

Solaris 3分 クッキング

Cook Book

DTrace

Chime Visualization Tool 編

Solaris 3分クッキング:Chime を使って DTrace 出力データのビジュアル化

OpenSolaris のプロジェクト “Chime” (チャイム) を使って、DTrace の出力データをビジュアル化するデモを紹介致します。DTrace では、解析結果を集計で表示する Aggregation (集積関数) が用意されていますが、現在は CLI (コマンドラインインターフェース) でのみ提供されています。Chime は DTrace の Java API を実装して、Aggregation をグラフィカルに表示するためのツールです。今回のナイトセミナーでは、画面で簡単な使用方法のデモだけ行います、Chime のインストールやデモで使用した設定方法はこちらのレシピを参考にしてください。

デモの材料:

パソコン	TOSHIBA dynabook TX550LS, Intel (R) Celeron(R) M 1.40GHz, Memory: 1GB HDD:100GB
OS	Solaris Express Developer Edition (SXDE) 9/07

紹介するデモの内容

- ① Chime Visualization Tool に組み込み済みのデモを紹介致します。
- ② 新しい“ディスプレイ”の作成方法を紹介します。

第1章: インストールと起動

Chime の入手方法とインストール方法を紹介します。Chime は OpenSolaris のプロジェクトとして <http://opensolaris.org/os/project/dtrace-chime/> で紹介されています。OpenSolaris の build 35 以上、もしくは Solaris 10 8/07 以降であれば、Chime package をダウンロードしインストールするだけで動作可能となります。

```
solaris-devx> wget http://opensolaris.org/os/project/dtrace-chime/install/osol0chime-i386-1.4.pkg.gz
--15:50:49-- http://opensolaris.org/os/project/dtrace-chime/install/osol0chime-i386-1.4.pkg.gz
=> `osol0chime-i386-1.4.pkg.gz'
Resolving opensolaris.org... 72.5.123.5
Connecting to opensolaris.org[72.5.123.5]:80... connected.
HTTPによる接続要求を送信しました、応答を待っています... 200 OK
長さ: 8,780,051 [application/x-gzip]

100%[=====>] 8,780,051 429.28K/s ETA 00:00

15:51:09 (433.86 KB/s) - `osol0chime-i386-1.4.pkg.gz' を保存しました [8780051/8780051]

solaris-devx> gunzip osol0chime-i386-1.4.pkg.gz
solaris-devx> su
# pkgadd -d ./osol0chime-i386-1.4.pkg
```

インストールが終わったら、Chime を起動しますが、ユーザーモードで起動できるように、デフォルトの privilege (特権) を下記のように変更しておくといでしょう。

/etc/user_attr ファイル内に以下を追加します。<user> は使用する username で置き換えます。

```
<user>:::defaultpriv=basic,dtrace_kernel,dtrace_proc,dtrace_user
```

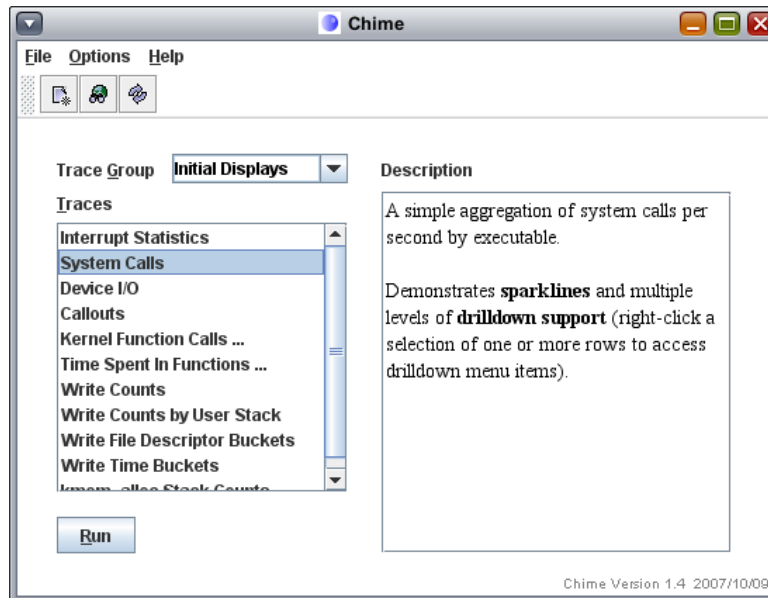
尚、この特権は再ログイン後に反映されます。

Chime の起動:

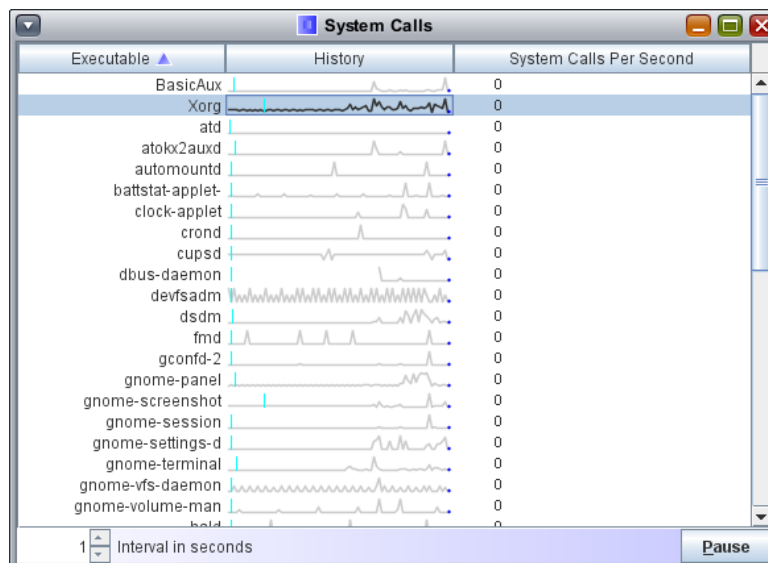
```
solaris-devx> cd /opt/OSOL0chime/bin
solaris-devx> ./chime
```

第2章: Chime デモの実行

Chime を起動すると下の初期画面が現れます。Trace Group “Initial Displays” でいくつかの組み込み済みのデモが実行できるようになっています。Traces メニューから “System Calls” を選択して実行してみましょう。



“System Calls” ラインをクリックして、左下の “Run” ボタンをクリックします。



このデモでは、下記の D-Script (集積関数) が実行されています。

```
syscall:::entry
/ pid != $pid /
{
    @a[execname] = count ();
}
```

表示の1列目に、system call を呼び出したプログラム名、3列目にそのプログラムが一秒間に system call を呼び出した回数を青の棒グラフで表示します。また2列目にその呼び出した回数の履歴がグラフで表示されています。このように DTrace の CLI では、下記のように数値でしか表示できなかった結果をグラフでより解りやすく表示します。

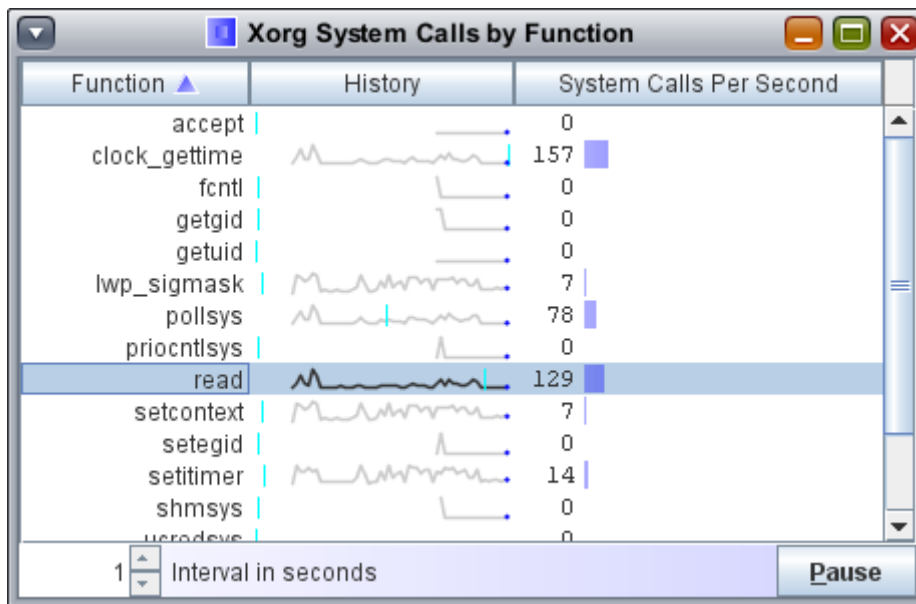
```

solaris-devx> dtrace -n 'syscall:::entry / pid != $pid / { @a[execname] = count() }'
dtrace: description 'syscall:::entry' matched 234 probes
^C

 automountd                1
 devfsadm                  10
 nautilus                  21
 init                      26
 gedit                     27
 dsdm                      32
 sendmail.sendmai         42
 xscreensaver              74
 metacity                  81
 cupsd                     132
 gnome-terminal            141
 mixer_applet2             398
 soffice.bin               745
 java                      2082
 Xorg                      2503
solaris-devx>

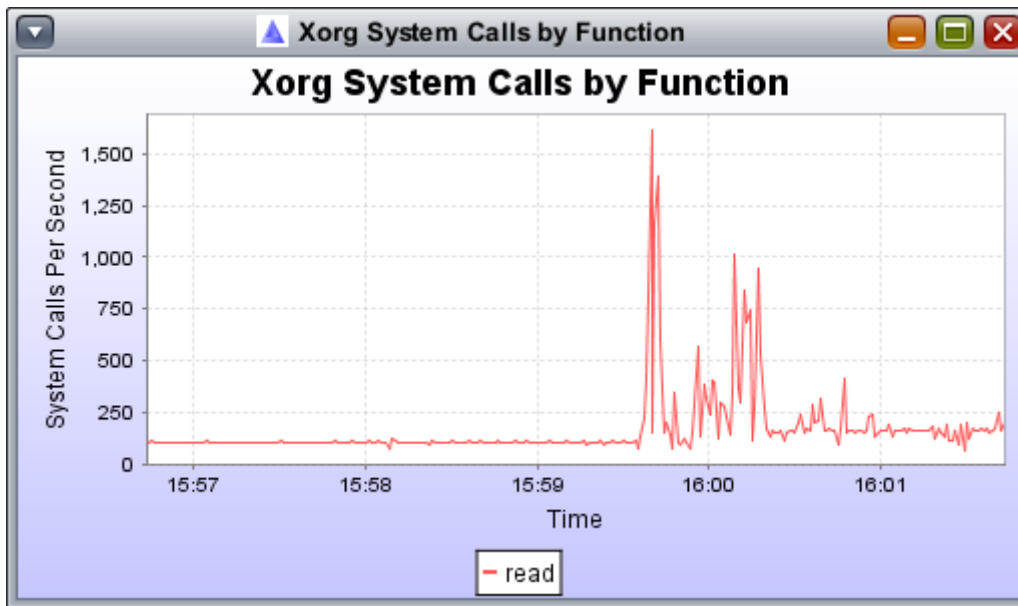
```

さらに、Xorg の行を選択し、マウスの右ボタンをクリック、“Xorg by Function” を選んで Xorg の中を覗いてみましょう。



今度は Xorg 中の各関数での system call の頻度がグラフで表示されます。

次に system call の頻度が高い、read() 関数ラインを選択して、マウスの右ボタンクリック、“Plot Over Time” をメニューから選択して、その詳細を表示します。



この他にも、システムの解析にそのまま使えるようなデモ. サンプルが多数用意されています。

第3章: 新しい Display の作成

自分で作成した D-Script を Chime を使ってグラフ表示させることも可能です。ここでは簡単な例をステップ by ステップで紹介します。

例題:

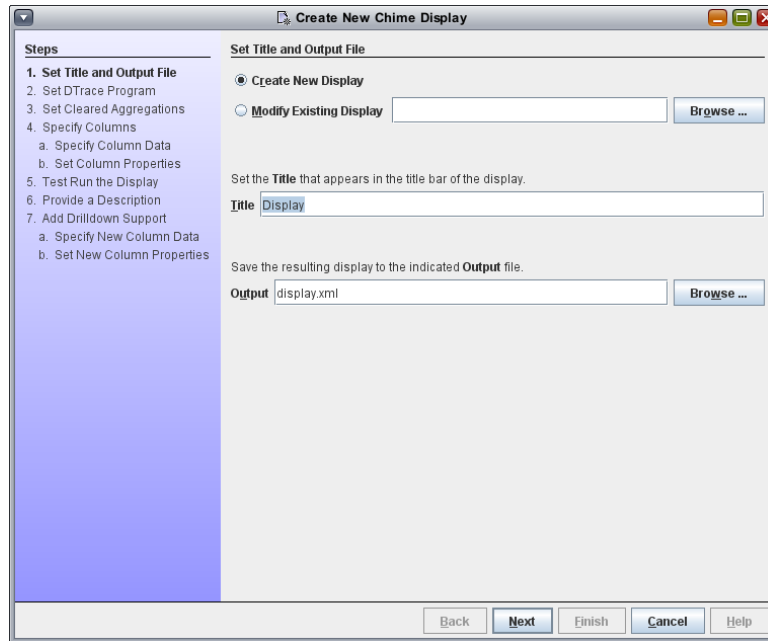
```
sysinfo:::pswitch
{
    @app[execname]=count();
}
```

この例では、sysinfo プロバイダーと、pswitch プローブを使い、CPU のプロセス切り替えカウンターを利用して実行中のプログラムの CPU の使用頻度を集計します。コマンドラインでは下記のようになります。

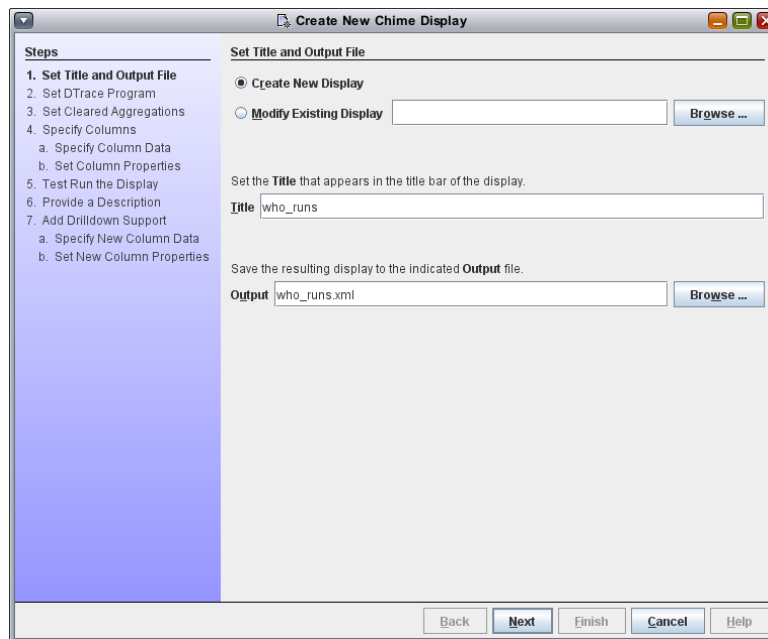
```
solaris-devx> dtrace -n 'sysinfo:::pswitch{ @progs[execname]=count()}'
dtrace: description 'sysinfo:::pswitch' matched 3 probes
^C

BasicAux                                1
... <省略> ...
cupsd                                    26
dtrace                                    33
gnome-terminal                            55
soffice.bin                               72
Xorg                                       73
mixer_applet2                             190
java                                       1126
sched                                     3047
solaris-devx >
```

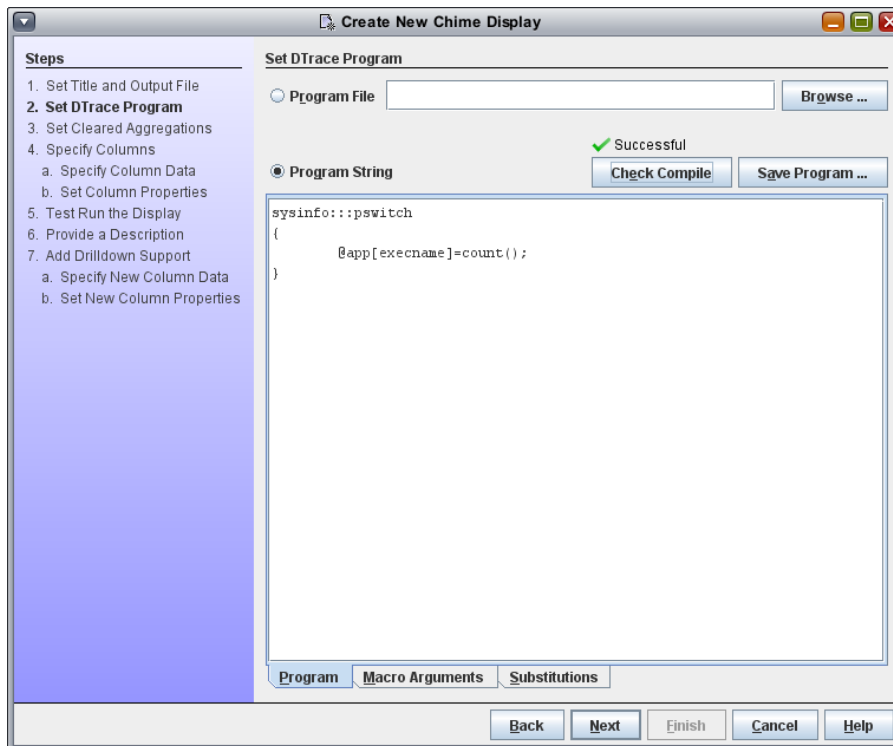
1) Step-1: Chime 初期画面から、“File”メニューから“New Display...”を選択します。



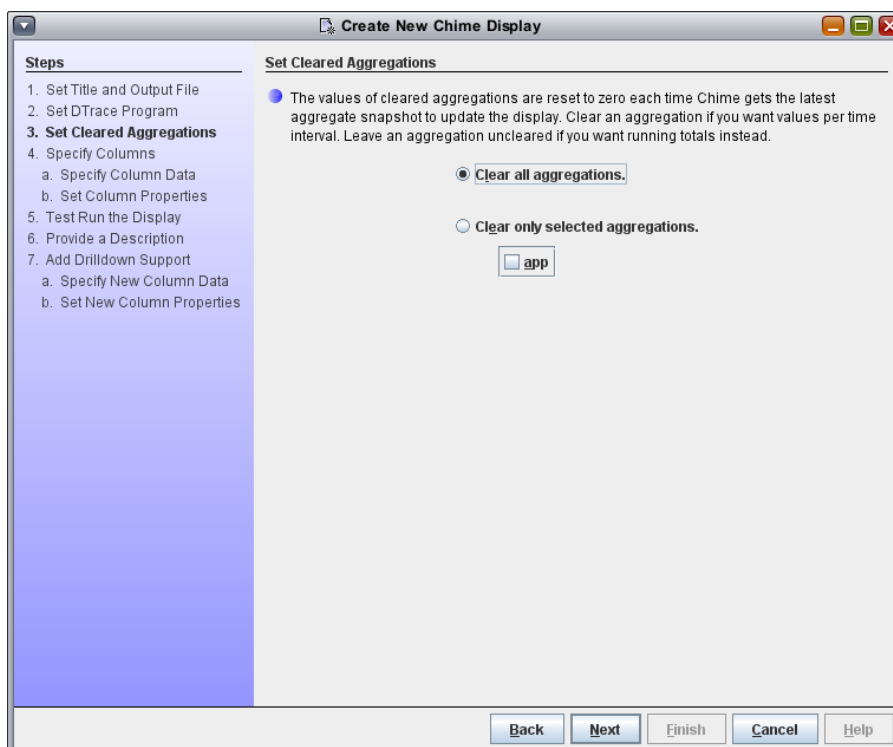
2) Step-2: Title: に 例として “who_runs” を名前をつけて “Next” ボタンで次に進みます。



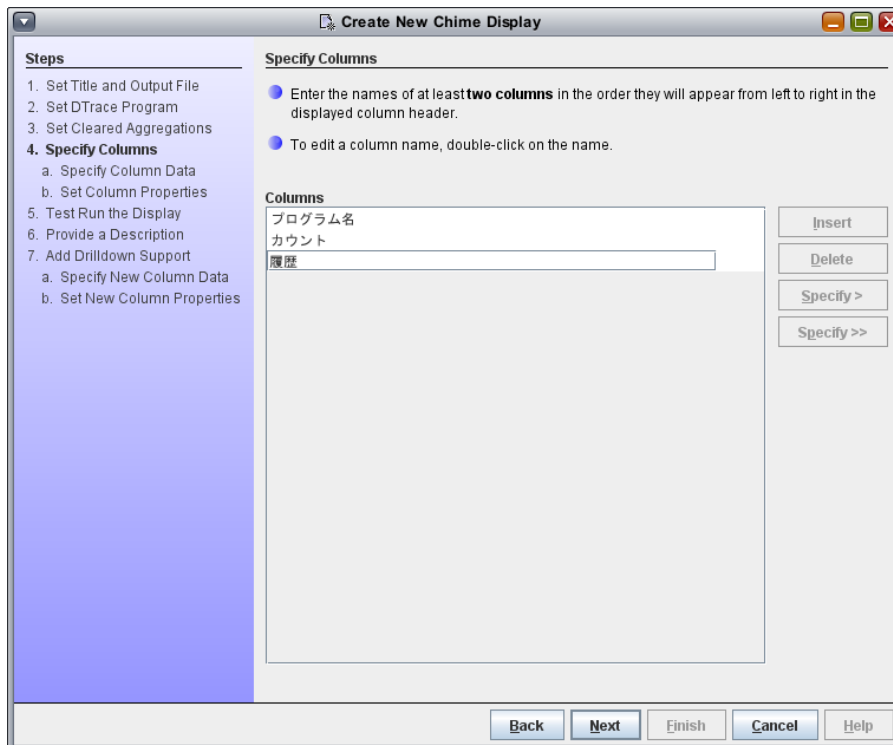
3) Step-3: "Set DTrace Program" 画面で、"Program String" を選択し、テキストボックスに例題の D-script を入力します。入力が終わったら "Check Compile" ボタンを押してコンパイルチェックしましょう。コンパイルが成功すると "Successful" と表示されます。"Next" ボタンで次に進みます。



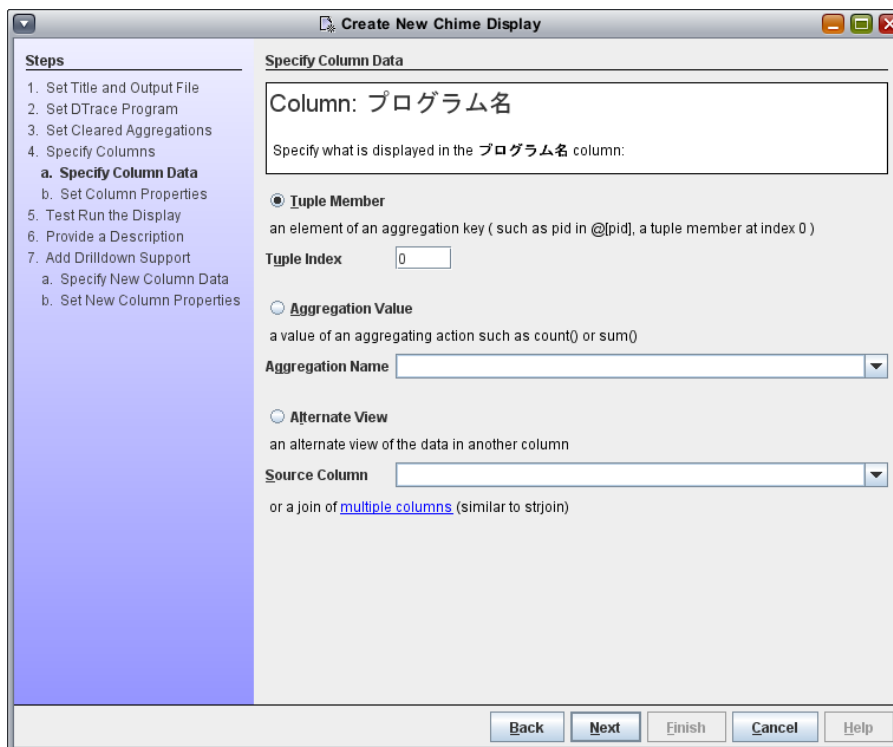
4) Step-4: "Set Cleared Aggregations" 画面では、計測時にクリアする集積関数を指定します。今回の例では時間毎の集計を表示させるので、デフォルトの "Clear All Aggregations" のまま Next ボタンで次に進みます。



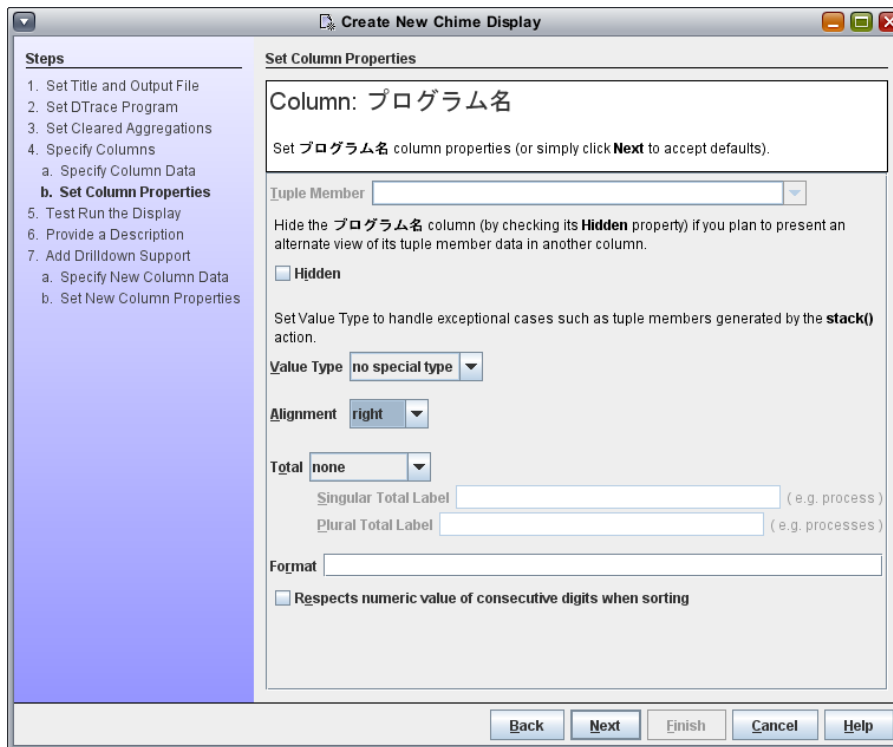
5) Step-5: "Specify Columns" では、表示させる各列のタイトルを "Columns" に入力します。今回の例では、1 列目に "プログラム名"、2 列目に "カウント"、3 列目に "履歴" を表示させるので、この順番で入力します。



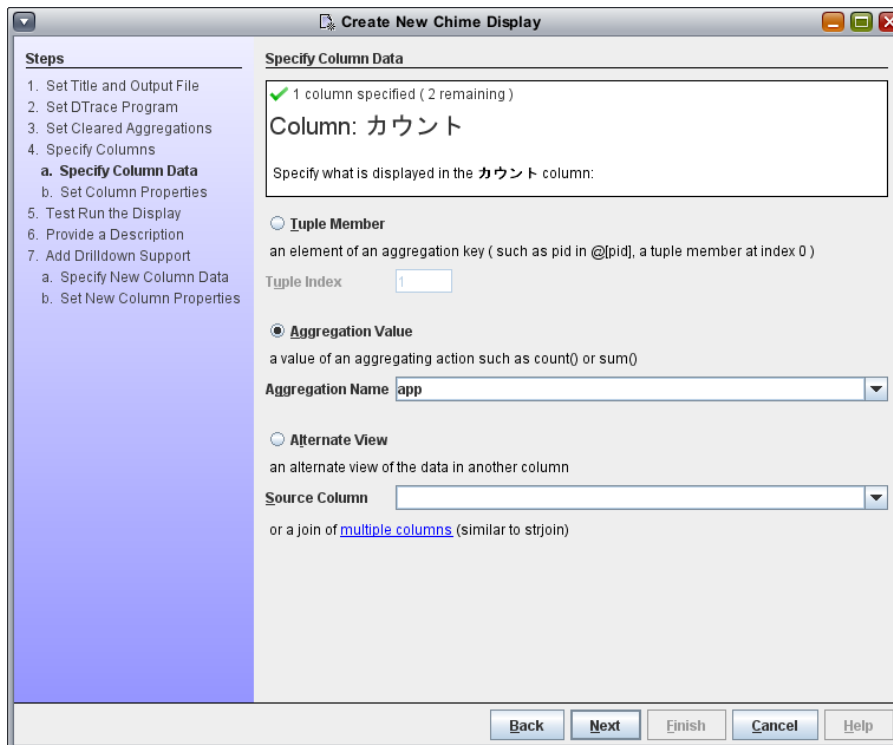
6) Step-6: "Specify Column Data" 画面の最初の "Column: プログラム名" では、"プログラム名" は今回の例の集積のキーとなりますので、"Tuple Member" を選択します。"Next" で次に進みます。



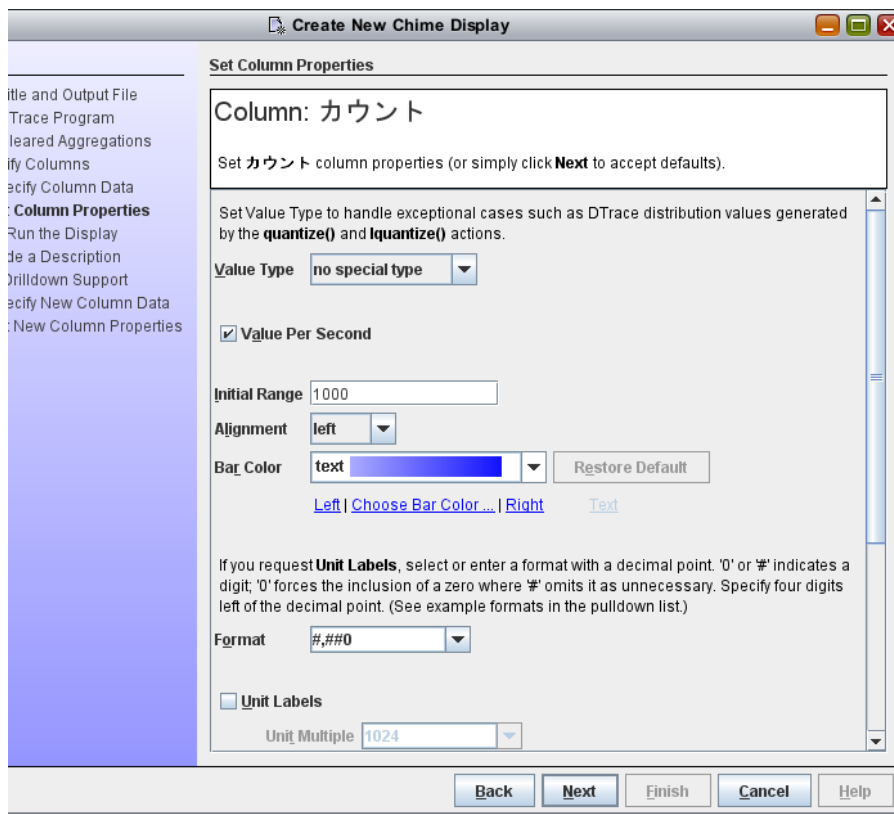
7) Step-7: Column: プログラム名のプロパティを設定します。この例では “Alignment” (配置位置) を “right” (右端) にだけ設定します。Next ボタンで次に進みます。



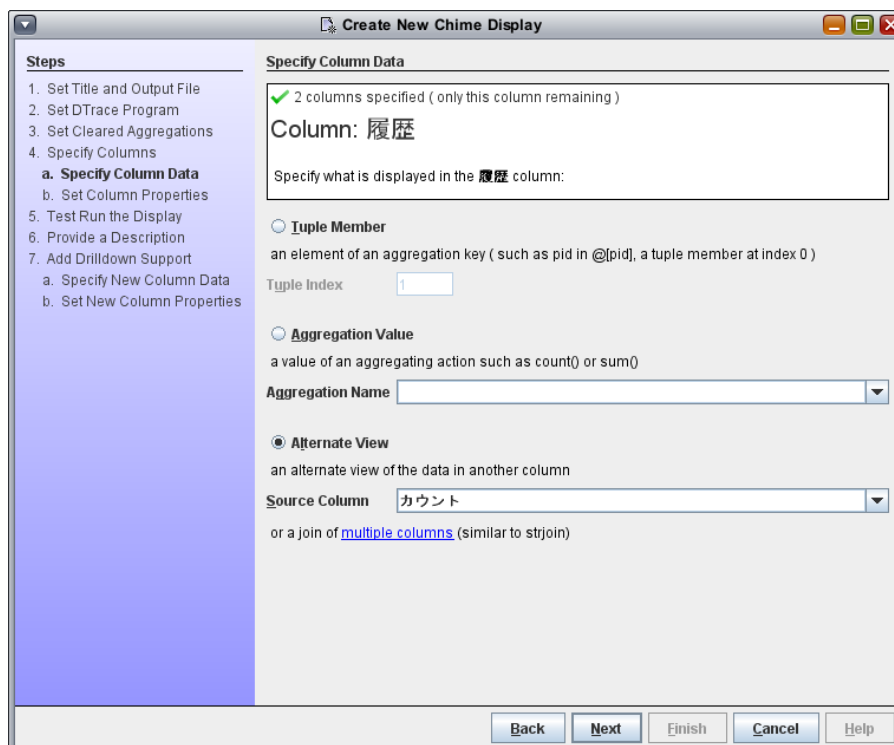
8) Step-8: 2列目の “Column: カウント” を設定します。この例では、集積結果 count() の結果をここで表示させるので、“Aggregation Value” を選択し、D-script で指定した集積関数名 “app” を Aggregation Name に指定します。Next ボタンで次に進みます。



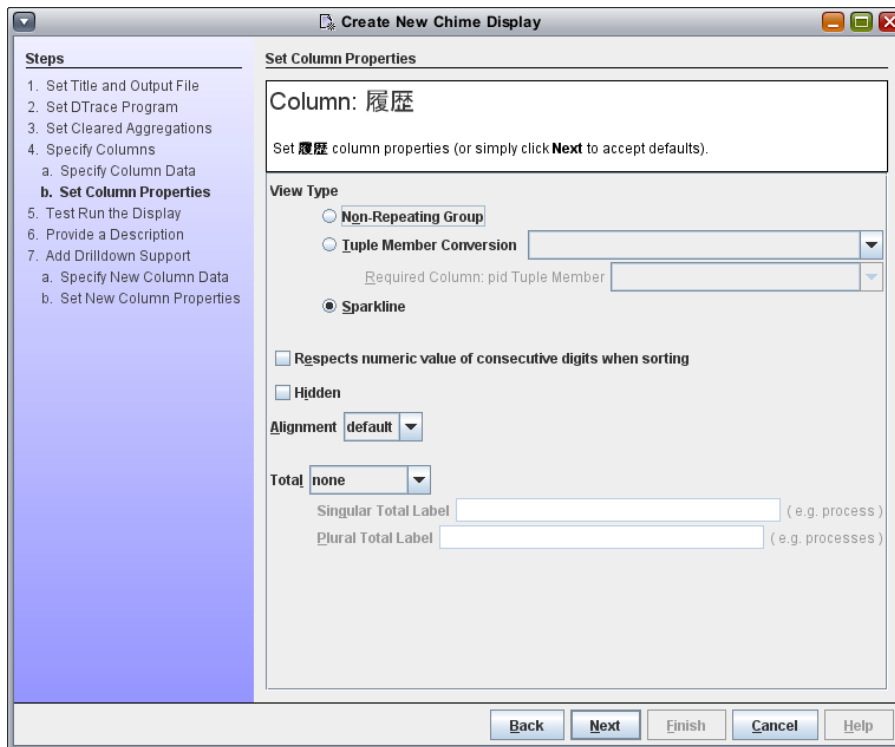
9) Step-9: “Column: カウント” の表示プロパティを設定します。“Value Per Second”, “Alignment:left” また数値の表示フォーマットとして “#,##0” を選択します。Next ボタンで次に進みます。



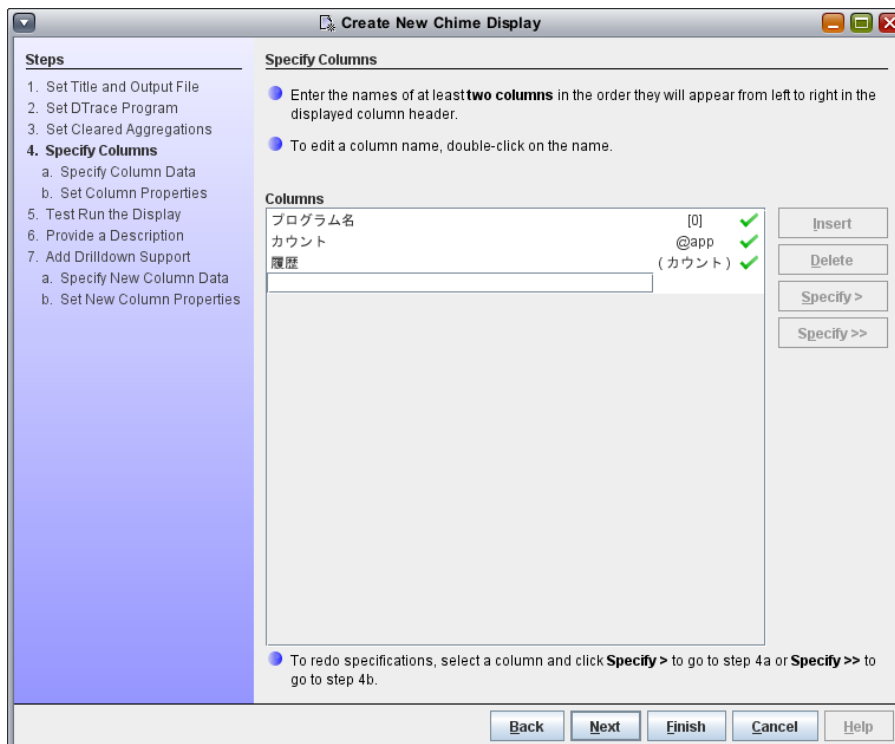
10) Step-10: “Column: 履歴” の設定を行います。“カウント” 表示の “Alternate View”(他の表示方法)として表示させますので、“Alternate View” と、その “Source Column” として “カウント” を選択します。



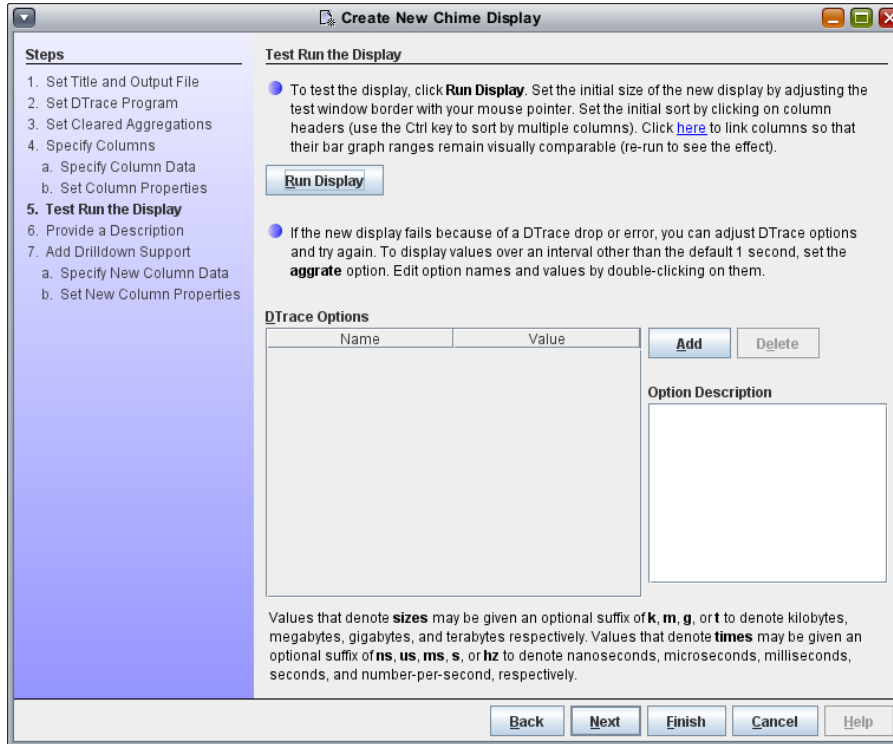
11) Step-11: “Column: 履歴” の表示プロパティを設定します。この例では特に変更の必要がありませんので Next ボタンで次に進みます。



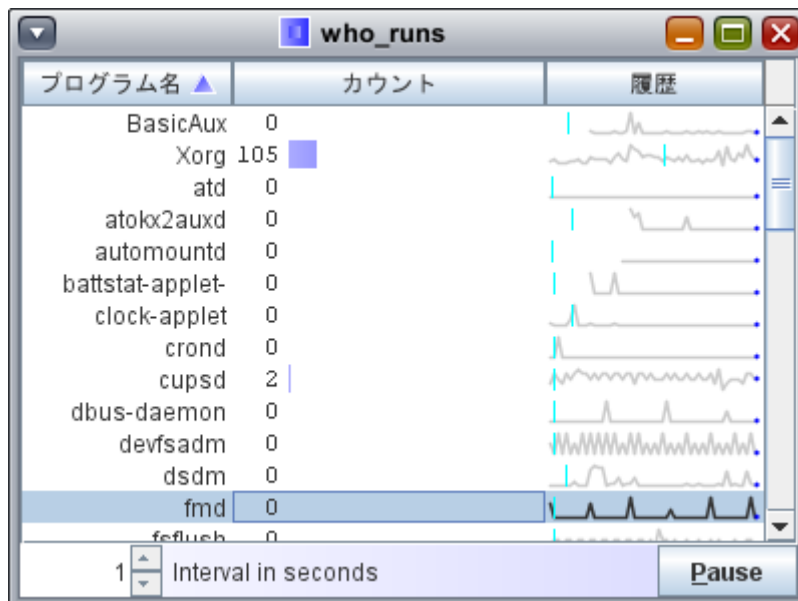
12) Step-12: 表示の設定はすべて終わりました。Next ボタンで次に進みます。



13) Step-13: テスト表示です。“Run Display” ボタンで表示のテストを行います。



テストの結果です。



今回のデモでの紹介はここまでです。“Finish” ボタンで、作成した Display を保存し終了します。作成した Display は、Chime 初期画面の “New Display” から実行することが可能です、また編集画面により修正や機能追加も可能です。

<作成: サン・マイクロシステムズ 原口>

参考情報:

[OpenSolaris Project: Chime Visualization Tool for DTrace:](http://opensolaris.org/os/project/dtrace-chime/)

<http://opensolaris.org/os/project/dtrace-chime/>