

# MOZART

Content Management Framework

## Назначение системы

Mozart - это программный продукт с открытым исходным кодом, написанный на языке Java, распространяемый под лицензией GPLv3, предназначенный для создания и последующего сопровождения веб-проектов любого назначения – порталы, информационные интернет- и интранет-системы, интернет-магазины, каталоги товаров и услуг, B2B торговые площадки, аукционы и корпоративные сайты, онлайн-СМИ и т.п.

Открытость проекта гарантирует его развитие и независимость от конкретного разработчика. Продукт представляет завершенную систему, отлаженную на множестве проектов в течение многих лет.

При помощи Mozart можно легко создавать высокотехнологичные системы управления бизнес-процессами любой сложности, информационно-технической поддержки клиентов и партнеров, эффективного использования банков знаний, развлекательные порталы и социальные сети, словом там, где требуется структурирование большого объема разнообразной информации, управление контентом, персонализация данных. Причем механизмы продукта позволяют легко осуществить реализацию уникальных решений под конкретную задачу.

Надежность, гибкость и безопасность Java-платформы позволят стабильно работать созданному приложению. А возможность использовать в качестве системы хранения данных различные СУБД, включая MySQL и Oracle, позволят более оптимально реализовать необходимые требования приложения, в том числе интегрировав его в какую-то уже существующую IT-инфраструктуру компании.

## Современные технологии

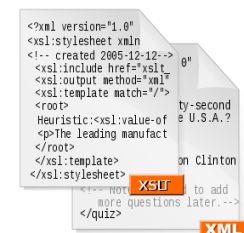
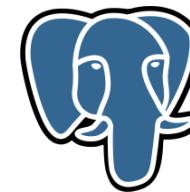
Реализованная на основе открытых технологий Java EE, поддерживаемых такими вендорами, как SUN, IBM, Oracle, система Mozart наследует в своей архитектуре все достоинства стандарта Java EE. Все данные, которыми будет оперировать созданная на основе Mozart информационная система, будут представлены в формате XML — один из основных стандартов обмена данными. Независимо от способа хранения данных, работа внутри системы с данными в формате XML обеспечивает возможность взаимодействия с любыми базами данных. Поддерживаемые стандарты: SQL, DBM, CSV, XML.

Данные обрабатываются механизмом XSL - eXtensible Stylesheet Language, позволяющим применять таблицы стилей к сгенерированным данным, а значит по-разному использовать одни и те же данные. Что позволяет один раз, создав массив обработанных данных, использовать его во всевозможных приложениях, компоновать информацию для различных аудиторий и предлагать пользователю данные в том виде, в каком он предпочитает с ними работать (HTML, TEXT, PDF, RTF, XML, GIF/JPEG/PNG и других форматов).

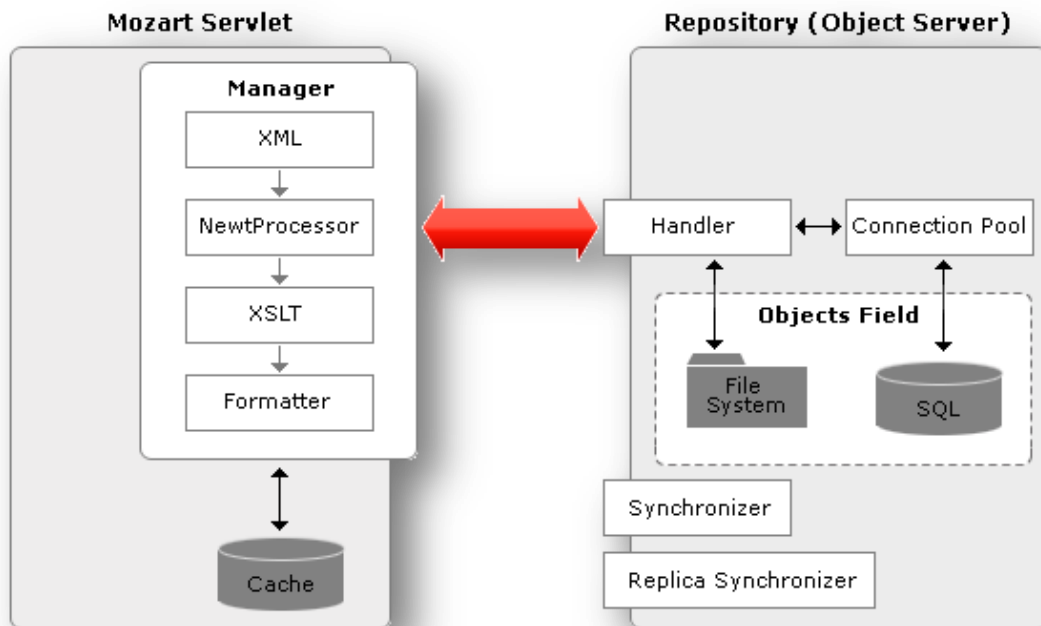
Возможно также обратное преобразование данных из общеупотребительных форматов в формат XML и внесение этих данных в информационную систему. Такая особенность позволяет системе взаимодействовать с распространенными программными продуктами, такими как Microsoft Office.

Кроссплатформенность Java и использованных компонентов позволяет запускать Mozart на различных операционных системах, включая Windows. А открытость исходного кода позволяет существенно расширить возможности по использованию Mozart.

ORACLE®



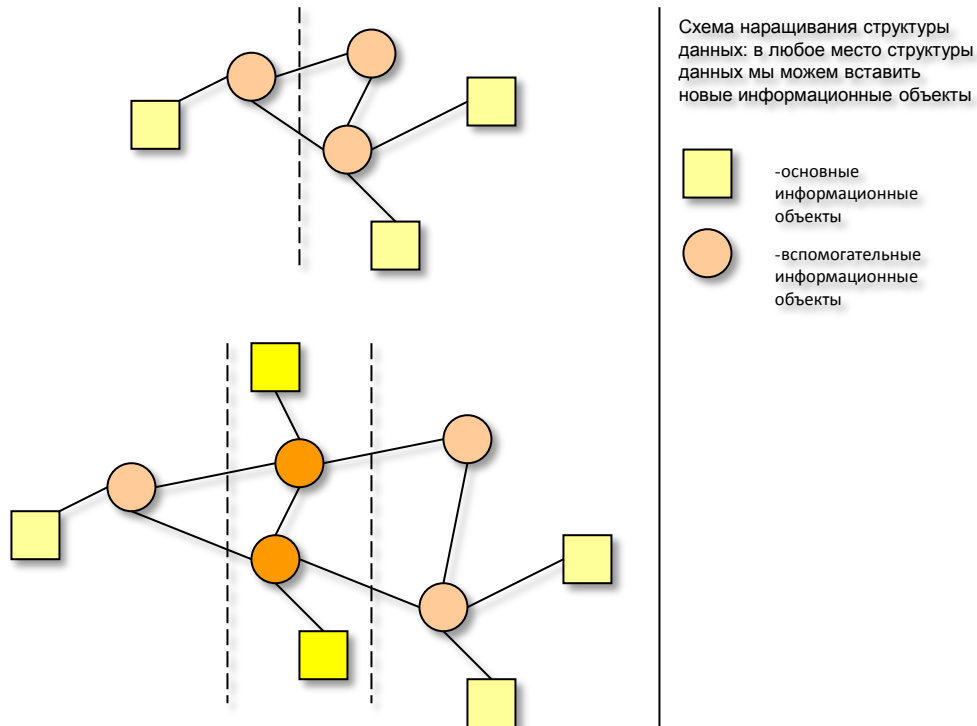
## Архитектура



В объектном хранилище данных (репозитории) систематизируется большой объем разнообразной информации, документов и материалов. Иерархия хранимых объектов описывается графом. Это позволяет связывать информационные объекты в любой последовательности. Репозиторий отвечает на запросы клиентского приложения (в нашем случае Mozart), оперируя информацией об объектах, их связях, взаимодействиях и методах. При этом хранилище можно масштабировать — наращивать и изменять структуру данных. Возможно реализовать практически любые структуры данных. При создании схемы объектов можно оперировать концепцией терминологического поля.

Встроенная система кэширования настроена таким образом, чтобы каждый уровень архитектуры имел свой собственный кэш. Поверх этого работает общий кэш, который содержит сохраненные данные конечной страницы.

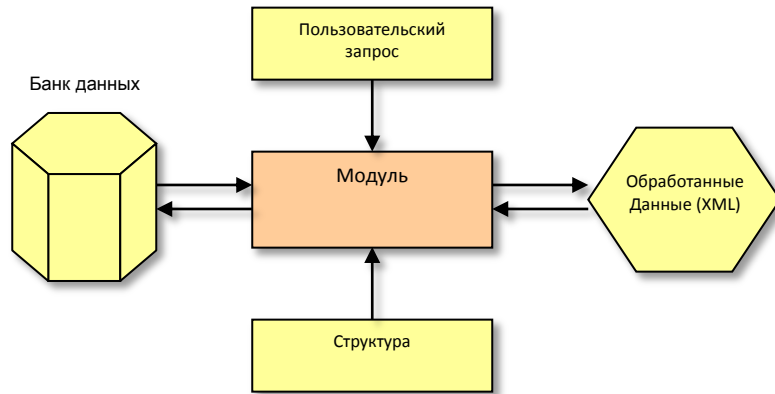
## Банк данных



Разработчику нет необходимости работать напрямую с базой данных. Поверх БД запущен репозиторий, представляющий собой объектный сервер, который оперирует понятием объектов и их связей. В банке данных (репозитории) систематизируется большой объем разнообразной информации, документов и материалов. Иерархия хранимых объектов описывается графом. Это позволяет связывать информационные объекты в любой последовательности. При этом банк данных можно масштабировать – наращивать и изменять структуру данных. Возможно реализовать любые структуры данных, не привязываясь к ограничениям CMS, многие из которых накладывают их из-за особенностей своей реализации. Основным форматом для описания структуры служит XML-формат, который обеспечивают хорошую читаемость и простое оперирование структурой.

PostgreSQL, Oracle, MySQL или HSQLDB, возможна интеграция с любыми другими БД.

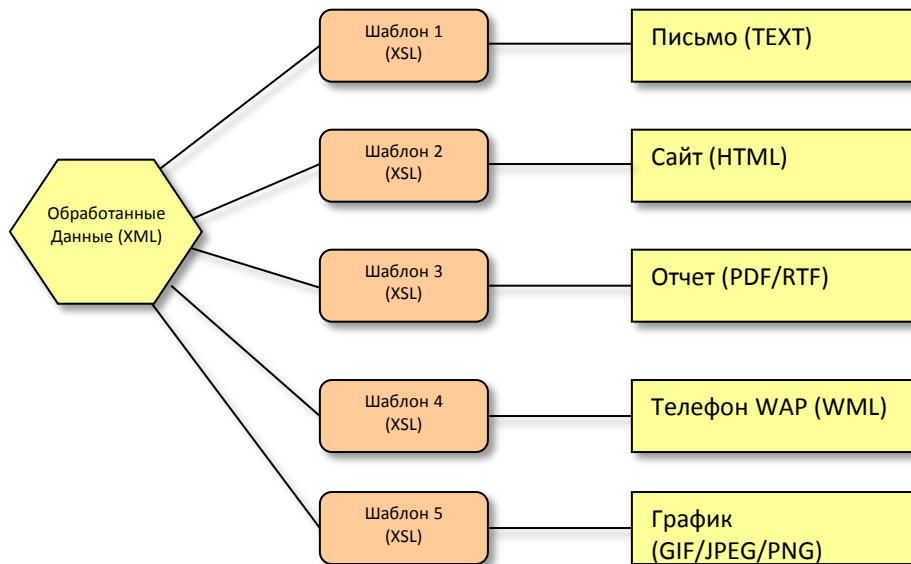
## Модульность



Модульности позволяют оснастить создаваемую информационную систему богатыми функциональными возможностями с возможностью простой дальнейшей доработкой проекта и портированию функционала между проектами. На всех архитектурных уровнях имеется возможность каждый отдельный функциональный блок хранить в виде самостоятельной структуры, подключаемой простыми механизмами `import`'а. Рассматриваем ли мы структура базы данных или функциональных файлов на диске.

Типичным примером модуля является система управления контентом (CMS). Пример работы вы можете увидеть по адресу <http://demo.mozartcms.ru>.

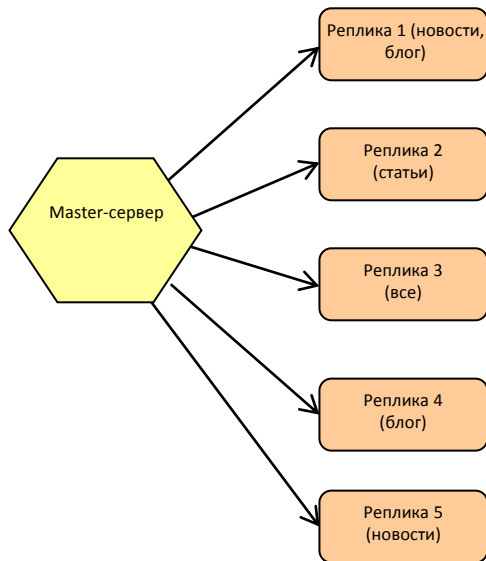
## Шаблоны визуализации



Один раз созданный массив обработанных данных может обрабатываться одним или несколькими шаблонами. Шаблоны визуализации соединяют данные с элементами оформления, свойственных тому или иному формату данных. Таким образом, можно придавать информации любой вид (HTML, TEXT, PDF, RTF, XML и других форматов) и использовать во всевозможных приложениях. В основ шаблонов положен механизм XSL - eXtensible Stylesheet Language. Это известная технология от Консорциума Всемирной Паутины (W3C) заслужила широкое распространение благодаря своей простоте и гибкости.

В ситуации, когда вам необходимо один и тот же документ выводить на разные устройства (принтер, мобильный телефон, карманный ПК), Вы храните свой документ как XML, а для его вывода применить к нему разные XSL-файлы.

## Синхронизация данных



Особенно важной возможностью Mozart является так называемая репликация данных. Механизм репликации данных позволяет вносить любые изменения на тестовом сервере, которые после подтверждения автоматически переносятся на реальный сайт, тем самым избегая любых искажений реального проекта. Вы можете содержать несколько самостоятельных реплик вашего проекта, централизованно управляя частичной или полной информацией на них.

Возможности механизма репликации таковы, что возможно одновременное изменения информации как на тестовом сервере, где разрабатывается/дорабатывается проект, так и на реальном. Никакие данные не будут утеряны при синхронизации.

## Преимущества

Раздельное проектирование информационной базы, программной части (функциональности) и дизайна.

Все данные, которыми оперирует Mozart, представлены в формате XML - последний и основной стандарт обмена данными.

Масштабируемость – простое наращивание и изменение структуры данных в том числе за счет простой реализации модульности. Возможность реализации любых структур данных.

Работа практически с любой базой данных. Настраиваемая пользователем структура базы данных без ограничений из-за специфических реализаций системы управления.

Встроенный механизм управления данными обеспечивает легкое простое и быстрое редактирование информации (CMS).

Независимость функционала от внешнего представления данных позволяет не изменяя программное обеспечение полностью изменить дизайн сайта целиком или отдельные его разделы.

Совместимость с внешними приложениями: возможность ввода/вывода данных в различных форматах (HTML, PDF, XML и других).

Механизм репликации позволяет вносить любые изменения на тестовом сервере, которые после подтверждения автоматически переносятся на реальный сайт, тем самым избегая любых искажений реального проекта.

Скорость и стабильность работы Java EE платформы, которая изначально создавалась как платформа уровня предприятия.

Основной API Mozart нацелен на то, чтобы позволить разрабатывать проекты в максимально короткие сроки. Уровень знаний разработчика зачастую ограничен лишь знанием верстки, т.е. HTML/CSS/JS.

Вспомогательный инструментарий Mozart позволяет создавать функционал с применением различных скриптовых языков (Python, Groovy, PHP, JavaScript), это в том числе значительно упрощает поиск специалистов для работы над проектом.

## Технические требования и поддержка

Для работы с Mozart необходимы:

- Операционные системы сервера: UNIX/Linux или Windows
- HTTP-серверы: Самостоятельный Tomcat, либо связка Apache или Microsoft IIS - Tomcat. (возможны любые другие надстройки в виде nginx и т.п.)
- Базы данных: Postgres, MySQL или Oracle, возможна интеграция с любыми другими БД, а также работа с HSQLDB.

WWW: <http://www.mozartframework.ru>

E-mail: [support@mozartframework.ru](mailto:support@mozartframework.ru)