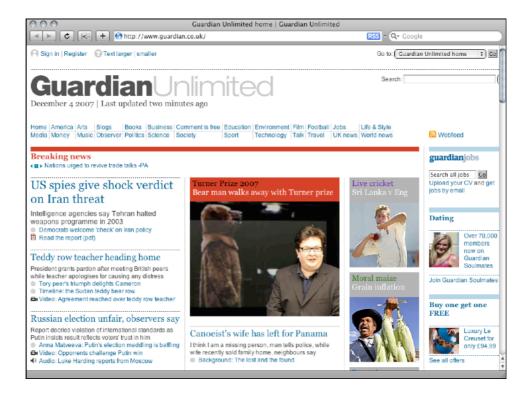
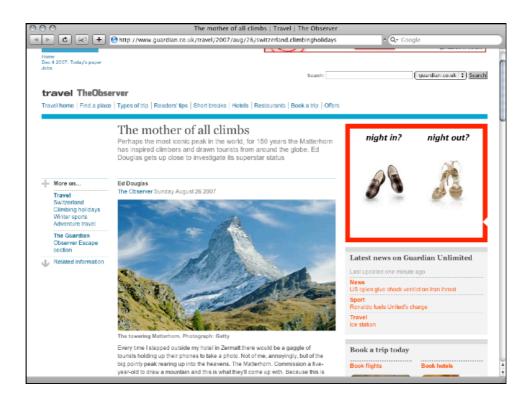
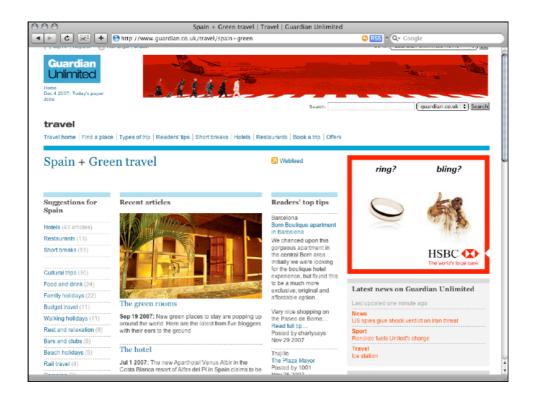
the new guardian.co.uk

Mat Wall Technical Architect Guardian Unlimited Erik Dörnenburg
Principal Consultant
ThoughtWorks







The Stack

Contemporary Java Open Source Software stack

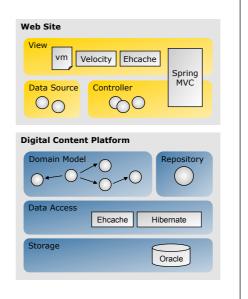
Key frameworks:

 Spring 2, Hibernate 3, and Velocity 1.5

Other frameworks:

 YUI/JSON, Ehcache, ROME, C3P0, etc

Developed in Jetty and deployed in Resin



Guardian Unlimited

It's that simple

Business drive design with DDD

- clear and explicit model of the domain
- visual representation of complex relationships

Ubiquitous language

- little bit of formality lots of value
- even techies speak the same language

Domain is reflected in all code

- does have momentum, but can change
- non-Java developers write velocity templates

**

Simplicity is in the eye of the beholder

Each transaction in the admin app changes the site

- but we are using caches in the frontend app
- tx must be coupled with a clear cache instruction

Simplistic solution

- · Manage tx and clearcache inline in code
- Places burden on developers. Limits refactoring.

More complex solution

- Uses an aspect and request scoped bean
- Cleaner solution and a simple life for developers

Simple = Cannot express intent any clearer

Guardian Unlimited Clear Cache solution injected (request scoped) ClearCache Message Builder Aspect: around advice **Transactional** Controller creates uses and sends commits changes ClearCache Message Database

_

Simple != simplistic

Pages are made up of components

Initial approach

- Controller provides all required domain objects
- How do controllers share and re-use code?
- Where do we put complex rendering logic?

Data Sources empower designers

- Velocity devs can design their own API for domain
- Slightly more complicated approach reduces overall complexity

**

Guardian Unlimited

Performance

150m+ hits per month, all dynamic

- Pages feature complex relationships
- Cannot write page de-caching logic

Site uses two caching systems

- EHCache for Hibernate objects
- HTML component cache

How to retain good performance:

- Simple design, don't add other caches
- Continuous performance tests



URL Handling

Once published, always available

Custom apache module allocates URLs

different back end servers

Want non technical, RESTful URLs

- http://www.guardian.co.uk/story/0,,125425,0.html
 - http://www.guardian.co.uk/news/iraq/2007/06/28/cabinet
 - http://cms.guprod.gnl/article/125425/edit

R2 migration will redirect over 1,000,000 URLs

- Permanent redirects (301) written to disk
- Served by mod_asis

ж

Questions

