Cocoon Features (2.1 legacy document)

Table of contents

Copyright © 1999-2005 The Apache Software Foundation. All rights reserved.

Table of contents

Copyright © 1999-2005 The Apache Software Foundation. All rights reserved.

Warning:
This document was copied as is from the Cocoon 2.1 documentation, but has not yet been fully reviewed or moved to its new home.

1. General information

- Apache Cocoon is a web development framework built around the concepts of component-based web development and separation of concerns, ensuring that people can interact and collaborate on a project without stepping on each other toes.
- Cocoon implements these concepts around the notion of **component pipelines**, each component on the pipeline specializing in a particular operation (usual pipeline uses a Generator, Transformers and a Serializer). This makes it possible to use a Lego(tm)-like approach in building web solutions, hooking together components into pipelines without requiring programming.
- **Advanced Control Flow**: continuation-based page flow hides the complexity of request/response processing and is cleanly separated from the view and data components.
- Cocoon is open source software (based on the Apache Software License).
- Cocoon does not duplicate efforts but tightly integrates many technologies.
- Cocoon is in use at many live sites and on many company networks.
- Cocoon has a strong community, with many active developers and more than 15 active committers!
- There is free support from the thousands of people on our mailing lists and commercial support is available from various companies and consultants.
- There are many Cocoon sessions at different conferences:
  - Cocoon GetTogether
  - ApacheCon
  - Austrian Cocoon Day
  - WJAX
  - JAX
- To get started see the "first steps" documentation track. Basically you only need to download Cocoon, unpack it and follow the simple INSTALL.txt instructions. A minimal version of the Jetty servlet container is included with Cocoon.

2. Usage scenarios

As you would expect, all of these scenarios can be combined.

- Dynamic multi-channel web publishing (see below for the possible datasources and output formats)
- Create static content (automatically) by separating data from view
- Offline generation modes with Cocoon's own offline facilities: command-line interface (CLI), ant task, bean. Also with Apache Forrest which utilises Cocoon.
- Dynamic document preparation with Apache Forrest, the 'forrest run' mode. Use many different data input formats, see the transformed result immediately in the browser.
- Advanced web applications with J2EE integration (with separation of your data, the view and the flow logic --> this really means you can change one of the parts without touching another)
- Develop your company portal using the Cocoon Portal framework
- Support multiple clients, layouts and languages (i18n) without code duplication
- Integrate Cocoon with your existing web applications or use it to put a better face on them (page scraping)
- Add full-text search to any datasource that can be converted to XML (see below)
- Use Cocoon as the base for Enterprise Application Integration (EAI)

Copyright © 1999-2005 The Apache Software Foundation. All rights reserved.

- Use Cocoon as the base for your Content Management System (CMS) (see Apache Lenya for a Cocoon based CMS)
- Use Cocoon for producing mobile content (mobile phones, pdas)
- Datawarehouse reporting across multiple formats (see xReporter)

3. Connect your datasources

Out of the box, the following data can be converted to XML to be processed by Cocoon pipelines.

- XML Files
- XML based (Web) services
- RDBMS (via JDBC, including connection pooling)
- XML databases
- SAP (r) Systems by adding the SAP JavaConnector see http://service.sap.com/connectors/ (accessible for all SAP (r) customers)
- WebDAV
- CVS (supported by the external project CVSSource)
- Text-based file formats, either using the integrated Chaperon parser for a yacc-like approach to parsing, or the "slop" component (Simple Line Oriented Parser).
- Velocity templates
- JXPath/Jexl templates
- eXtensible Server Pages (XSP) with wide range of logicsheets (database, mailing, ...)
- Python (Jython) and generic BSF support
- JSP
- Filesystem (traversing directory hierarchies)
- Any information provided by environment (request, session)
- Flash
- XMidi
- LDAP - Lightweight Directory Access Protocol
- Easily aggregate different datasources

4. Transform your XML based on standards

- XSLT (The default XSLT-Engine is Apache Xalan, XSLTC is included in the Cocoon distribution, other XSLT-Engines like Saxon can be easily integrated)
- STX (Streaming Transformations for XML)
- XInclude with XPointer framework support

5. Serialize your XML to various output formats

- XML
- HTML
- XHTML
- PDF
- OpenOffice.org/StarOffice
- MS Excel
- RTF
- Postscript
- Charts (see external project Fins)
- Flash
- Plain text
- Scalable Vector Graphics (SVG)

Copyright © 1999-2005 The Apache Software Foundation. All rights reserved.

- MIDI
- ZIP archives

6. What else we can do for you

- Coexist and interoperate side-by-side with your existing J2EE solutions (EJB, JMS, ...)
- Build your Portals based on Cocoon (expect support for JSR168 soon)
- Scheduler - Run background tasks for maintenance, etc.
- Caching on many levels
- Integrated search engine (using Lucene)
- DELI (detect client configuration)
- Catalog Entity Resolver to map to local copies of DTDs and other resources
- Publish your own WebServices (Apache Axis is integrated)
- Java Mail support
- Easy integration of object-relational frameworks (OJB, Hibernate, ...)
- I18n support (translation support)
- Easily extensible by clear interfaces (write your own components following Avalon patterns)
- Many, many examples and samples
- Configurable build mechanism based on Ant (you decide which parts of Cocoon you need)
- Integration of Java data binding frameworks (Castor, Betwixt)

7. Form handling frameworks

- Enhanced form handling with strong validation through Cocoon Forms
- Easy integration of (future) XForms clients

8. Cocoon deployment and integration

- Cocoon can be run in every servlet container or J2EE application server that supports Java Servlets 2.3 and above, e.g. Tomcat, Jetty, JBoss JRun, Resin, Websphere, Weblogic, ...
- Command line execution, without requiring a servlet container
- Embeddable in any Java application

1. Comments

add your comments

Copyright © 1999-2005 The Apache Software Foundation. All rights reserved.