения с инструктором и Web-обучения дает богатый набор возможностей, из которых всегда можно выбрать те, которые соответствуют потребностям вашей организации
- Всемирный штат профессиональных сервисных консультантов IBM Rational с многолетним опытом успешной работы поможет вам заложить прочный фундамент для успеха проектов по разработке программного обеспечения. Вы можете воспользоваться профессиональными услугами Rational в крупномасштабных консалтинговых проектах, настроенных на соответствие уникальным потребностям вашей организации, или пригласить местную группу специалистов для краткосрочного содействия в решении конкретной проблемы. Имеется также несколько типов «пакетов» услуг, включая предложения по объективной оценке ваших возможностей в разработке ПО, внедрению нового инструмента или методологии, содействию в получении конкретных результатов путем проведения специализированного семинара.
- IBM Rational Customer Support гарантирует, что ваш проект будет двигаться вперед, предоставляя вам помощь тогда, когда она нужна. Программы сопровождения позволяют эффективно разрешать проблемы благодаря он-лайн доступу к десяткам тысяч технических заметок, патчей и обновлений, а также прямому доступу к талантливой группе инженеров поддержки. IBM Software Maintenance – это стандартное предложение по обслуживанию программного обеспечения, которое в зависимости от конкретных требований вашего бизнеса можно дополнить пакетом услуг Technical Support Account Services.
- IBM DeveloperWorks и другие технические ресурсы для сообщества профессиональных разработчиков, использующих или оценивающих инструменты и практические рекомендации IBM Rational

Полезные ссылки в Интернет

Официальный сайт IBM Rational

<http://www.ibm.com/software/rational/>

Сайт, посвященный программному обеспечению IBM Rational, которое помогает организациям автоматизировать процесс разработки программного обеспечения. Продукты, и услуги Rational создают дополнительные возможности в рамках платформы IBM Software Development Platform, предназначенной для групп, которые создают бизнес-приложения, встроенные системы и программные продукты. Это полное, модульное решение позволяет использовать бизнес-ориентированный подход к разработке, основанный на открытых стандартах, включая структуру Eclipse с открытым исходным кодом. Результатом является значительный рост производительности бизнеса.

IBM DeveloperWorks

<http://www.ibm.com/developerworks/rational/>

Этот ресурс дает разработчикам возможность проходить обучение он-лайн и обмениваться опытом с другими разработчиками во всем мире.

На IBM Rational Developer Works публикуются вспомогательные материалы, такие как тренинги, примеры практического использования инструментов и методологии, обучающие артефакты и другие наработки. На IBM Rational Developer Works собираются и хранятся полезные статьи и публикации.

Кроме того, предоставляются мощные сервисы по участию в дискуссиях в виде онлайн-форумов, а также развитые механизмы поиска.

Электронный журнал RationalEdge

<http://www.ibm.com/developerworks/rational/rationaledge/>

Электронный ежемесячный журнал, посвященный следующим вопросам:

- как программное обеспечение IBM Rational помогает организациям автоматизировать и интегрировать основной бизнес-процесс разработки программного обеспечения;
- возможности IBM Software Development Platform
- описание возможностей разработки и развертывания программных активов в бизнес-приложениях, встроенных системах и программных продуктах;
- опыт применения бизнес-ориентированного подхода к разработке, основанный на открытых стандартах, включая структуру Eclipse с открытым исходным кодом и многое другое.

Интернет-портал средств Eclipse

<http://www.eclipse.org/>

Публикации, статьи, обучающие материалы, посвященные платформе Eclipse

Рекомендуемые книги

1. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, Гради Буч
Издательство: Бинум, ноябрь 2001, 560 стр.; обложка: мягкая
2. UML. Основы (2-е издание), Фаулер, Скотт
Издательство: Символ-Плюс, ноябрь 2001, 192 стр.; обложка: мягкая
3. UML. Руководство пользователя, Грейди Буч, Джеймс Рамбо, Айвар Джекобсон
Издательство: ДМК, 2001 г., 432 стр.; обложка: мягкая
4. UML. Проектирование систем реального времени, параллельных и распределенных приложений, Хассан Гома
Издательство: ДМК, 2002 г., 704 стр.; обложка: мягкая
5. UML. Специальный справочник, Джеймс Рамбо, Айвар Якобсон, Грэдди Буч
Издательство: Питер, 2002 г., 656 стр.; обложка: мягкая
6. Базы данных и UML. Проектирование, Мюллер
Издательство: Лори, 2002 г., 432 стр.; обложка: мягкая
7. Визуальное моделирование с помощью Rational Rose 2002 и UML, Терри Кватрани
Издательство: Вильямс, 2003 г., 192 стр.; обложка: мягкая
8. Применение UML и шаблонов проектирования (2-е издание), Крэг Ларман
Издательство: Вильямс, 2002 г., 624 стр.; обложка: мягкая
9. Применение объектного моделирования с использованием UML и анализ прецедентов, Розенберг, Скотт
Издательство: ДМК, 2002 г., 160 стр.; обложка: мягкая
10. Разработка Web-приложений с использованием UML, Коналлен
Издательство: Вильямс, 2001 г., 288 стр.; обложка: мягкая
11. Принципы работы с требованиями к программному обеспечению. Унифицированный подход, Леффингуэлл, Уидриг
Издательство: Вильямс, 2002 г., 448 стр.; обложка: мягкая
12. Приемы объектно-ориентированного проектирования. Паттерны проектирования, Гамма, Хелм, Джонсон, Влиссидес
Издательство: Питер, 2001 г., 368 стр.; обложка: мягкая
13. Управление проектами по созданию программного обеспечения, Ройс
Издательство: Лори, 2002 г., 424 стр.; обложка: мягкая
14. Управление конфигурацией программных средств. Практическое руководство по Rational ClearCase, Уайт
Издательство: ДМК, 2002 г., 272 стр.; обложка: мягкая
15. Введение в Rational Unified Process, Крачтен
Издательство: Вильямс, 2002 г., 240 стр.; обложка: мягкая
16. UML и Rational Rose, Боггс, Боггс
Издательство: Вильямс, 2001 г., 582 стр.; обложка: мягкая



IBM Восточная Европа/Азия

123370, Россия, Москва,
Краснопресненская наб., 18
Тел.: +7 095 775-8800
Факс: +7 095 258-6468
www.ibm.com/ru

Более подробную информацию о компании IBM, ее продуктах и услугах можно найти на сайте: ibm.com/ru

Логотипы IBM, DB2, MQSeries, VisualAge, SecureWay и WebSphere являются зарегистрированными торговыми марками International Business Machines Corporation в США и/или других странах.

Lotus и Lotus Notes, Sametime и QuickPlace являются торговыми марками Lotus Development Corporation в США и/или других странах.

Tivoli является торговой маркой Tivoli Systems, Inc. в США и/или других странах.

Java и все торговые марки и логотипы, использующие слово Java и Solaris являются торговыми марками Sun Microsystems, Inc. в США и/или других странах.

Microsoft, Windows, Windows NT и логотип Windows являются торговыми марками Microsoft Corporation в США и/или других странах.

Наименования других компаний, продуктов и услуг могут быть торговыми марками или сервисными марками третьих лиц.

Все заявления в данной брошюре о продуктах, решениях и сервисе IBM не означают, что IBM предоставляет их во всех странах.

Все заявления в данной брошюре относительно намерений и перспективных планов IBM могут быть изменены без уведомления.

Анализ ситуации в рамках реализованных проектов, представленных в данной брошюре, иллюстрирует использование отдельным заказчиком технологий и сервиса корпорации IBM и/или бизнес-партнеров IBM. Описанные результаты и выгоды складываются из многих факторов. Компания IBM не гарантирует сопоставимые результаты.

Вся информация о проекте, содержащаяся здесь, предоставлена описываемым заказчиком и/или бизнес-партнером. Компания IBM не удостоверяет ее точность.