Санкт-Петербургский Государственный Университет
Институт Информационных Технологий

Система управления процессом оценки качества компиляторов

версия 1.0 Лукин Е.

Санкт-Петербург
2002

Содержание

1 Введение

Данная система предназначена для оценки качества компиляторов и быстрого создания пригодных для печати отчётов. Оценка качества компиляторов производится путём вычисления времени работы тестовых приложений с помощью комплекса `SPEC INT2000`[1]. На основе поизведённых замеров вычисляются метрики описанные в [1]. Значения этих метрик и являются показателями качества компилятора.

Система, представляющая собой набор скриптов на `Perl` и `bash`, а также небольшого числа бинарных исполняемых модулей, является надстройкой над `SPEC INT2000` и реализует следующую функциональность:

- тестирование произвольного набора компиляторов
- использование ранее полученных результатов
- генерацию отчётов в форматах TeX, `PostScript`

1.1 Требования и ограничения

- ОС `UNIX`
- `SPEC INT2000 CPU2000` для `UNIX`
- `BASH`
- `Perl`
- TeX (LaTeX, `dvips`)
- компилятор `cc` совместимый с `gcc`

1.2 Структура системы

Дерево каталогов системы:

```
SpecReport/
   ConfigGen/
      compilers.<os>
      template.<os>
   GenGraph/
      GenGraph.pl
      template.tex
   Metrics/
      Metrics.pl
   oops-spec/
      c-src/
      run-oops-spec.sh
      make-cfg.pl
      check-binaries.sh
      check-time.*
   texdraw/
   run-oops
```

[1] `http://www.spec.org`

Здесь

— `run-oops` — основной скрипт. Обсуждается в части 3.

— `ConfigGen` — генератор конфигурационных файлов `SPEC INT2000` по набору компиляторов.

— `compilers.<os>` — описание набора компиляторов для ОС <os> [2].

— `template.<os>` — шаблон конфигурационного файла для ОС <os>.

— `GenGrapg.pl` — генератор отчётов в TeX и `PostScript`.

— `template.tex` — шаблон отчёта.

— `Metrics.pl` — модуль вычисления метрик по результатам замеров.

— `c-src/` — исходные файлы для бинарных модулей.

— `run-oops-spec.sh` — модуль запуска `SPEC INT2000` и генерации отчётов в фоновом режиме.

— `make-cfg.pl`

— `check-binaries.sh`

— `check-time.sh`

— `texdraw/` — пакет графики для TeX.

1.3 Алгоритм работы

Система запускается с помошью скрипта `run-oops`. После разбора параметров (см. часть 3.1), их проверки и оценки совместимости происходят следующие действия:

1. Проверка существования бинарных компонентов системы, и их компиляция в случае необходимости[3]. Для этого используется скрипт

   ```
   oops-spec/check-binaries.sh
   ```

2. Создание и сохранение конфигурационного файла системы на основе уже имеющегося и параметров, передаваемых через коммандную строку. Этот файл предназначен для конфигурации повторного запуска системы.

   Шаги 3—4 исполняются только при создании новых или переопределении старых результатов замеров:

[2] <os> — значение системной переменной `OSTYPE`.

[3] Данный подход избран из соображений переносимости системы. Так, например, широко используемая утилита `mktemp` отсутствует в стандартной поставке ОС Solaris. В то же время, хранить скомпилированный бинарный код для всех платформ представляется недопустимым.

3. Генерация конфигурационных файлов SPEC INT2000 для каждого компилятора из тестируемого набора. Каждый файл создаётся на основе шаблона для данной платформы и набора параметров для конкретного компилятора.
4. Конфигурирование SPEC INT2000 .
5. Запуск run-oops-spec.sh в фоновом режиме. Это позволяет выйти из системы без прекращения тестирования.
6. Запуск check-time.sh также в фоновом режиме. Этот скрипт отслеживает время исполнения основного процесса и прерывает его если время исполнения превышает заранее установленное (см. часть 3.1).

2 Настройка системы

2.1 Начальная настройка

Начальная настройка состоит из следующих шагов:

1. Добавить корневой каталог системы в PATH:

```
export PATH=<some_dir>/SpecReport/:$PATH
```

2. Добавить модули texdraw в TEXINPUTS:

```
export TEXINPUTS=<some_dir>/SpecReport/texdraw:$TEXINPUTS
```

2.2 Регистрация компиляторов

Прежде чем тестировать компилятор в системе необходимо зарегестрировать его. При этом указываются сам компилятор, редактор связей (может отличаться от первого), а также параметры для обоих.

Регистрация производится вручную путём редактирования файла

```
SpecReport/ConfigGen/compilers.<os>
```

В приложении A.1 описан его формат.

3 Запуск и генерация отчётов

Система запускается из коммандной строки следующего формата:

```
run-oops [options] [config]
```

[options] определены в части 3.1, [config] — имя необязательного конфигурационного файла, имеющего формат <param>=<value>. <param> также определены в части 3.1.

При запуске параметры считываются из конфигурационного файла (если указаны) и перекрываются соответствующими параметрами командной строки. Полученная таким образом конфигурация сохраняется в каталоге сессии для повторного использования.

Часть параметров является обязательной (см. часть 3.1), они всегда должны присутствовать в конфигурации, либо в командной строке, либо в конфигурационном файле.

3.1 Параметры запуска

- Список тестовых приложений

  Командная строка: `-b, -benchmarks '<bench list>'`

  Конфигурационный файл: `BENCHMARKS='<bench list>'`

  `<bench list>` — список тестовых приложений SPEC INT2000 , которые используются при вычислении *Selected Performance* [1].

  В ходе тестирования этот параметр обычно применяется при повторных запусках. После первого тестирования производится предварительный анализ результатов на основе которого и составляется этот список.

  Названия тестовых приложений разделяются пробелами.

  При отсутствии этого параметра *Selected Performance* не вычисляется.

  Класс: Необязательный

- Список компиляторов

  Командная строка: `-c, -compilers '<compiler list>'`

  Конфигурационный файл: `COMPILERS='<compiler list>'`

  Список названий компиляторов, которые участвуют в тестировании. В качестве названий используются заголовки секций в списке компиляторов (см. часть A.1), дополненные, при необходимости новыми параметрами. Дополнительные параметры присоединяются к списку указанному в `C_FLAGS`. Так, указание `gcc -O3` (см. приложение B.1) в списке компиляторов приведёт к тестированию `/usr/local/bin/gcc -I /usr/lib/ -O3`. Компиляторы в списке разделяются запятыми; дополнительные параметры отделяются от имени пробелами.

  Класс: **Обязательный**

- Временное ограничение

  Командная строка: `-t, -timeout <hh:mm>`

  Конфигурационный файл: `TIMEOUT=<hh:mm>`

  `<hh:mm>` — время в часах и минутах, по истечении которого с начала тестирования процесс принудительно завершается. При отсутствии этого параметра временные рамки не определяются.

  Класс: Необязательный

- Эталонный компилятор

Командная строка: `-r, -refc <compiler name>`

Конфигурационный файл: `REF_COMP=<compiler name>`

`<compiler name>` — название эталонного компилятора.

Эталонный компилятор, как и все остальные, должен быть зарегестрирован (см. приложение A.1), а кроме того должен входить в *список компиляторов*.

Используется для вычисления *Overall Performance* [1].

Если не указан, *Overall Performance* не вычисляется.

Класс: Необязательный

- Название сессии

  Командная строка: `-e, -testing <testing name>`

  `<testing name>` — название каталога, в котором будут храниться результаты тестирования. Если этот параметр не указан, то название будет автоматически сгенерено исходя их названия конфигурационного файла и текущего времени.

  Класс: Необязательный

- Повторное использование результатов

  Командная строка: `-u, -reuse`

  Приводит к повторному использованию результатов тестирования. Здесь к результатам относятся как результаты замеров `SPEC INT2000` , так и сохранённая конфигурация предыдущего запуска.

  В режиме повторного использования не производится запуск `SPEC INT2000` , а лишь используются результаты замеров для пересчёта метрик и построения отчётов.

  В случае, если не задан конфигурационный файл, используется конфигурация предыдущего запуска, которая сохраняется в каталоге данной сессии. При этом игнорируются изменения в списке компиляторов.

  Класс: Необязательный

- Перекрытие результатов

  Командная строка: `-o, -override`

  В режиме перекрытия результатов уничтожаются все ранее вычисленные замеры, отчеты и пр. Происходит запуск `SPEC INT2000` с дальнейшим вычислением метрик и созданием отчётов.

  Класс: Необязательный

- Режим эмуляции SPEC INT2000

  Командная строка: `-d, -demo`

  Режим эмуляции (предназначен для отладки).

  Класс: Необязательный

A Форматы данных

A.1 Список компиляторов

Список компиляторов[4] состоит из секций следующего вида:

```
[<name>]
COMPILER=<compiler>
C_FLAGS=<cparams>
LINKER=<linker>
LFLAGS=<lparams>
```

Здесь

— `<name>` — имя компилятора, которое используется при при запуске системы (см. часть 3.1)

— `<compiler>` — полное имя компилятора. Если указан в путях, то достаточно короткого имени

— `<cparams>` — основные параметры компилятора. См. часть 3.1 о дополнительных параметрах

— `<linker>` — полное имя редактора связей. Если указан в путях, то достаточно короткого имени

— `<lparams>` — параметры радактора связей

A.2 Графические настройки GenGraph

Нестандартные параметры `GenGraph` указываются в отдельном файле, который передаётся скрипту с помощью параметра командной строки `-c`. Параметры задаются в виде `<name>=<value>`. Строки, не удовлетворяющие этому правилу, а также содержащие непредусмотренные значения `<name>`, игнорируются.

Список переменных (в скобках указано значение по умолчанию):

— `hsize (=10.0)` — горизонтальный размер графика в текущих еденицах измерения [5]

[4] Правильнее было бы говорить о **конфигурациях компиляторов**, так как данный формат позволяет задавать один и тот же компилятор под разными именами, для возможности варьирования параметров.

[5] Задаются в шаблоне графика `template.tex`

— `vsize (=10.0)`— вертикальный размер графика в текущих еденицах измерения

— `rindent (=1.0)`,`lindent (=1.0)` - правый и левый отступ от краёв графика

— `colgap (=1.0)` — расстояние между столбцами

— `vscale (=0.9)` — отношение между наибольшей высотой столбца и общей высотой графика

— `ticklev (=2)` — количество делений второго уровня на оси ординат [6]

— `axisfile (="drawaxis.tex")` — имя файла сопределением осей координат

— `columnfile (="drawcolumns.tex")` — имя файла с определением столбцов графика

— `template (="template.tex")` — имя файла шаблона графика

— `tex (="latex")` — имя процессора TEX

— `dvips (="dvips")` — имя программы преобразователя TEX`-> PostScript`

B Пример входных данных

B.1 Список компиляторов

```
[gcc]
COMPILER=/usr/local/bin/gcc
C_FLAGS=-I /usr/lib/
LINKER=/usr/local/bin/gcc
LFLAGS=

[lcc]
COMPILER=/usr/local/bin/lcc
C_FLAGS=
LINKER=/usr/local/bin/gcc
LFLAGS=

[vpo]
COMPILER=/usr/local/zephir/vpo
C_FLAGS=
LINKER=/usr/local/bin/gcc
LFLAGS=
```

[6] Количество делений первого уровня определяется по наибольшей высоте столбца.

Список литературы

[1] D. Boulytchev, D. Lomov. *An Empirical Study of Retargetable Compilers.* 4th International Andrei Ershov Memorial Conference on Perspectives of System Informatics, Akademgorodok, Novosibirsk, Russia, July 2-6, 2001. LNCS 2244, p. 328