

MPE (Message Passing Environment)

przygotował: Krzysztof Jurczuk

Zaawansowane funkcje biblioteki MPE umożliwiają tworzenie logów wykonania programów, które później mogą być bardzo łatwo zwizualizowane, np. programami:

- jumpshot (tryb graficzny)
- states (semaryczne wydruki czasów)

1. Generowanie logów

Procedura użycia na przykładzie prostego kodu (zakładana jest podstawowa znajomości MPI):

```
int main( int argc, char *argv[])
{
    int size, rank;
    MPI_Init(&argc,&argv);
    MPI_Comm_rank( MPI_COMM_WORLD, &rank );

    MPE_Init_log();           //inicjalizacja biblioteki MPE
    MPE_Start_log();         //rozpoczecia logowania

    MPE_Finish_log("cpilog"); //zakończenie logowania MPE

    MPI_Finalize();          //zakończenie MPI

    //nadawanie unikalnych numerow dla punktow w programie
    eventCPU0_start = MPE_Log_get_event_number();
    eventCPU0_koniec = MPE_Log_get_event_number();
    eventCPU1_start = MPE_Log_get_event_number();
    eventCPU1_koniec = MPE_Log_get_event_number();

    //specyfikacja zdarzen
    //przedostatni parametr określa nazwę zdarzenia, ostatni natomiast graficzną prezentację
    //zdarzenia (kolor:stylMaptyBitowej), styl mapy bitowej jest używany, gdy pracujemy w
    //środowisku monochromatycznym
    MPE_Describe_state( eventCPU0_start, eventCPU0_start, "nazwa zdarzenia1","green" );
    MPE_Describe_state( eventCPU1_start, eventCPU1_koniec, "nazwa zdarzenia2","red:vlines3" );

    if( rank == 0 ){
        MPE_Log_event( eventCPU0_start, 0, NULL );
        ProcesyPreInteracji();
        MPE_Log_event( eventCPU0_koniec, 0, NULL );
    }
    else{
        MPE_Log_event( eventCPU1_start, 0, NULL );
        ProcesIteracji();
        MPE_Log_event( eventCPU1_koniec, 0, NULL );
    }

    //zakończenie logowania i zrzut logów do pliku podane w parametrze
    MPE_Finish_log("cpilog");
    MPI_Finalize();
}
```

Uwaga:

Samodzielne nadawanie numerów dla zdarzeń nie jest, zgodnie z dokumentacją, polecane. Jednak doświadczenia pokazały, że czasami trudno było uniknąć takiej sytuacji. Należy jednak wtedy uważać, aby nie trafić w numery przyporządkowane, np. dla funkcji MPI_Send, MPI_Recv itd.

Kompilacja:

Zrówny przy kompilacji jak i przy linkowaniu dodajemy opcje: -mpilog.

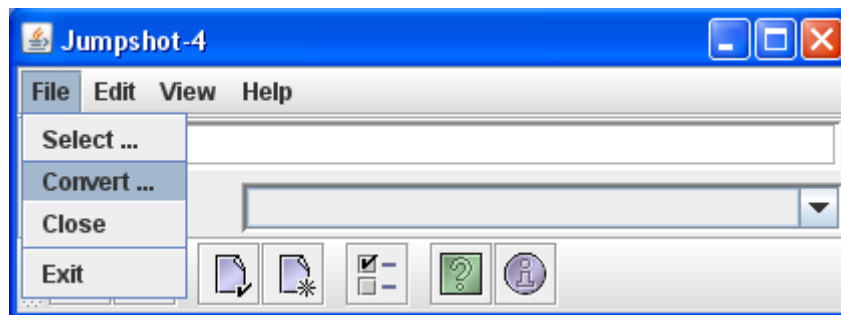
2. Wizualizacja wygenerowanych logów za pomocą JumpShot

Aktualnie zainstalowana implementacja MPI, czyli MVAPICH 0.9.5, na serwerach mordor i mordor2 dostosowane do używanego InfiniBand zawiera biblioteki MPE, które generują pliki CLOG. W celu użycia ich przez najnowszego Jumpshot'a należy je przekonwerować do formatu SLOG2.

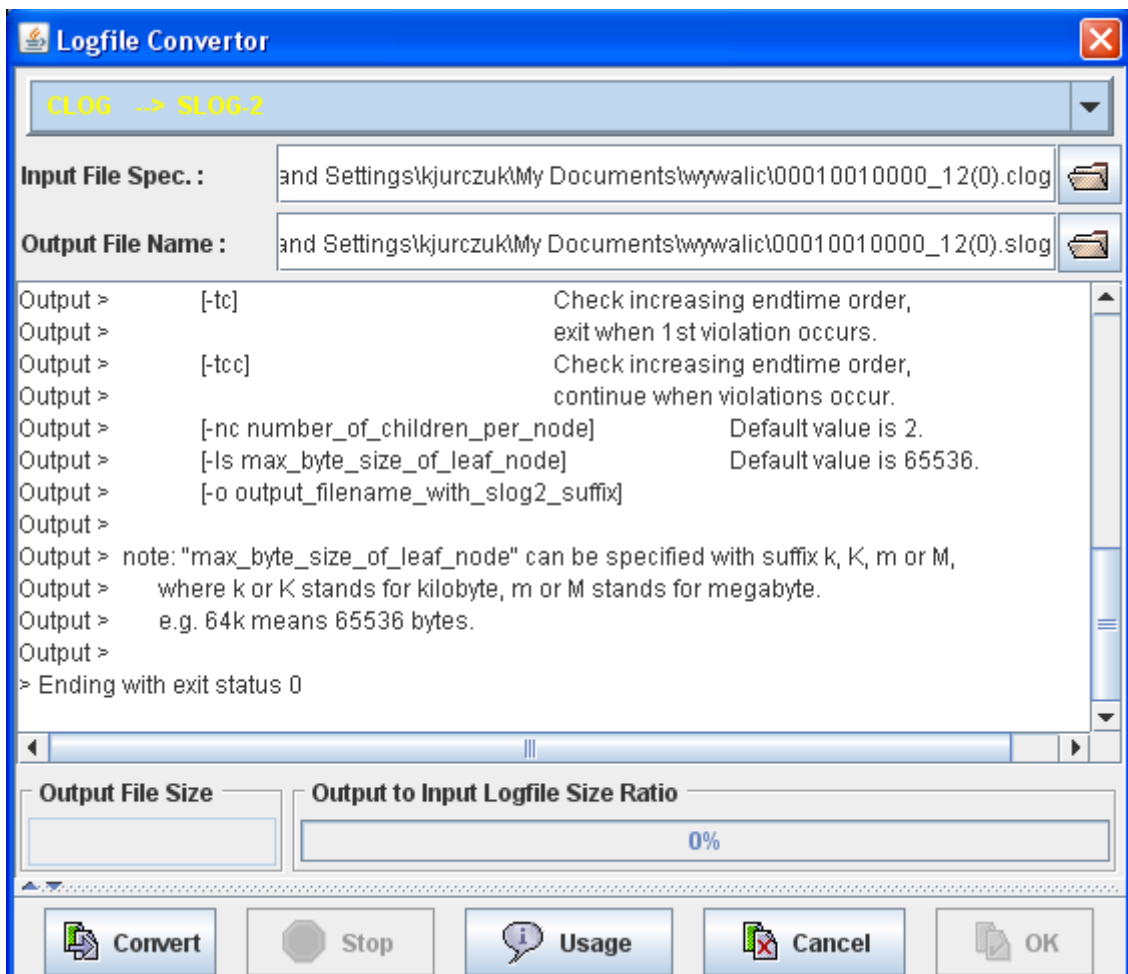
Uruchamiamy jumpshot'a:

```
C:\ Command Prompt - java -jar jumpshot.jar
D:\Documents and Settings\kjurczuk>cd F:\praca naukowa\publikacje\2008_ZeszytyNaukowe_LoadBalancing\dane z profilowania\oprogramowanie\slog2rte\slog2rte-1.2.6\lib
D:\Documents and Settings\kjurczuk>f:
F:\praca naukowa\publikacje\2008_ZeszytyNaukowe_LoadBalancing\dane z profilowania\oprogramowanie\slog2rte\slog2rte-1.2.6\lib>java -jar jumpshot.jar
Java is version 1.6.0_05.
Starting the SLOG-2 Display Program . . . . .
Jumpshot-4 setup file : D:\Documents and Settings\kjurczuk\.jumpshot4.conf
Initialize Parameters:
Y_AXIS_ROOT_LABEL = SLOG-2
```

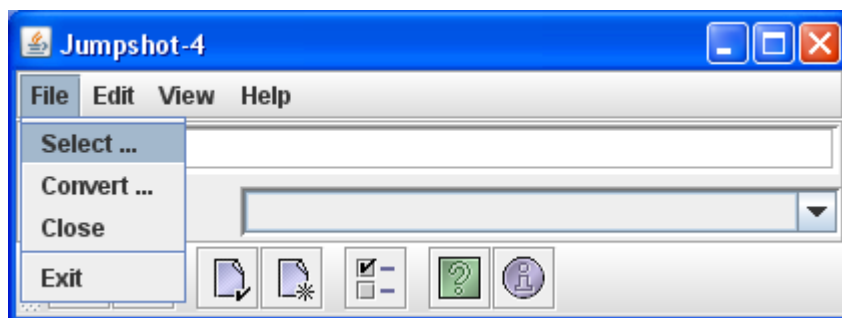
Wybieramy okno konwertera:



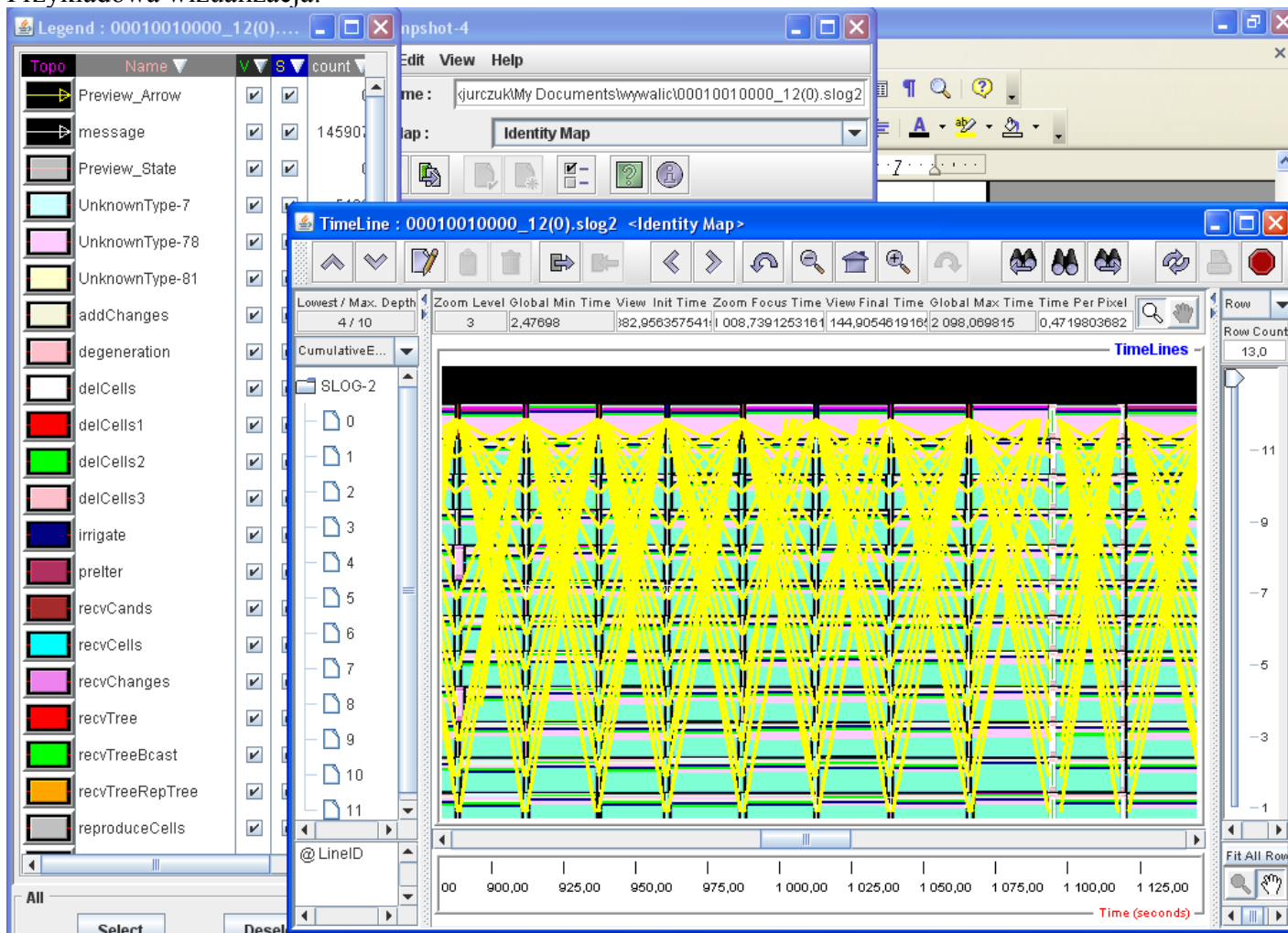
Wybieramy rodzaj konwersji clog -> slog2, wskazujemy plik z logami formatu CLOG oraz podajemy ścieżkę do pliku wynikowego.



Po udanej konwersji możemy od razu przejść do graficznej prezentacji logów poprzez wciśnięcie OK. Konwersje wykujemy raz, później wykorzystujemy już zapisany plik w formacie SLOG2.



Przykładowa wizualizacja:



Rady:

- w początkowej fazie wyłączamy Preview_Arrow i message (strzałki pokazujące kierunku wysyłania wiadomości MPI)
- odznaczmy zdarzenia UnkownType (sa to zdarzenia funkcji MPI_Send, MPI_Recv itd), które są tak identyfikowane podczas wyspecyfikowania własnych zdarzeń
- w przypadku braku specyfikacji własnych zdarzeń w pliku z kodem źródłowym przy wizualizacji sa zaprezentowane konkretne zdarzenia MPI_Send, MP_Recv itd

Zródła:

1. <http://www-unix.mcs.anl.gov/perfvis/>
2. <http://www.mcs.anl.gov/mpi/mpich1/docs/mpeman.pdf>
3. <ftp://ftp.mcs.anl.gov/pub/mpi/slog2/js4-usersguide.pdf>

Download:

1. LOG-2 Software Development Kit (slog2sdk) which provides both the Software Development and Runtime environments for [SLOG-2](#) and [Jumpshot-4](#).
Unix and Cygwin <ftp://ftp.mcs.anl.gov/pub/mpi/slog2/slog2sdk.tar.gz>
Windows <ftp://ftp.mcs.anl.gov/pub/mpi/slog2/slog2rte.tar.gz>
2. MPE
Unix and Cygwin <ftp://ftp.mcs.anl.gov/pub/mpi/mpe/mpe2.tar.gz>