



++  
Российские  
интернет-технологии  
2010

# Смена web- платформы «на лету»

Евгения Фирсова

# Постановка задачи



# Постановка задачи

Цель: смена среды исполнения.



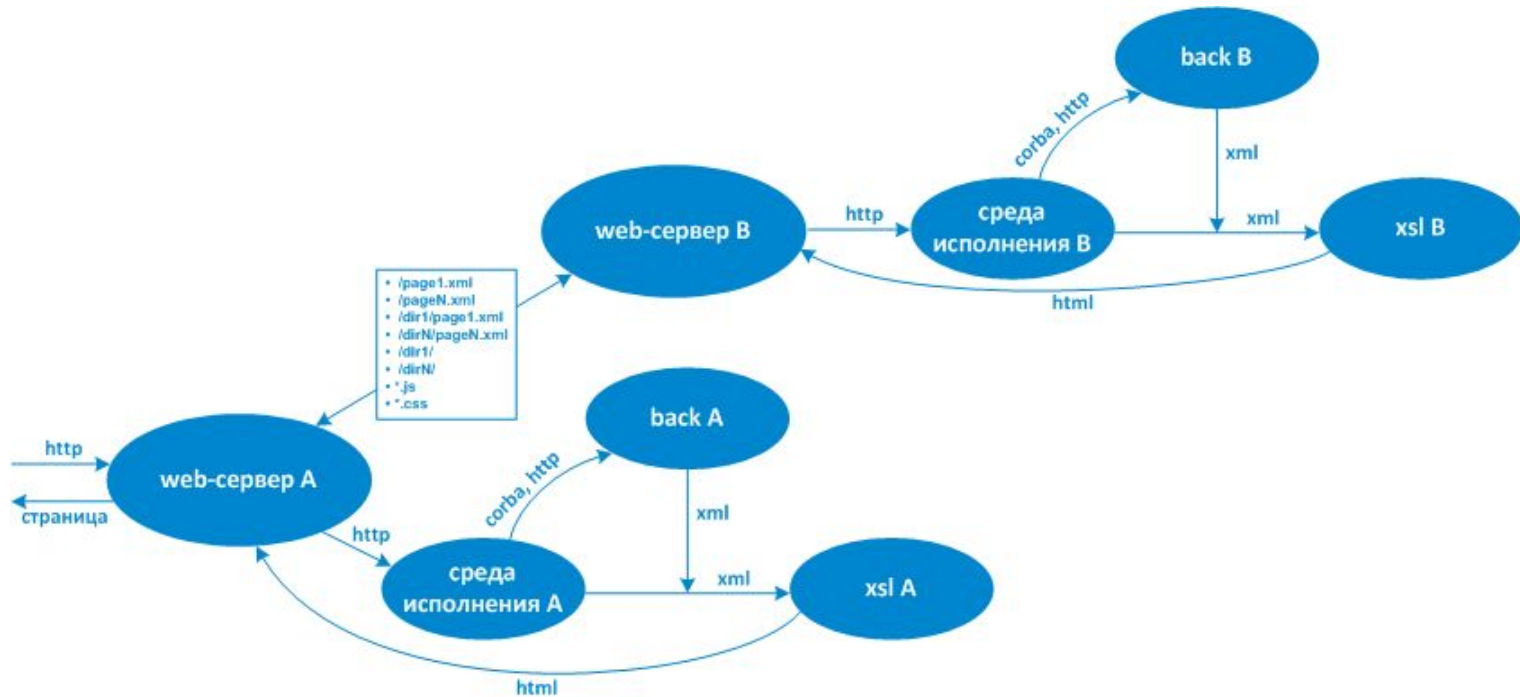
Проблемы:

- ✓ несовместимость новой среды исполнения со старым кодом;
- ✓ ограничение ресурсов.

# Ограничения, метрики

- «прозрачность» для пользователей:
  - отсутствие простоев;
  - отсутствие новых требований к оборудованию/окружению пользователя;
- длительность «переходного» периода:
  - затруднения при внедрении нового функционала;
  - планирование «на потом»;
- качество работы в «переходный» период:
  - новые требования к участникам процесса;
  - масштаб усилий по синхронизации двух технологий.

# Проксирование запросов





# Выбор web-сервера «на входе»

	старый web-сервер «на входе»	новый web-сервер «на входе»
использование преимуществ нового web-сервера	только после окончания «переходного» периода	внедрение при переводе каждой новой страницы
риски в «переходный» период	ничтожны; используем привычные технологии	велики; экспериментируем, учимся, набираемся опыта
риски по завершению «переходного» периода	велики; непредсказуемость процесса переключения web-серверов	отсутствуют; выключаем ставший ненужным web-сервер
момент выявления проблем	после финального переключения web-серверов	сразу, как только они возникнут
суммарный объем работ	новая версия каждой страницы + тестирование каждой страницы + повторное тестирование после переключения web-серверов	новая версия каждой страницы + тестирование каждой страницы
		✓ выбор сделан

# Выбор «добровольцев»

	критерии	риски	скорость
1-я страница	<ul style="list-style-type: none"> <li>✓ возможность отката</li> <li>✓ заинтересованность заказчика</li> <li>✓ простота в реализации</li> </ul>	●●●●●	●○○○○
2-5-я страницы	<ul style="list-style-type: none"> <li>✓ возможность отката</li> <li>✓ максимальная вариантивность функционала</li> <li>✓ средняя нагруженность</li> </ul>	●●●●○	●○○○○
6-10-я страницы	<ul style="list-style-type: none"> <li>✓ крупный важный функционал</li> </ul>	●●●●○	●●●○○
остальные	<ul style="list-style-type: none"> <li>✓ новый функционал или значительные изменения в существующем</li> <li>✓ экономия на тестировании</li> <li>✓ возможность рефакторинга</li> </ul>	●○○○○	●●●●●

# Риски

- **внутренние:**
  - ошибки при внедрении новых технологий;
  - неоптимальное использование новых технологий;
  - потеря функционала при переводе на новые технологии;
  - непредсказуемость тестирования;
- **бизнес:**
  - срыв сроков;
  - «публичность» ошибок;
  - несовпадение обещаний и результатов;
- **организационные:**
  - отложенные проблемы с производительностью;
  - ошибки в выборе новых технологий.



# Что влияет на оценку сроков

- ↑ чем ближе мы к началу «переходного» периода, тем сильнее ошибаемся:
  - в оценке скорости разработки;
  - в оценках скорости и качества работы тестеров;
- ↑ вероятность поддаться соблазну фонового проведения рефакторинга;
- ↑↓ отказ от рефакторинга в пользу скорости внедрения функционала;
- ↑ сходимось процесса разворачивания и поддержки новой среды;
- ↑ непредсказуемый рост портала за «переходный» период.

# «Пересечения» кода

- код, без изменений работающий в обеих средах исполнения:
  - подключаем из единого источника:
    - физическая адресация к единой версии;
    - средства системы контроля версий: амперсанд-модули `cvs`, `svn:externals`, ...
    - при сборке пакета/версии для релиза;
  - (вероятно) двойное тестирование;
- код, требующий изменений для работы в обеих средах исполнения:
  - планируем (не)синхронные изменения в старом коде;
  - двойное тестирование;
- несовместимый код, выполняющий одинаковые/близкие функции:
  - отслеживанием и синхронизируем изменения;
  - двойное тестирование.

# Синхронизация изменений

- строго синхронно:
  - единый визуальный или функциональный элемент на страницах, работающих в двух средах исполнения;
  - поддержка процесса, требующего перехода по страницам, работающим в двух средах исполнения;
- первое внедрение изменений – в новый код:
  - решаем, понадобится ли когда-нибудь изменение в старом коде;
- первое внедрение изменений – в старый код :
  - решаем, не стоит ли сразу перенести функционал в новый код:
    - оцениваем стоимость перетестирования;
    - оцениваем выгоды от использования новых технологий при реализации изменений;
  - планируем отличия в реализациях для старой и новой сред исполнения.

# Специфика тестирования

- рост объёмов тестирования:
  - проксирование;
  - пересекающийся код;
  - рост числа компонент;
- необходимость отслеживать, на какой среде исполнения работает функционал;
- проверка конфигураций проксирования;
- контроль за правильностью переноса конфигураций проксирования.

# Синхронные выкладки/релизы

- временное отключение части хостов от боевой среды:
  - необходим пул хостов;
  - половина пула должна выдерживать двойную нагрузку;
- последовательная выкладка закрытого «рубильниками» функционала с последующим синхронным включением:
  - тех. возможность реализации «рубильников»;
  - разработка и тестирование «рубильников».

# Конец «переходного» периода

- финальное переконфигурирование проксирования запросов между web-серверами;
- отключение старого web-сервера;
- чистка кода от ненужных «рубильников»;
- ✓ долгожданный подсчёт цыплят:
  - качество выполнения поставленной цели;
  - длительность «переходного» периода;
  - что мы получили, помимо заказанного:
    - качественный код после чистки и рефакторинга;
    - опыт работы с новыми технологиями.

# Вопросы?

[saigo@yandex.ru](mailto:saigo@yandex.ru) – задать любые вопросы

[www.control-freak.ru](http://www.control-freak.ru) – почитать подробнее  
об управлении версиями, задачами, проблемами и людьми