## Solaris 3分 クッキング

## **Cook Book**

# DTrace Chime Visualization Tool 編

Solairs Cook Book

1 DTrace Chime Visualization Tool Demo

## 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。

#### <u>デモの材料;</u>

パソコン	TOSHIBA dynabook TX550LS, Intel(R) Celoron(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

Chime Visualization Tool に組み込み済みのデモを紹介します。
 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 0K 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よいでしょう。

/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/OSOLOchime/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。

	Chime
File Options Help         File Options Help         Image: Second	<ul> <li>Chime</li> <li>Description</li> <li>A simple aggregation of system calls per second by executable.</li> <li>Demonstrates sparklines and multiple levels of drilldown support (right-click a selection of one or more rows to access drilldown menu items).</li> </ul>
Write File Descriptor Buckets Write Time Buckets Issue alles Stock Counts	Chime Version 1.4 2007/10/09

"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。

	System Calls	_		
Executable 🔺	History	S	stem Calls Per Second	
BasicAux	A	0		-
Xorg	MmmmM	0		
atd		0		
atokx2auxd	^	0		
automountd	//	0		
battstat-applet-	La contra la la contra la contra da	0		
clock-applet	L	0		
crond	Λ	0		
cupsd		0		
dbus-daemon	L	0		
devfsadm	hhhhhhhhhhhhh.	0		
dsdm		0		
fmd		0		
gconfd-2	L	0		
gnome-panel		0		
gnome-screenshot		0		
gnome-session	L	0		
gnome-settings-d		0		
gnome-terminal		0		
gnome-vfs-daemon		0		
gnome-volume-man	L	0		-
1 Interval in seco	nds	0	Paus	se

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 / pic dtrace: description 'syscall:::entry ' matchec ^C	d != \$pid / { @a[execname] = count() }' d 234 probes
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。

	Xorg System Calls by	Function	📃 🗖 🗙
Function 🔺	History	System Calls Pe	Second
accept		0	<b></b>
clock_gettime	M	157	
fcntl	L	0	
getgid	L	0	
getuid	I	0	
lwp_sigmask	Multim	7	=
pollsys	Mann	78	
priocntlsys	٨	0	
read	Manna	129	
setcontext	Multim	7	
setegid	٨	0	
setitimer	M_/Mmm_	14	
shmsys	L .	0	_
uerodeve		0	
1 📮 Interval	in seconds		Pause

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そうなデモ.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::::pswicth
{
    @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{ dtrace: description 'sysinfo:::pswitch' matcl ^C	<pre>@progs[execname]=count()}' ned 3 probes</pre>	
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。

	🕞 Create New Chime Display	
Steps	Set Title and Output File	
1. Set Title and Output File 2. Set DTrace Program	Create New Display	
3. Set Cleared Aggregations     4. Specify Columns     a. Specify Column Data     b. Set Column Properties	O Modify Existing Display	Br <u>o</u> wse
5. Test Run the Display	Set the <b>Title</b> that appears in the title bar of the display.	
<ol> <li>Provide a Description</li> <li>Add Drilldown Support</li> </ol>	<u>T</u> itle Display	
<ul> <li>a. Specify New Column Data</li> <li>b. Set New Column Properties</li> </ul>	Save the resulting display to the indicated <b>Output</b> file.	
	Output display.xml	Bro <u>w</u> se
	Back Next Einish Cano	el <u>H</u> elp

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。

	🕼 Create New Chime Display 🧧 🔲 🗙
Steps	Set Title and Output File
Steps 1. Set Title and Output File 2. Set DTrace Program 3. Set Cleared Aggregations 4. Specify Column B a. Specify Column Data b. Set Column Properties 5. Test Run the Display 6. Provide a Description 7. Add Drilldown Support a. Specify New Column Data b. Set New Column Properties	Set Title and Output File  C Create New Display  Modify Existing Display  Set the Title that appears in the title bar of the display.  Title who_runs  Save the resulting display to the indicated Output file.  Output who_runs.xml Browse
	Back Next Einish Cancel Help

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 Dscript 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 "Sucessful" と表示されます。"Next" ボタンで次に進みます。

	🔩 Create New Chime Display		
Steps	Set DTrace Program		
1. Set Title and Output File 2. Set DTrace Program	○ P <u>r</u> ogram File		Br <u>o</u> wse
<ol> <li>Set Cleared Aggregations</li> <li>Specify Columns</li> </ol>		🗸 Successful	
a. Specify Column Data b. Set Column Properties	Program String	Ch <u>e</u> ck Compile	S <u>a</u> ve Program
5. Test Run the Display	sysinfo:::pswitch		
6. Provide a Description	{     Rapp[execname]=count();		
a. Specify New Column Data	}		
b. Set New Column Properties			
	<u>Program M</u> acro Arguments <u>S</u> ubs	titutions	
	<u>B</u> ack	<u>N</u> ext <u>Finish</u> <u>C</u> i	ancel <u>H</u> elp

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。

🖸 🛛 🕞 Create New Chime Display 🧧 🗖 🗙			
Steps	Set Cleared Aggregations		
<ol> <li>Set Title and Output File</li> <li>Set DTrace Program</li> <li>Set Cleared Aggregations</li> <li>Specify Columns         <ul> <li>a. Specify Column Data</li> <li>b. Set Column Properties</li> </ul> </li> <li>Test Run the Display</li> <li>Provide a Description</li> <li>Add Drilldown Support</li> <li>a. Specify New Column Data</li> <li>b. Set New Column Properties</li> </ol>	<ul> <li>The values of cleared aggregations are reset to zero each time Chime gets the latest aggregate snapshot to update the display. Clear an aggregation if you want values per time interval. Leave an aggregation uncleared if you want running totals instead.</li> <li>@ Clear all aggregations.</li> <li>Clear only selected aggregations.</li> </ul>		
	Back Next Finish Cancel Help		

5) Step-5: "Specify Columns" では、表示させる各列のタイトルを "Colums" 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。

	🕞 Create New Chime Display	
Steps 1. Set Title and Output File 2. Set DTrace Program	Specify Columns  Enter the names of at least two columns in the order they will appear from displayed column header.	left to right in the
<ol> <li>Set Cleared Aggregations</li> <li>Specify Columns         <ol> <li>Specify Column Data</li> <li>Set Column Properties</li> </ol> </li> </ol>	<ul> <li>To edit a column name, double-click on the name.</li> <li>Columns</li> </ul>	
<ol> <li>Test Run the Display</li> <li>Provide a Description</li> </ol>	プログラム名	Insert
7. Add Drilldown Support	アフント 履歴	<u>D</u> elete
b. Set New Column Properties		<u>Specify</u> >
		Specify >>
	Back Next Einish Ca	ncel <u>H</u> elp

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。

	🕞 Create New Chime Display	
Steps	Specify Column Data	
Set Title and Output File     Set DTrace Program     Set Cleared Aggregations     Specify Columns     a. Specify Column Data	Column: プログラム名 Specify what is displayed in the ブログラム名 column:	
b. Set Column Properties     Test Run the Display     Provide a Description     Add Drilldown Support     a. Specify New Column Data     b. Set New Column Properties	Tuple Member an element of an aggregation key ( such as pid in @[pid], a tuple member at index 0 ) Tuple Index	
	<ul> <li>Aggregation Value</li> <li>a value of an appregating action such as count() or sum()</li> </ul>	
	Aggregation Name	
	Atternate View an alternate view of the data in another column Source Column or a join of multiple columns (similar to strjoin)	
	<u>B</u> ack <u>Next</u> <u>Finish</u> <u>Cancel</u>	Help

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。

🗋 Create New Chime Display 📃 📃 🔀
t Column Properties
t Column Properties Column: プログラム名 et プログラム名 column properties (or simply click Next to accept defaults). apple Member itide the プログラム名 column (by checking its Hidden property) if you plan to present an iternate view of its tuple member data in another column. Hidden iter Value Type to handle exceptional cases such as tuple members generated by the stack() ction. alue Type no special type マ lignment right マ Singular Total Label (e.g. processs) Plural Total Label (e.g. processs) primat Respects numeric value of consecutive digits when sorting
Back Next Einish Cancel Help

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。

🕞 Create New Chime Display 📒 🔲 🔀				
	Set Column Properties			
itle and Output File Trace Program leared Aggregations ify Columns acify Column Data : Column Properties Run the Display de a Description Orilldown Support acify New Column Data : New Column Properties	Column: カウント Set カウント column properties (or simply click <b>Next</b> to accept defaults).			
	Set Value Type to handle exceptional cases such as DTrace distribution values generated by the quantize() and lquantize() actions.          Value Type       no special type         Value Per Second         Initial Range       1000         Alignment       left         Bar Color       text			
	Left   Choose Bar Color   Right       Text         If you request Unit Labels, select or enter a format with a decimal point. '0' or #' indicates a digit, '0' forces the inclusion of a zero where # omits it as unnecessary. Specify four digits left of the decimal point. (See example formats in the pulldown list.)         Format       #,##0         Unit Labels       Vnit Multiple         Unit Multiple       1024			
	Back Next Einish Cancel Help			

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。

🖸 🛛 🕞 Create New Chime Display 📃 🔲 🛛					
Steps	Specify Column Data				
1. Set Title and Output File	✓ 2 columns specified ( only this column remaining )				
2. Set Dirace Program 3. Set Cleared Aggregations	Column: 履歴				
4. Specify Columns					
b. Set Column Properties	Specify what is displayed in the 履歴 column:				
5. Test Run the Display	○ <u>T</u> uple Member				
7. Add Drilldown Support	an element of an aggregation key ( such as pid in @[pid], a tuple member at index 0 )				
a. Specify New Column Data	Tuple Index 1				
D. Set New Column Properties					
	a value of an aggregating action such as count() or sum()				
	Aggregation Name				
	Atternate View				
	an alternate view of the data in another column				
	or a join of <u>multiple columns</u> (similar to sujoin)				
	Back Next Finish Cancel Help				

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。

	🕼 Create New Chime Display	<b>— — ×</b>
Steps	Set Column Properties	
Set Title and Output File     Set DTrace Program     Set Cleared Aggregations     Specify Columns	Column: 履歴 Set 履歴 column properties (or simply click <b>Next</b> to accept defaults).	
<ul> <li>b. Set Column Properties</li> </ul>	View Type	
5. Test Run the Display 6. Provide a Description 7. Add Drilldown Support a. Specify New Column Data b. Set New Column Properties	Non-Repeating Group <u>Tuple Member Conversion</u>	•
	Required Column: pid Tuple Member	-
	Sparkine      Respects numeric value of consecutive digits when sorting     Hidden      Alignment default      Total none     Singular Total Label      Plural Total Label	(e.g. process) .g. processes)
	Back Next Einish Canc	el <u>H</u> elp

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。

🖸 🔹 Create New Chime Display 🗧 🖬 🔀				
Steps	Specify Columns			
1. Set Title and Output File     2. Set DTrace Program     3. Set Cleared Aggregations     4. Specify Columns     a. Specify Column Data     b. Set Column Properties	<ul> <li>Enter the names of at least two columns in the order they will appear from left to right in the displayed column header.</li> <li>To edit a column name, double-click on the name.</li> </ul>			
5. Test Run the Display	ブログラム名 [0] <b>イ</b>	Insert		
<ul> <li>a. Specify New Column Data</li> </ul>	カウント @@app ✔ 履歴 (カウント) ✔	Delete		
b. Set New Column Properties		Specify >		
	<ul> <li>To redo specifications, select a column and click Specify &gt; to go to step 4 go to step 4b.</li> </ul>	a or <b>Specify &gt;&gt;</b> to		
	Back Next Finish C	ancel <u>H</u> elp		

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。

	who_runs	📃 🗖 🔀
プログラム名 🔺	カウント	履歴
BasicAux	0	/\ <b>^</b>
Xorg	105	mmm
atd	0	
atokx2auxd	0	۲
automountd	0	· · · · · · · ·
battstat-applet-	0	L/
clock-applet	0	
crond	0	١
cupsd	2	M
dbus-daemon	0	
devfsadm	0	M.M.M.M.M.M.M.M.
dsdm	0	_~~~~~~~~~~~~
fmd	0	
feflueh		
1 - Interv	al in seconds	Pause

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/