GT2004Bertrand

Bertrand Delacretaz: Cocoon: what do yo need to know

Overall structure of a cocoon app shortest path to an app example is nouvo.ch website of a Swiss TV show - discussion forums pages, forums, nothing special

multi-screen, access via the web and via mobiles loose sheet in handouts shows basic structure of app. We will follow the steps, and this will show you what you need to know about cocoon.

0. request from the web, hits Apache good idea to put apache in front, security, uptime. mod_proxy, mod_cache

1. instance of cocoon

hits the sitemap, tells cocoon what to do with it 2. hits flowscript flow and forms 3. business logic java code that decides what to do with the request document store vote manager 4. using ojb to map to database very powerful, performant 5. back to forums manager 6. back to flowscript 7. XSLT, Cinclude, i18n, LinkRewriter, XSLT Serializer 8. back to servlet 9. back through apache 10. back to client

User Sees:
HTML + CSS

Using
XSLT, Flow, CForms, Java biz logic, OJB for data access Not the only way

4. Presentation Layer

You don't need to know java biz objects to use xml docs in cocoon. You need to know:
Sitemap config
XSLT
HTML + CSS
his media students learn this in 6 days
they are not it specialists (0 xml knowledge)
so it's not hard

media students are scared by the idea of editing at the commandline
you only need to know a little of the sitemap to make a start
students are taught not about all details
you don't need to know all the details to start with cocoon.

Cocoon concepts used in nouvo.ch:

- i18n transformer maps keys to messages
- linkrewriter (for translating semantic, named links to real HTML links)
- Flowscript as glue: holds together the business logic of your application:
Uses javascript syntax. yet another language, because it works. Only way to do continuations easily. Just a few lines of code. Get a component, error handling, send a page.
Continuations are good to handle 'loose connections'.

7. Cocoon Forms

You need 3 def files

- form model
- binding
- form template

the form framework takes care of load-edit-validate-save cycle until it is happy with the business object.

less code, but more time to find out where you need the five lines of code that will do the work. Need to take the time to find out where you have to put your code in cocoon. Usually application specific.

8. Business Logic

Usually in Java. POJO. Ugo will later show even better way to separate business logic.

9. Database Access

OJB rocks (except the name...). Simple for simple things. Only need to write xml mapping file to map objects to fields. Setup requires SQL/JDBC skills. This is one way to do it, works for him

Testing is important Easy to write testable code People are fed up with software that half works Software cannot be better than the tests that you write testing javacode with JUnit testing the front end via HTTPUnit or Anteater

10. Cocoon Blocks

Can be scary. Nouvo uses only two blocks. Linkrewriter and Forms. Don't worry about the blocks. Most of them are exotic. You don't need most of them

11. Skills - Installation

Not sure who cocoon is for. Should be easier, or not. It's better to fail early? Command line is where you do the work. Don't expect buttons. No need to be a commandline guru, but basic skills are required.

1. install JVM

2. unpack Cocoon 3. build cocoon 4. run cocoon

You need a JVM, 1.4, 1.3 still supported. Download cocoon and please read Install.txt!

Enough info in the install.txt to get you going.

No need for tomcat or apache httpd. Play with the samples and look at the logs (don't be shy)

12. Where to start

Supersonic tour block(written by Bertrand) is an overview of how cocoon works Documentation tracks on the website that show you how to explore the samples.

Explore explore explore!!!! Without this, you will not learn You have to be willing to be digging.

13. Skills - Presentation

Need to learn XSLT - very strange language for procedural programmers .... they need to change their mindset to work with the language XSLT is event based.

Cocoon sitemap. Start with simple, not very deep pipelines.

CInclude, bring in external feeds

i18n - define the messages of your app

HTTPUnit/Anteater - testing the front end behaviour

14. Skills - Forms

You need to understand the forms concepts. These are simple for simple problems, but easy to customize if you need to do complex things.

but you can do a lot without customisation

15. Skills - Backend development

Java coding, OJB, some SQL Creating and setting up your db You need to be able to create components, which now means avalon components. Not very scary, just a bad reputation. If you need to create e.g. a Transformer it's easy to use an existing one as a template. Maybe in the future you can use Spring. Use Lucene to replace db searching, index the db into Lucene

Need to be able to test the code: JUnit

16. Skills - Debugging

Very important to debug, separate chapter. If you can pinpoint your problem it's easier to get help! You have to understand log-configuration.

Need to understand how to change the log setting Learn log filter tools, grep, sed, awk, tail Run cocoon in debugging mode, it's not hard, cocoon scripts are prepared for remote debugging. Also useful for understanding how the cocoon internals work at the java level.

Screenshot of Chainsaw (comes with log4j) GUI tool for viewing and filtering logs in realtime

17. Skills - Integration

If you remember one thing, remember this: The hard part is the integration. Many tools and libraries play a part in your application.

Need someone who can have an overall view of the whole app, for integration and debugging, not everyone on the team needs to be able to do this.

18. Will you make it?

Of course - if you are willing to invest.

No one knows everything about Cocoon. You need to know a bit about a lot of things.

Learn as you go. Go in small steps

Need strong Architect/Integrator

19. Will it be worth it?

Yes, if you are comfortable with the command line, if you take the time to explore (the jungle). You need a long term vision for the complexity of cocoon to pay off.

Suitable for heavy lifting, but can be used for simple projects, especially if you are planning to take the project further in the future. Invest in your skills, learn learn learn.

20. Conclusions

Cocoon is like the toolbox of a very rich man. Which tool do you need? Take time to select the right tool. Otherwise you get lost. Don't be distracted by what is all around you. Full blown apps need a wide range of skills.

Need to be an explorer, and be happy always learning

21. Coda

Starting up? Look at supersonic tour. If you have the skills take the time to explore Usually people who take their time with cocoon like what they see We're going to see amazing things.

Q: Is an architect the same thing as an integration guru? A: an Integration Guru is more what he meant. Also an architect who doesn't want to get his hands dirty should really change his ideas.