Profiling Cmajor Applications

Seppo Laakko

October 19, 2015

1 Profiling in Windows

1.1 Creating a profile report

The steps for creating a profile report for the *Downloads* example application are as follows.

(This example profiles the LLVM backend but the steps are the same for profiling the C backend.)

- If you haven't build the system library using profile configuration in installation time, now is the time to do it:
  - Open Cmajor Development Environment.
  - Open Built-in Projects | System Library project.
  - Select profile configuration from Configuration combo box.
  - Select Build | Build Solution. This builds the projects in the System solution and instruments the functions in its projects with calls to collect timing information.
- Open Built-in Projects | Examples project.
- Select profile configuration from Configuration combo box.
- Right click the Downloads project and choose Build. This builds the Downloads project and instruments the functions in it with calls to collect timing information. Also a **downloads.cmprof** file is created that contains a unique function identifier for each function in the program.
- Open command prompt and change to %APPDATA%\Cmajor\examples\downloads\profile\llvm directory:

  ```
  cd %APPDATA%\Cmajor\examples\downloads\profile\llvm
  ```

- Run downloads program:

  ```
  downloads
  ```

  This creates a binary file **downloads.profdata** that contains the timing information.

- Run **cmprof downloads.cmprof**:

  ```
  cmprof downloads.cmprof
  ```

  This scans the profile data and creates a profile report (**downloads.cmprofreport**).

1.2 Inspecting the profile report

Run profile report viewer **cmprofview** by issuing command:

```
cmprofview downloads.cmprofreport
```

Alternatively you can double click the **downloads.cmprofreport** file in the file explorer to open it in the profile report viewer.

The columns in the report are as follows:

- Function: Full name of the function.
- Called: The number of times the function is called.
- Elapsed Inclusive ms: Time spent in executing this function and the functions it calls in milliseconds.
- Elapsed Inclusive %: Percentage of time spent is executing this function and the functions it calls compared to total execution time.
- Elapsed Exclusive ms: Time spent in executing this function excluding the time spent in executing the functions it calls in milliseconds.
- Elapsed Exclusive %: Percentage of time spent is executing this function excluding the time spent in executing the functions it calls compared to total execution time.

You can click the column headers in the report to sort the rows in ascending or descending order based on the clicked column.

2 Profiling in Linux

2.1 Creating a profile report

The steps for creating a profile report for the *Downloads* example application are as follows.

(This example profiles the LLVM backend but the steps are the same for profiling the C backend.)

- If you haven't build the system library using profile configuration in installation time, now is the time to do it:
  - Open a terminal window and change to `<major>/system` directory:

    ```
    seppo@raid:~$ cd Programming/cmajor-1.2.0/system
    ```

  - Build system library using profile configuration:

    ```
    seppo@raid:~/Programming/cmajor-1.2.0/system$
    cmc -config=profile system.cms
    ```

- Change to `<major>/examples/downloads` directory:

  ```
  seppo@raid:~/Programming/cmajor-1.2.0/system$
  cd ../examples/downloads
  ```

- Build the downloads project using profile configuration:

  ```
  seppo@raid:~/Programming/cmajor-1.2.0/examples/downloads$
  cmc -config=profile downloads.cmp
  ```

- Change to the profile/llvm subdirectory:

  ```
  seppo@raid:~/Programming/cmajor-1.2.0/examples/downloads$
  cd profile/llvm
  ```

- Run downloads program:

  ```
  seppo@raid:~/Programming/cmajor-1.2.0/examples/downloads/profile/llvm$
  ./downloads
  ```

  This creates a binary file **downloads.profdata** that contains the timing information.

- Run **cmprof downloads.cmprof**:

  ```
  seppo@raid:~/Programming/cmajor-1.2.0/examples/downloads/profile/llvm$
  cmprof downloads.cmprof
  ```

  This scans the profile data and creates a profile report (**downloads.cmprofreport**) and three additional text files: **downloads.count.txt**, **downloads.inclusive.txt** and **downloads.exclusive.txt**.

2.2 Inspecting profile report files

The **downloads.cmprofreport** file contains data in random order. The **downloads.count.txt** file contains the data sorted by call count. The **downloads.inclusive.txt** file contains the data sorted by elapsed inclusive milliseconds. The **downloads.exclusive.txt** file contains the data sorted by elapsed exclusive milliseconds.

The columns in the report are as follows:

- Function identifier. Unique integer of the function.
- Function: Full name of the function.
- Called: The number of times the function is called.
- Elapsed Inclusive ms: Time spent in executing this function and the functions it calls in milliseconds.
- Elapsed Inclusive %: Percentage of time spent is executing this function and the functions it calls compared to total execution time.
- Elapsed Exclusive ms: Time spent in executing this function excluding the time spent in executing the functions it calls in milliseconds.
- Elapsed Exclusive %: Percentage of time spent is executing this function excluding the time spent in executing the functions it calls compared to total execution time.