llvmparse user manual

Title	llvmparse
Description	LLVM assembly parsing standalone utilities
Author	Nikolaos Kavvadias 2012-2013
Copyright	Nikolaos Kavvadias
Contact	nikos@nkavvadias.com
Website	http://www.nkavvadias.com
Release Date	25 February 2013
Version	1.0.0
Rev. history	
v1.0.0	25-02-2013 First commercial release.

1. Introduction

`llvmparse` is a set of standalone parsers for the popular LLVM textual assembly format. This format is used as a human-readable representation of the LLVM IR (intermediate representation) which is used by the LLVM compiler (http://www.llvm.org).

This version of `llvmparse` includes two standalone parsers:

1) a flex/bison-based parser (compiled) and

2) a TXL-based parser (interpreted)

that are compatible with LLVM 3.1 and 3.2. TXL is a functional programming language that is commonly used for source-to-source transformation tasks (http://www.txl.ca).

`llvmparse` will always be kept up-to-date with the latest official LLVM release.

2. File listing

The llvmparse distribution includes the files listed in the following table. There is a single release for llvmparse containing all source code, Windows (MinGW) binaries as well as Linux static binaries.

/llvmparse	Top-level directory

build.sh clean-all.sh env-lin.sh env-win.sh test-all.sh	Automated build script. Clean the generated debris files in /tests. Script for setting up the environment on Linux. Script for setting up the environment on Windows. Exercise all test cases found in /tests.
/bin/lin	Linux executable directory
llvmparse.exe	Static executable of the flex/bison-based parser for Linux platforms.
/bin/win	Windows executable directory
llvmparse.exe	Static executable of the flex/bison-based parser for Windows platforms.
/doc	Documentation directory
AUTHORS LICENSE LLVMOSL.TXT README README.html README.pdf VERSION	List of authors. End-user license agreement. The LLVM source code open-source license. This file. HTML version of README. PDF version of README. Current version of mprfgen.
/src	Source code directory
Makefile llvm.Grm llvm.l llvm.y llvmparse.Txl llvmparse.c	Makefile for building the flex/bison-based parser. TXL grammar for the LLVM assembly format. Flex/lex scanner for the LLVM assembly format. Bison/yacc grammar for the LLVM assembly format. Driver TXL program for the TXL-based parser. Driver C program for the flex/bison-based parser.
/tests	Directory for testing llvmparse
*.c *.ll	ANSI/ISO C source for a test case. LLVM textual assembly files generated by clang-3.2 (not included).

3. llvmparse setup

For using llvmparse, this distribution comes with ready-made native executables for Windows and Linux, which can be found in the corresponding `/bin/exedir` directory, where `exedir` is `lin` for Linux or `win` for Windows.

In order to build llvmparse from source on your platform, just run the `build.sh` bash script from the command prompt. Cygwin or MinGW are suggested as POSIX emulation environments on Windows.

First, change to the appropriate directory, for instance:

```
cd /home/user/llvmparse/
```

assuming `llvmparse` was installed inside `/home/user`. A few environmental variables should be set and for this purpose the `env-*.sh` scripts can be used. For instance, a Windows installation requires the following:

```
$ ./env-win.sh
```

The llvmparse distribution can be built from sources by using:

```
$ ./build.sh
```

4. llvmparse usage

The llvmparse parsers can be invoked as shown in the test-all.sh script. However, for exercising a single test case, this can be accomplished using the flex/bison-based parser as follows:

```
$ cd $LLVMPARSE_PATH
$ ./bin/llvmparse.exe fibo.ll >& fibo.txt
```

which parses the test program fibo.ll with llvmparse.exe and stores the parsed program to fibo.txt.

For using the TXL-based parser, it is suggested that the txl interpreter is visible. Usually txl is installed in the /usr/local/bin path in Linux and Cygwin/Windows systems. For MinGW/Windows systems, it is suggested that the Windows port of TXL is used. For instance, TXL 10.6 can be installed to /c/Txl106/.

The env-*.sh scripts can be used to set the appropriate value for the TXL_PATH environment variable. An example invocation of the TXL-based parser is as follows:

```
$ $TXL_PATH/bin/txl -q -raw -o $fibo.txt
$LLVMPARSE_PATH/src/llvmparse.Txl fibo.ll
```

5. Running the test suite

The test suite that accompanies llvmparse can be found in the /test subdirectory. It consists of C source code and the generated LLVM textual assembly files, which use the .ll suffix. These LLVM IR files can be exercised by running the top-level test-all.sh script:

```
$ cd $LLVMPARSE_PATH
$ ./test-all.sh
```

The generated debris files (using the *.bisonparsed and *.txlparsed patterns) can be deleted using the corresponding clean-all.sh script:

```
$ cd $LLVMPARSE_PATH
$ ./clean-all.sh
```

6. Prerequisities

- Standard UNIX-based tools (tested on MinGW/Windows XP, Fedora 8 (Linux) and Ubuntu 11.10 (Linux).

  - make
  - bash (shell)

  For rebuilding on Windows, the MinGW (http://www.mingw.org) is a good solution, since it provides a lightweight, yet working, POSIX environment.

- TXL interpreter from http://www.txl.ca

  The latest TXL distribution (10.6) for either Linux or Windows is needed.