Hlxml Documentation

1. Project Docs

1.1. Half-Life Server Logs to XML Converter

1.1.1. What the *** is this?

I wrote a small command-line tool in Java, that reads Half-Life Server logs and transforms them into a XML Representation.

The project is hosted on SourceForge.

-- jan

1.1.2. Features

- SAX XMLReader Implementation
- XML Configuration with HiveMind
- Can be used for other log formats as well, basically everything that can be parsed using a set of Regular Expressions.

1.2. History of Changes

RSS

1.2.1. Version 0.3.1 (CVS only)

- Added generation of HLSW Player DB text import format files (jawe)
- Changed a lot of internal classes to be HiveMind services (jawe)
- Externalized and localized most of the messages using HiveMind's localization mechanism (jawe)
- Split HiveMind modules into core and cstrike. The core module is intended to be reusable for virtually any kind of logfiles (jawe)
- Minor documentation updates (jawe)

1.2.2. Version 0.3 (2004-08-30)

- Initial CVs Import and upload to sf.net. (jawe)

- Reads single log files or whole directories. (jawe)
- Generated XML can be written to a directory. (jawe)
- Nice command line interface. (jawe)
- Created and uploaded the homepage (= documentation) using forrest. (jawe)

1.3. Todo List

1.3.1. medium

- **[parser]** Add Ant targets for compiling JUnit Tests. # jawe
- **[parser]** Write/expand JUnit tests. # jawe
- **[parser]** Implement a SAX ContentHandler that accumulates statistics. # jawe
- **[reader]** Implement receiving logs via rcon. # jawe
- **[stats]** Generate HTML displaying the stats. # jawe

1.3.2. low

- **[www]** Write developer docs: Architecture Overview, Configuration, Guide. # jawe
- **[project]** Try Maven. # jawe

1.4. Hlxml Roadmap

1.4.1. Release 0.4

This version will include the ability to gather overall stats.

1.4.2. Release 0.5

This version will include the ability to gather specific stats (players, maps, weapongs etc.).

1.4.3. Release 0.6

This version will generate a website displaying the stats.

1.4.4. Release 0.7

This version will include the ability to receive and parse logs via rcon.

2. User Docs

2.1. Hlxml User Documentation

2.1.1. Purpose

Hlxml is the first step of an attempt to implement a XML based player stats sytem for half-life servers.

2.2. Hlxml Getting started

2.2.1. Preparation

1. Download a distribution package
2. Install it using a compression tool like WinZip, unzip, tar etc.
3. Make sure you have some Half-Life server log files handy.
4. Choose or create a directory to write the generated XML files to.

2.2.2. Usage

Note:

At the moment the program is only able to simply write the generated XML to files in a directory. Later it's main purpose will be to pipe the XML to a stats generator.

Open a command line prompt and type `java -jar hlxml.jar --help` to get a short description of the available command line options.

See the Usage Notes for more information.

2.3. Hlxml Download

2.3.1. Download

All files can be downloaded from the SourceForge project site.

2.3.2. Distributions

Binary
hlxml-<version>.<extension>
Documentation
hlxml-doc-<version>.<extension>
Source
hlxml-src-<version>.<extension>
Source without libraries
hlxml-src-nolibs-<version>.<extension>

2.4. Hlxml Installation

2.4.1. Installation

1. Download a distribution package.
2. Extract it into a directory of your choice using a compression tool like WinZip, unzip, tar etc.

2.5. Hlxml Command line usage

2.5.1. Usage

Change to the program directory (the one where you extracted the distribution package) and type `java -jar hlxml.jar --help` to get the following output:

```
java -jar hlxml.jar [options] [files]
 where options are:
 -h,--help                          display a help text and exit
 -o,--output <Ausgabeverzeichnis>   a directory to write the xml files to
 -p,--parser <Parser Typ>           parser type, currently only cstrike is
                                    supported
```

2.5.2. Parsing a single logfile

Change to the program directory and type:

```
java -jar hlxml.jar -o /path/to/output-directory /path/to/logfile.log
```

2.5.3. Parsing all logfiles in a directory

Change to the program directory and type:

```
java -jar hlxml.jar -o /path/to/output-directory /path/to/logfiles
```

3. Developer Docs

3.1. Hlxml Developer Documentation

3.1.1. Architecture Overview

TODO

3.2. Hlxml Parser Configuration

3.2.1. Apache HiveMind

TODO

3.2.2. Matching Log Entries: match

TODO

3.2.3. Matching Tokens: choose and when

TODO

3.2.4. Reacting on matches: do

TODO

3.3. Hlxml API Documentation

3.3.1. Guide

TODO

3.3.2. Reference

Click here to view the Javadoc generated API documentation

3.3.3. Standards Compliance

TODO: Describe use of SAX, JAXP and TRAX APIs.