DATAART
Enjoy IT

Какой фреймворк нам нужен для web

DATAART
Enjoy IT

Постановка вопроса

Зачем нужен web framework?

- В начале были Servlets & JSP
- JSF появляется позже он хорош, но не совершенен
- Мир многообразен и одного фреймворка для всех и навсегда видимо быть не может. Для разных сегментов разные требования (enterprise, public web, mobile web, …).
- Всего существует более 50 фреймворков.

DATAART
Enjoy IT

Возможные точки зрения

- Какой фреймворк будет самым-самым для для проекта (быстрым, простым, масштабируемым и т.п.)
- Какой фреймворк лучше всего выбрать на длительную перспективу (кто автор, community, supportability, перспективы развития, …)
- Какую технологию изчуть, чтобы быть востребованным на рынке труда

DATAART
Enjoy IT

О каких фреймворках поговорим

«Нельзя объять необъятное»

- Struts 1/2
- Wicket
- JSF
- Spring MVC

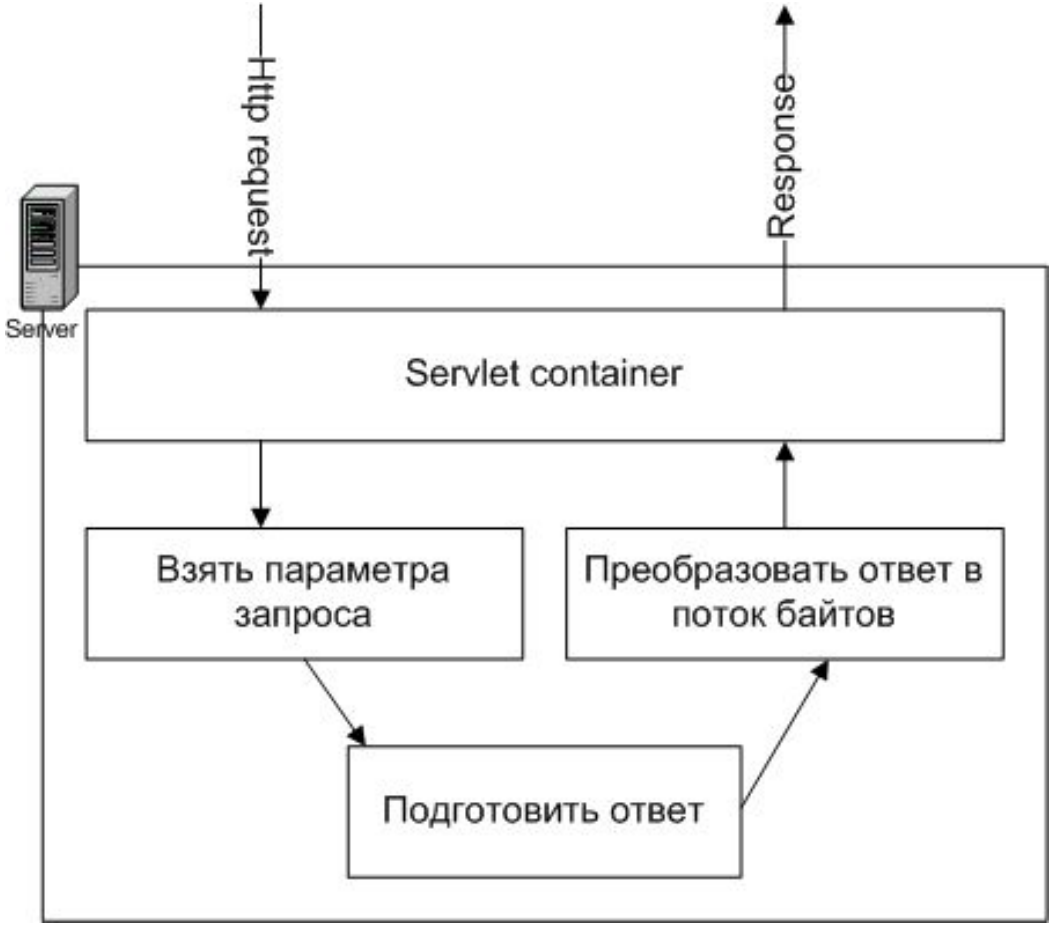
- Хочется больше но время поджимает

DATAART®
Enjoy IT

Что такое web фреймворк с прагматической точки зрения

DATAART
Enjoy IT

Запрос пользователя “stripped”



DATAART Enjoy IT

Например

```
protected void doPost(
        HttpServletRequest request,
        HttpServletResponse response)
                throws ServletException, IOException {

    String name = request.getParameter("name"); //взяли параметры запроса
    String welcomeMessage = "Welcome "+name; //подготовили ответ
    response.setContentType("text/html"); //начали преобразовывать его в поток байтов

    PrintWriter out = response.getWriter();

    out.println("<html><body>");
    out.println("<h1>"+welcomeMessage+"</h1>");
    out.println("</body></html>"); // готово

}
```

DATAART
Enjoy IT

Сервлеты это круто, зачем еще что-то?

Сервлеты и JSP в руках упорного но не изобретательного
разработчика это страшно

DATAART Enjoy IT

Зачем фреймворк

Фактически любой фреймворк в обмен на (один или несколько из пунктов)

- Падение производительности
- Сложности с масштабированием (иногда вплоть до полной невозможности)
- Время на изучение, иногда необходима существенная ломка сознания
- Ограничения функционала и/или совместимости
- Еще что-то плохое ;-)

DATAART®
Enjoy IT

DATAART Enjoy IT

… дает нам взамен

Что-то из нижеследующего

- Декларативная валидация и/или мапинг параметров/путей
- Различного рода удобства и/или гибкость в рендеринге ответа (вплоть до своего декларативного языка описания страниц (ZK))
- Структурирование кода (Model 1, MVC, IoC и т.п.), что уменьшает стоимость владения кодом ибо последущим поколениями программистов проще понять что/где.
- Локализация/интернационализация

DATAART Enjoy IT

… или даже такое

- Продвинутое управлние контекстом (session beans, conversation/page scope, и т.п.)
- Declarative pageflow *(декларативное управление последовательностью переходов между страниц?)*
- Ускоренная разработка UI за счет возможности использования RAD (тут мы скорее говорим о некоторых породах JSF). По моему опыту разработка типовых форм на JSF может ускоряеться до 5 раз
- Автоматизация каких-то традиционно сложных аспектов например AJAX
- Прозрачная, 2-way интеграция с html дизайном (Tapestry, Wicket)
- Подмена (частичная) элементов web стека Java стеком (GWT, WebOnSwing, …) */очень популярная тема, не хотели в свое время Java программисты учить web стек/*
- Упрощенное, декларативное создание web сервисов
- Защита от double POST
- **И многое, много другое …**

DATAART®
Enjoy IT

Сравниваем

DATAART
Enjoy IT

Struts 1/2

- Model 2 фреймворк
- Помощь в обработки и валидации форм
- Относительно простая архитектура
- Легко изучать / расширять / интегрировать
- Популярен, Struts знает куча людей
- Быстрый, не ограничивает сервлет контейнер.
- Документация не супер
- Морально несколько устарел, not hot
- Большая путаница между версиями 1 и 2 при поиске
- Скорее процедурная чем объектная модель

DATAART Enjoy IT

Struts: Выводы

- Эта технология совершенно точно работает
- Но скорее всего делать сайт на Struts будет не так быстро и не так весело. При этом все равно скорее всего Вам не помешает как минимум Spring IoC

DATAART
Enjoy IT

JSF
светлая сторона

- Первый и единственный феймворк от официального производителя, часть JavaEE
- Очень гибкий
- Очень быстро делается UI из «стандартных кубиков»
- Большое сообщество, много книг, много курсов, реально прост в изучении
- Гибкая декларативная валидация форм, гибкий рендеринг ответа (как пример - можно заставить рендерить в Swing для десктопа)
- Большое количество стандартных UI компонентов
- Много других мелких позитивных фишек (типа прозрачный декоаративный AJAX в IceFaces/RichFaces, Enterprise надстройка JBoss SEAM и т.п.
- Создан с учетом ошибок существовавших в тот момент фреймворков, в т.ч. Struts.
- Несколько top industry quality реализаций стандарта

DATAART
Enjoy IT

Казалось бы собаководы рекомендуют?

DATAART Enjoy IT

JSF темная сторона

- В версии 1.x - вообще не пригоден для public web ибо:
  - Потребляет неоправданно много памяти
  - Простые вещи могут неожиданно сильно нагружать CPU
  - Очень большой размер сесии делает фактически невозможной репликацию сессий (если у Вас не bloody enterprise ☺).

DATAART
Enjoy IT

...

- Проблемы с совместимостью версий и интеграцией с JavaEE спецификацией
- Фактически миграция с JSF 1.1 на JSF 1.2 требует миграции версии контейнера, что возможно далеко не всегда, запуск двух приложений на разных версиях JSF требует модификации контейнера.
- Отрисовка произвольного web UI из стандартных компонент может быть очень трудоемка
- Интеграция с JavaScript на странице возможна, но не тривиальна.

DATAART
Enjoy IT

. . .

- Смешивание JSF & JSP превращает web страницы в ад.
- Реализация REST сервисов непроста
- Настройка security также может быть не тривиальна
- Bookmarkable страницы возможно делать с помощью специального разрешения
- Генерирует много лишенего, очень сложно контролировать точное содержание HTML страницы. В том числе может подключать к странице множество не нужных ресурсов

DATAART Enjoy IT

JSF: Выводы

JSF исключительно хорош для быстрого создания интранет приложений с упором на бизнес логику и компонентный подход

Использовать для public web или чего-то сильно нестандартного можно только если Вы действительно очень хорошо понимаете что Вы делаете (и на всякий случай у Вас есть план Б). В общем лучше не надо

DATAART
Enjoy IT

Wicket

- Быстрый, ориентированный на 2- way интеграцию с web дизайнером фремворк
- Хорош для Java (но не для web) разработчиков
- Java код и html плотно связаны
- Активное community, hot topic
- Быстрый, позволяет хорошо контролировать потребление памяти сессией.
- Совершенно точно пригоден для public web

DATAART Enjoy IT

Что же в нем плохо

- Изучать реально сложно, Ваш опыт web разработки на «классическом» фреймворке Вам не поможет и даже немного помешает, многие вещи делаются очень странным образом.
- По умолчанию чудовищные URL страниц, к счастью в последнее время проблема исправляется с помощью специальных аннотаций
- Управление ресурсами не тривиально. Куда правильнее класть HTML, JavaScript и картинки до сих пор предмет обсуждения. Есть несколько вариантов, но все они не без недостатков

DATAART Enjoy IT

Wicket: Выводы

- Если Вы хорошо знаете Wicket и перед Вам не стоит задача быстро расширять команду, то с Wicket Вы будете счастливы
- Но для изучения фреймворк крайне не прост. Если у Вас нет опыта работы с ним, новый проект на Wicket лучше не начинать
- В целом есть варианты и попроще

DATAART Enjoy IT

Spring MVC

- Классический MVC фреймворк
- Начиная с версии 3 избавился от многих своих традиционных недостатков
- Быстрый, можно очень хорошо контролировать работу с памятью, URL, наполнение страниц, рендеринг (позволяет для разных случаев использовать разны способы рендеринга страниц)
- Наверное идеален для REST сервисов
- Spring IoC фактически индустриальный стандарт
- Очень и очень популярен, отличная документация, очень прост в обучении
- Простые вещи делатся просто, сложные сложнее, все логично

DATAART Enjoy IT

И на солнце есть пятна

- Фреймворк достаточно стар, успел набрать плохую карму в ранних реинкарнациях
- В некоторых случаях конфигурация может быть не столь проста
- Проект активно ударился в коммерциализацию, что немного раздражает (хотя в сравнении с JSF – Spring просто ангелы)

- Вообще наверное, есть еще недостатки, но мне сейчас сложно ругать Spring MVC, потому что он мне субъективно нравится. Недостатки Spring MVC взятые из Интернет относятся к ранным версиям.

DATAART
Enjoy IT

Тут я еще много что хотел рассказать, но время поджимает, если Вас интересует тема, расспросите меня например про GWT в перерыве

DATAART Enjoy IT

Тенденции

- HTML 5 шагает по планете
- Всё больше людей знает JavaScript
- JQuery стал крайне популярен
- Google научился при поиске частично выполнять JavaScript на страницах
- Все хотят запускать сайты в облаке на куче дешевых серверов по $20 за пучок

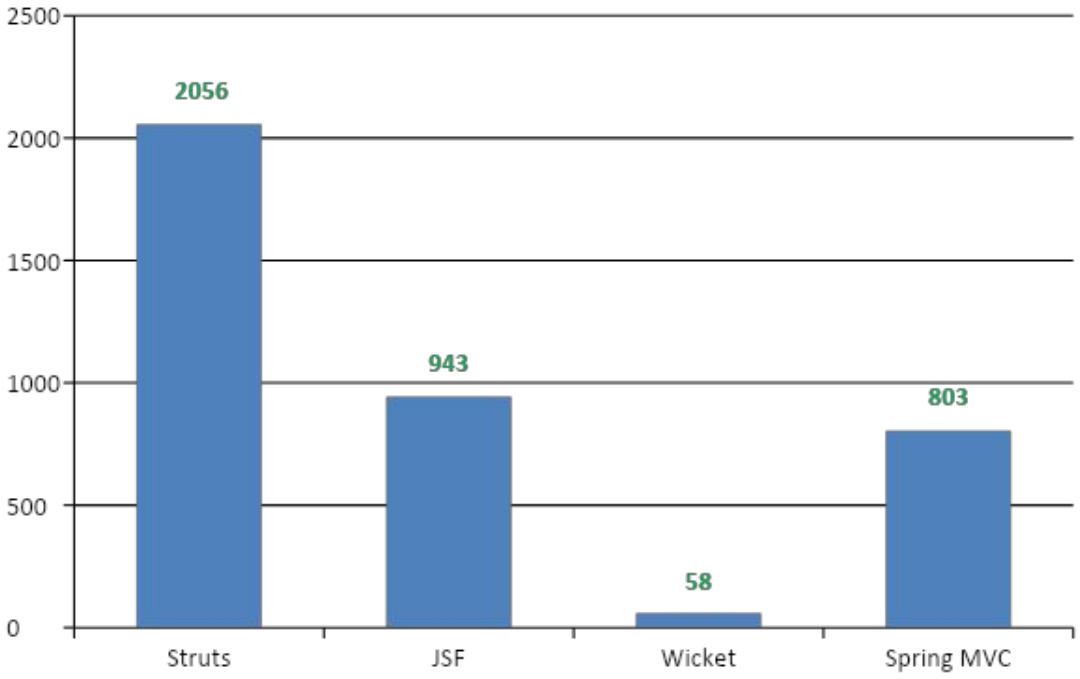
DATAART
Enjoy IT

Мой персональный выбор

- Enterprise – JSF 2 AND ((Spring IoC + Spring …) OR (EJB 3.1 + SEAM))
- Public Spring MVC + Spring * без ORM со Spring JDBC. Rich Internet Application много JQuery, JQuery плагины и много AJAX.

DATAART
Enjoy IT

Куда пойти учиться?



По данным сайта dice.com

DATAART
Enjoy IT

Вопросы

DATAART
Enjoy IT

Спасибо

DATAART®
Enjoy IT

DATAART®
Enjoy IT

Вдруг хватит
времени???

DATAART Enjoy IT

Tapestry

- Быстрый, ориентированный на 2- way интеграцию с web дизайнером фремворк
- Комонентно ориентирован, причем компоненты могут наследоваться друг от друга
- Код пишется быстро
- Ориентирован на решение практических задач, быстрый, экономно расходует память. Есть примеры очень удачных сайтов на tapestry с большим кол-вом посетителей
- Позволяет контролировать html страницу с точностью до байта. Что бывает крайне необходимо для mobile web сайтов.
- Содержит много полезных фичей, расширяем.

DATAART Enjoy IT

Что не хорошо

- Не прост для изучения, есть ряд нетривиальных моментов требующих перестройки сознания, документация отвратительная, литературы мало
- Фактически Tapestry это один человек который лидирует развитие фреймворка с самого начала
- Не очень популярен
- Community не развито
- Получить четкие URL для всех страниц приложения все еще не тривиально, раньше было еще хуже ☺
- Некоторые очевидные вещи могут делаться достаточно неочевидным образом

DATAART Enjoy IT

Tapestry: Выводы

- Если Вы хорошо знаете Tapestry и перед Вам не стоит задача быстро расширять команду, то с Tapestry Вы будете счастливы
- Но для изучения фреймворк крайне не прост. Если у Вас нет опыта работы с ним, новый проект на Tapestry лучше не начинать