

ES/1 NEO

CSシリーズ

CS-Web Option

HTTP Log Processor

使用者の手引き

第33版 2022年10月

©著作権所有者 株式会社 アイ・アイ・エム 2022年

© COPYRIGHT IIM CORPORATION, 2022

ALL RIGHT RESERVED. NO PART OF THIS PUBLICATION MAY
REPRODUCED OR TRANSMITTED IN ANY FORM BY ANY MEANS,
ELECTRONIC OR MECHANICAL, INCLUDING PHOTOCOPY RECORDING,
OR ANY INFORMATION STORAGE AND RETRIEVAL SYSTEM WITHOUT
PERMISSION IN WRITING FROM THE PUBLISHER.

“RESTRICTED MATERIAL OF IIM “LICENSED MATERIALS – PROPERTY OF IIM

目次

第 1 章	機能概要	1
1.1.	HTTP Log Processor の構成とデータの流れ.....	1
第 2 章	動作要件	2
2.1.	対応環境	2
第 3 章	logscn	3
3.1.	logscn の導入	4
3.1.1.	Unix (Linux) 環境への導入	4
3.1.2.	Windows 環境への導入	5
3.2.	logscn の実行	5
3.3.	スケジューリング.....	8
3.3.1.	Unix (Linux) 環境でのスケジューリング.....	8
3.4.	出力ファイル	9
3.5.	Logscn のアンインストール.....	9
第 4 章	log2f.....	10
4.1.	log2f の概要	10
4.2.	log2f の実行	10
4.3.	フィルタファイルの記述.....	11
4.4.	出力レコード	12
4.5.	URL とクエリ文字列の区切り指定.....	13
4.6.	WEB ページ集約機能	15
4.6.1.	ページアクセス	16
4.6.2.	ページ構成要素定義ファイル	19
4.7.	ロギングの指定	22
第 5 章	添付資料 Sun ONE Web Server におけるアクセスログフォーマット指定について	23

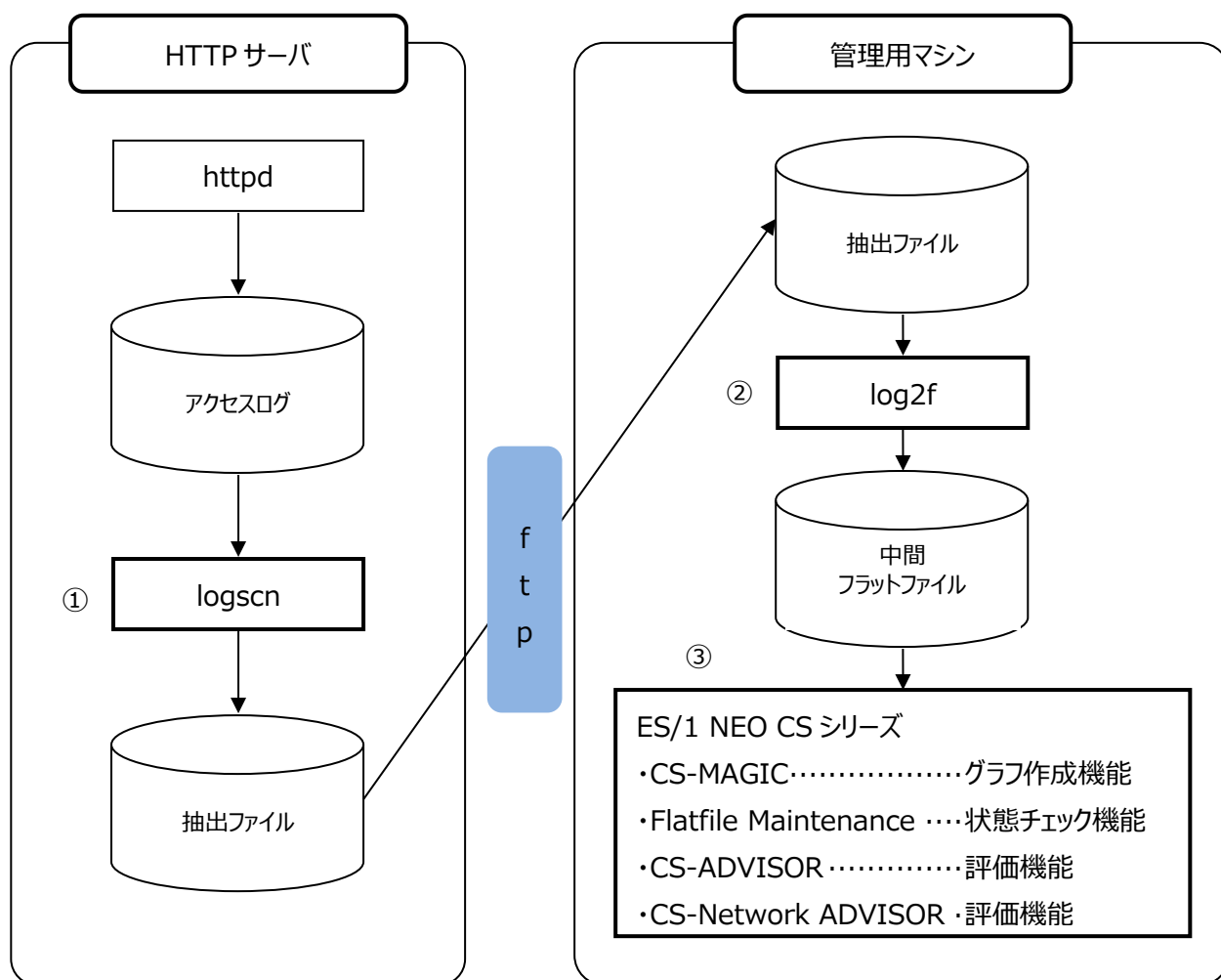
第1章 機能概要

HTTP Log Processor とは、HTTP サーバのアクセスログレコードを抽出し、ES/1 NEO CS シリーズの共通形式データ（フラットファイル）への変換・蓄積を行うプロダクトです。

logscn により、アクセスログファイルからログレコードが抽出されます。抽出ファイルは、log2f によりフラットファイルに変換されます。フラットファイルは、CS-MAGIC でグラフ作成／CS-ADVISOR／CS-Network ADVISOR で評価に利用することができます。

1.1. HTTP Log Processor の構成とデータの流れ

HTTP サーバのアクセスログレコードを抽出し、グラフファイルや CSV 形式ファイルへ出力するまでの流れと、各コンポーネントの動作を説明します。



① logscn によりアクセスログファイルからログレコードを抽出

② log2f により抽出ファイルをフラットファイルに変換

③ 中間フラットファイルを CS-MAGIC にインポートし、グラフ作成・CSV 形式ファイル出力

・作成されたグラフは Performance Web Service を利用し、Web 閲覧することができます。

・生成されたフラットファイルを用い、Flatfile Maintenance で状態チェック／CS-ADVISOR／CS-Network ADVISOR で評価を行うことができます。

第2章 動作要件

2.1. 対応環境

対応製品、動作環境については、「サポート環境」の「HTTP Log」をご参照ください。

〔処理対象アクセスログフォーマット〕

- ①W3C Extended Log Format
- ②Microsoft IIS Log Format
- ③Common Log Format
- ④Combined Log Format
- ⑤Apache httpd （以降、Apache）のカスタマイズされたログフォーマット
（Apache はログフォーマットの柔軟なカスタマイズ機能を提供しています。その為カスタマイズの方法によっては処理することができません。）

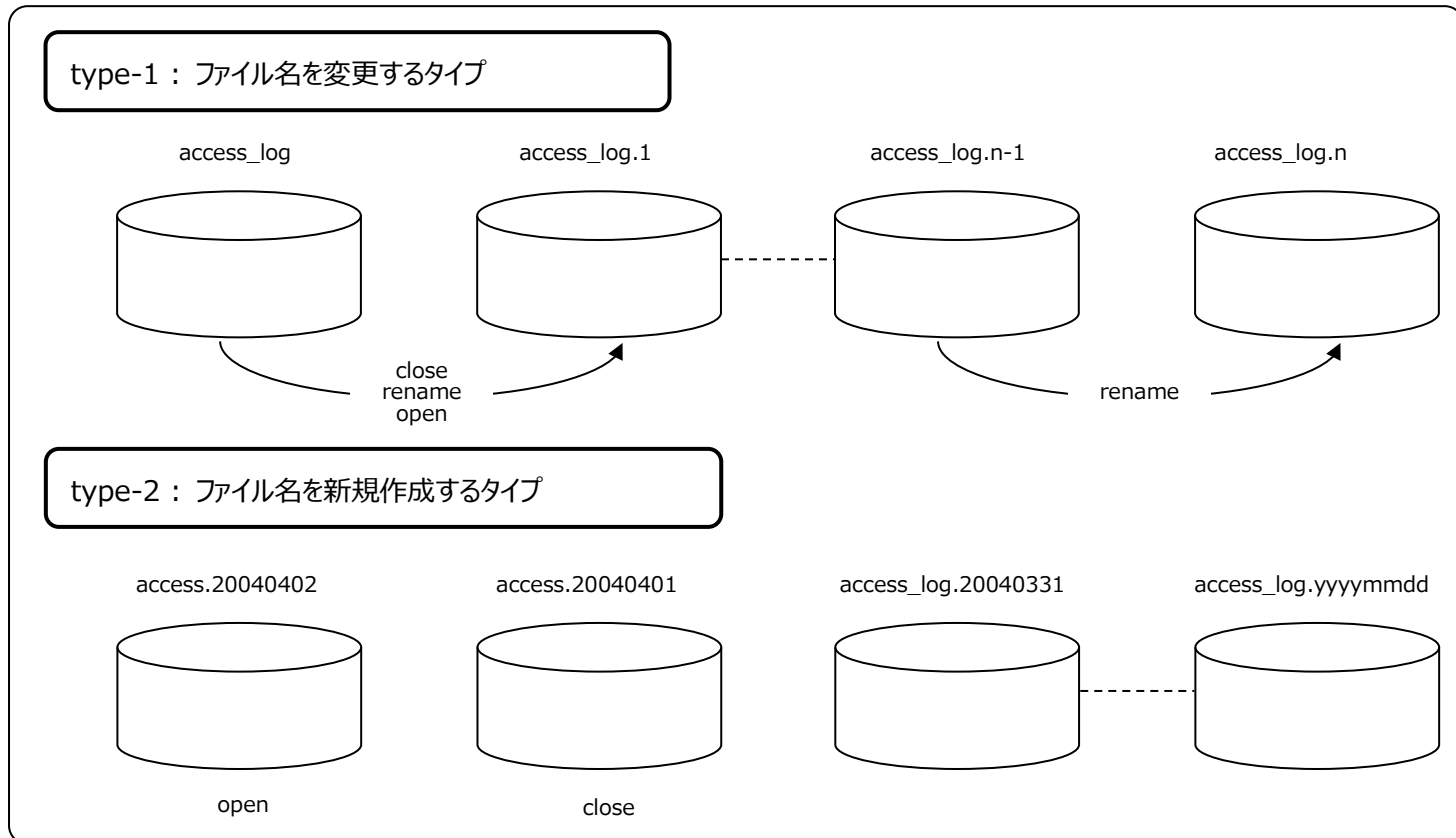
〔処理に必要な条件〕

- ①1 つのログが 1 行に収まっている
- ②URL がログに出力されている
- ③日時がログに出力されている
- ④フィールド区切りが確実に判断できるような出力形式である
（例えばスペース、タブ区切りとして判断する場合は、スペース、タブを含むフィールドは「" "」記号等でクオートされていることが必要です。）
- ⑤Apache 系のアクセスログの場合、レスポンス時間が秒、またはマイクロ秒で出力されている

第3章 logscn

logscn はアクセスログファイルからログレコードを欠落・重複の無いように抽出するプログラムです。また、抽出結果の圧縮も同時に行います。

HTTP アクセスログは逐次ログファイルに追記され、また多くの場合、以下の図のような方法で世代管理されています。



logscn はテキスト形式および gz 形式のアクセスログファイルをサポートします。

3.1. logscn の導入

logscn は基本的に HTTP サーバ上に導入します。

アクセスログファイルがネットワーク共有されている場合、アクセスログファイルを参照できるコンピュータ上に導入し、実行することも可能です。

3.1.1. Unix (Linux) 環境への導入

アクセスログファイルへの読取アクセス権をもつ logscn 実行用ユーザアカウントを用意してください。

(このアカウントは Athene Acquire 用アカウントとは別に必要です。)

実行用ユーザアカウントのホームディレクトリの下に任意の logscn 導入用ディレクトリを作成し、実行環境に合った DVD のディレクトリ内のすべてのファイルをコピーしてください。

実行環境	モジュールのディレクトリ
AIX	/DVD/logscn/AIX
HP-UX	/DVD/logscn/HP-UX
Linux(i386)	/DVD/logscn/linux_86
Linux(x86)	/DVD/logscn/linux_86
Linux(x64)	/DVD/logscn/linux_x64
Solaris(Sparc)	/DVD/logscn/SunSparc

/DVD は DVD マウントポイントとします

3.1.2. Windows 環境への導入

任意の logscn 導入用ディレクトリを作成し、DVD の「x:¥logscn¥Win32¥」内のすべてのファイルをコピーしてください。（x:は DVD ドライブとします）

3.2. logscn の実行

■ Unix（Linux）環境での実行方法

```
install-path/logscn -o output-directory [-c COLUMN] [-E] [-s YYYYMMDD[HH]] [-ux] input-file(s)
```

■ Windows 環境での実行方法

```
install-path ¥logscn -o output-directory [-c COLUMN] [-E|I[-d date-format]]  
[-s YYYYMMDD[HH]] [-ux] input-file(s)
```

- output-directory には抽出結果ファイルを保存する為の既存のディレクトリを指定します。
Unix（Linux）環境の場合は、output-directory のパーミッションを「777」に設定します。
- input-file(s)には処理対象とするアクセスログファイルを指定します。
- アクセスログファイルを含むフォルダには実行権限が必要です。
- アクセスログファイルが世代管理されている場合は、アスタリスク（*）を使用して世代管理の結果生成されるファイルも処理対象として指定します。
- logscn の 2 回目以降の実行時には前回実行時に抽出した日時以降のログを抽出します。
ただし、-o にて指定した出力ディレクトリを変更した場合は、前回の実行結果を引き継ぎません。
（出力ディレクトリに前回実行時の情報を記録・保存する為）

注意！

**Windows 以外の環境では logscn の処理対象ファイルサイズが 2GB を超えると処理できません。
対象とするアクセスログ 1 ファイルが 2GB を超えないよう運用を行ってください。**

オプションを指定しない場合 logscn は、アクセスログが以下の様に出力されているものとして処理を行います。
127.0.0.1 - - [01/Apr/2015:12:59:55 +0900] "GET / HTTP/1.1" ...

この形式は Common Log Format/Combined Log Format に当てはまります。
それ以外の形式のログではオプションを指定する必要があります。

logscn 実行時のオプション

①-c format-expr

ログが apache のカスタマイズされたフォーマットの場合に指定します。apache におけるカスタムログを指定する文字列と同等の文字列を指定します。

ただし、%t を%{書式指定文字列}tとして指定している場合、logscn が認識する書式指定子は以下のもののみです。

%%	(%文字)
%t	(タブ文字)
%a	(曜日の短い英語表現 例. Sun,Mon,...)
%A	(曜日の長い英語表現 例. Sunday,Monday,...)
%b	(月の短い英語表現 例. Jan,Feb,...)
%B	(月の長い英語表現 例. January,February,...)
%F	(yyyy-mm-dd 形式の日付 例. 2015-04-01)
%T	(HH:MM:SS 形式の時刻 例. 23:10:07)
%Y	(yyyy 形式の西暦年 例. 2015)
%y	(yy 形式の西暦年 例. 15)
%m	(mm 形式の月 例. 01,02,...,12)
%d	(dd 形式の日 例. 01,02,...,31)
%H	(HH 形式の時 例. 00,01,...,23)
%M	(MM 形式の分 例. 00,01,...,59)
%S	(SS 形式の秒 例. 00,01,...,59)
%z	(+hhmm または-hhmm 形式のタイムゾーン 例. +0900)

カスタマイズしたフォーマットのタイムスタンプ(%t)フィールドまでの形式が Common Log Format/Combined Log Format と同じであればこのオプションの指定は必要ありません。

このオプションは②-E、③-I オプションを指定した場合には無視されます。

【実行例 1】

```
/home/iim00/logscn -o /home/iim00/custom -c "%D %h %l %u
%t ¥"%r¥" %s %b"/usr/local/apache2/logs/custom_log*
```

【実行例 2】

```
"C:¥IIM¥CS¥HTTP LOG PROCESSOR¥win32¥logscn" -o "
C:¥IIM¥CS¥logscn_output" -c "%h %I %u %t ¥"%r¥" %s %b %T"
"C:¥IIM¥CS¥access_log¥in*.log"
```

メモ !

logscn で指定可能なログ出力項目を識別する英字 (%r 等) は「4.4.出力レコード」に記載されている英字のみです。

注意 !

- apache のログフォーマットの指定でクォートされているフィールドは、合わせてクォートする必要があります。
- Windows システムにおいて、バッチファイルを作成して logscn を実行する場合、「%%B」のように%を2個続けて記述する必要があります。

②-E

ログのフォーマットが W3C Extended Log Format で出力されている場合にこのオプションを指定します。
このオプションを指定した場合、logscn はアクセスログが以下の様に出力されているものとして処理を行います。
2004-04-01 12:59:55 ... (各行に日付・時刻を出力)

③-I

このオプションは Windows 環境でのみ指定可能です。
このオプションを指定した場合、logscn はアクセスログが IIS ログフォーマットで出力されているものとして処理を行います。

④-d date-format

このオプションは Windows 環境で③ -I オプションを指定した場合のみ有効です。
-I のみを指定した場合、logscn はアクセスログの日付が"6/13/04"や"06/13/2004"のように月-日-年の順で'/'で区切られて出力されているものとして処理を行います。
それ以外の形式で日付が出力されている場合は、このオプションで出力日付の書式指定を行ってください。
書式は%x(年)、%m(月)、%d(日)と区切り文字で指定します。
(例: "2004/06/13"と出力されている → -d "%x/%m/%d"と指定)
logscn は年月日の出力桁数は区切り文字と合わせて解釈します。

【実行例 (Windows)】

```
"C:¥IIM¥CS¥HTTP LOG PROCESSOR¥win32¥logscn" -o "C:¥IIM¥CS¥logscn_output" -I -d
"%x/%m/%d" "C:¥IIM¥CS¥access_log¥in*.log"
```

⑤-s YYYYMMDD[HH]

このオプションを指定すると特定の日時以降のログを抽出対象とします。
YYYYMMDD を指定 → 指定日の 0 時以降のログを抽出
YYYYMMDDHH を指定 → 指定日時 00 分以降のログを抽出
抽出済期間より前の日(時)を指定することはできません。

⑥-u

抽出結果を圧縮しません。このオプションを指定しない場合は出力結果を圧縮します。

⑦-x

logscn 実行時に存在するすべてのログレコードを抽出します。このオプションを指定しない場合は実行時における最新ログレコードのタイムスタンプの時間マイナス 1 時間迄のログレコードを抽出します。
(例: 実行時の最新タイムスタンプが 4 月 1 日 / 10 時台 → 4 月 1 日 / 9 時台のログレコードまでを抽出)

⑧-r keepday

このオプションは Windows 環境でのみ指定可能です。

処理対象のアクセスログファイル群の中で実行日の keepday 前より古い日付のものを削除します。

例えば実行日が 3/10 で keepday に 3 を指定した場合は、日付が 3/7 より古いファイルを削除します。

このオプションは HTTP サービスが動作している環境以外にアクセスログファイルをコピー(転送)後、コピーしたアクセスログファイルを対象に logscn を実行する場合に使用することを想定しています。

【実行例】

■ Unix (Linux)

```
/home/iim00/logscn/logscn -o /home/iim00/logscn/logout  
/var/log/http/access_log*
```

■ Windows

```
c:¥iim¥logscn¥logscn -o c:¥iim¥logscn¥logscn¥logout -E  
c:¥inetpub¥logs¥w3svc1¥ex*.log
```

3.3. スケジューリング

logscn の実行をスケジューリングする際は、ログファイルの世代管理による切り替えタイミングと logscn の実行タイミングが異なるようにしてください。

3.3.1. Unix (Linux) 環境でのスケジューリング

cron 機能を使用して logscn を実行する際、-c オプションを指定すると%文字が復帰改行文字に変換されてしまいます。次の例に従って設定してください。

【設定例】

①/home/iim00/ctab.sh ファイルを作成し、logscn 実行文を記述してください。

```
/home/iim00/logscn/logscn -o /home/iim00/custom -c "%D %h %I %u ¥"%t¥"  
¥"%r¥" %s %b" /usr/local/apache2/logs/custom_log*
```

②crontab に以下のように編集し、登録してください。

```
30****/home/iim00/ctab.sh
```

-c オプションを指定しない場合は、logscn 実行文を crontab に登録することができます。

【設定例】

```
30****/home/iim00/logscn -o /home/iim00/custom /usr/local/  
apache2/logs/custom_log*
```

注意！

Unix(Linux)環境で logscn を実行する場合、処理対象とするアクセスログのサイズが 2GB 以下になるようにローテーションを行ってください。

3.4. 出力ファイル

logscn の結果出力ディレクトリ (-o output-directory で指定) にはログの抽出結果ファイルとして、下記のファイルが 1 つ以上生成されます。(前回実行時以降新規のログレコードが出力されていない場合は結果ファイルは生成されません。)

lxxx_YYYYMMDD_HHMMSS_YYYYMMDD_HH.gz (xxx の部分は clf、w3c、iis の何れかになります)

この出力ファイルは log2f プログラムによりフラットファイルに変換することが可能です。結果出力ディレクトリには他にピリオド (.) で始まる数個のファイルが生成されますが、これらのファイルには次回実行時に引き継ぐ実行結果情報が含まれています。log2f プログラム処理の為にファイル転送する場合は、ピリオド (.) で始まるファイルを転送しないように注意してください。

3.5. Logscn のアンインストール

(Unix/Linux)

- (1)logscn の実行をスケジュールしている場合はそれを削除します。
- (2)「3.1.1. Unix (Linux) 環境への導入」で作成した実行用ユーザーアカウントのホームディレクトリを削除します。
- (3)「3.1.1. Unix (Linux) 環境への導入」で作成した実行用ユーザーアカウントを削除します。
- (4)logscn の出力フォルダを上記(2)の階層以外に作成していた場合はそれを削除します。

(Windows)

- (1)logscn の実行をスケジュールしている場合はそれを削除します。
- (2)「3.1.2. Windows 環境への導入」で作成した logscn 導入用ディレクトリを削除します。
- (3)logscn の出力フォルダを上記(2)の階層以外に作成していた場合はそれを削除します。

第4章 log2f

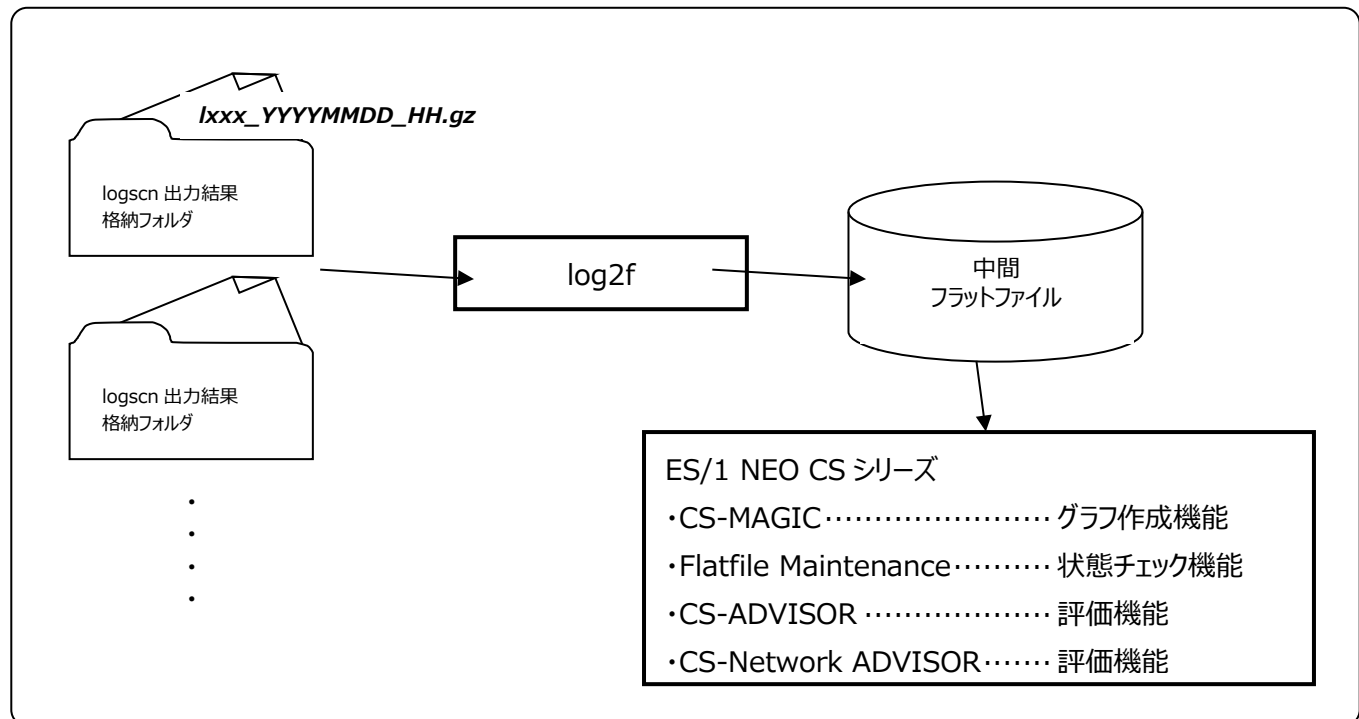
4.1. log2f の概要

log2f は logscn の出力結果を中間フラットファイルに変換するプログラムです。

複数サーバの logscn の出力結果を変換する場合は、サーバ毎に出力結果格納ディレクトリが必要です。

log2f は以下の 2 種類のレコードを作成します。

- (1) 時間帯毎・URL (パス) 毎にアクセス回数や送信バイト数といった情報を集計したレコード (以降、集約レコード)
- (2) 1 つのログの情報をそのまま記録したレコード (以降、詳細レコード)



4.2. log2f の実行

コマンドラインにて `install-path ¥ log2f.exe` として実行してください。特にオプションや引数を指定する必要はありません。iim configuration assistant の設定に従い処理を実行します。

(install-path は管理用コンピュータの ES/1 NEO CS シリーズインストールフォルダ以下の log2f ディレクトリです。)

log2f の動作設定は、別紙マニュアル「CS-Utility iim configuration assistant 使用者の手引き」をご参照ください。

4.3. フィルタファイルの記述

フィルタファイルはテキスト形式ファイルで 1 行が 1 つのフィルタを表現します。

フィルタは以下の形式で記述します。

行頭には '+' あるいは '-' を指定し、':' を間にはさみ、続けて条件文字列を記述します。

[+または -] : [条件文字列]

条件文字列 は、各ログの URL (パス) と比較されます。以下のワイルドカードが指定可能です。

- ? 任意の 1 文字
- * 任意の 0 文字以上の文字列

ログの URL (パス) が条件文字列にマッチした場合、詳細レコードを作成するか、対象外かを決定します。

- + 詳細レコードを作成
- 詳細レコードの作成対象外

フィルタは複数行記述することが可能であり、1 行目から順に条件文字列との比較が行われます。あるフィルタ行にマッチした場合は、それ以降の条件文字列にはフィルタによる比較は行われません。(上の方にあるフィルタが優先されます。)

どのフィルタにもマッチしなかったログは詳細レコードは作成されません。

【例 1】

- :/test/*.jsp
- :/work/*.jsp
- + :*.jsp

この例では /test/ と /work/ 配下以外に存在する、拡張子が ".jsp" という URL へのアクセスログから詳細レコードを作成します。

【例 2】

- :*.gif
- :*.jpg
- :*.png
- + :*

この例では拡張子が ".gif"、".jpg"、".png" 以外の URL へのアクセスログから詳細レコードを作成します。

インストール時には log2f インストールディレクトリ内に、noimage.flit と jsponly.flit という 2 つのサンプルフィルタファイルが存在しています。

noimage.flit は上記の【例 2】と同じ内容であり、jsponly.flit には "+:*.jsp" という 1 行のみ記述されています。

4.4. 出力レコード

log2f が作成するレコードとそれに含まれるフィールドの内容は以下の通りです。

フィールドの値に相当する情報が元々のログに含まれていない場合、そのフィールドの値は欠損値となります。

例えば、Common Log Format/Combined Log Format には応答時間は含まれていませんので、以下の応答時間のフィールドの値は欠損値となります。

■ 集約レコード (ES/1 NEO CS シリーズでの表名 : WLOGSUM)

フィールドの内容	ES/1 NEO CS シリーズ での列名	対応する apache の カスタマイズログのフィールド	対応する W3C-extended- format のフィールド
ログ出力アプリケーション ※1	LOGID		
URL (パス)	URL	%r or %U	'cs-uri-stem'
アクセス回数	ACCCNT		
平均応答時間 (ミリ秒)	RSPAVG	%T or %D	'time-taken'
最大応答時間 (ミリ秒)	RSPMAX	%T or %D	'time-taken'
平均送信バイト数	SNDBYTES	%B or %b	'sc-bytes'
レスポンスコード 1xxx 件数	CNT1XX		
レスポンスコード 2xxx 件数	CNT2XX		
レスポンスコード 3xxx 件数	CNT3XX		
レスポンスコード 4xxx 件数	CNT4XX		
レスポンスコード 5xxx 件数	CNT5XX		

■ 詳細レコード (ES/1 NEO CS シリーズでの表名 : WLOGDET)

フィールドの内容	ES/1 NEO CS シリーズ での列名	対応する apache の カスタマイズログのフィールド	対応する W3C-extended- format のフィールド
ログ出力アプリケーション ※1	LOGID		
アクセス時刻 (分) ※2	ACCMIN	%t	'time'
アクセス時刻 (秒)	ACCSEC	%t	'time'
URL (パス)	URL	%r or %U	'cs-uri-stem'
メソッド	METHOD	%r or %m	'cs-method'
クエリ文字列	QSTR	%r or %q	'cs-uri-query'
クライアントホスト	PEER	%a or %h	'c-ip'
送信バイト数	SNDBYTES	%B or %b	'sc-bytes'
レスポンスコード	RSPCD	%s	'sc-status'
応答時間 (ミリ秒)	RSPMS	%T or %D	'time-taken'
ユーザー名	USERID		※3
リファラー	REFERER	%{Referer}i	

※1 変換設定ユーティリティプログラムで指定する任意の項目です。

※2 アクセス日付はシンボル DATE で、アクセス時刻の時間はシンボル HOUR で求められます。

"... or ..."となっているフィールドが1つのログレコードに両方存在する場合は、後に（行の後ろの方に）出力されている方を使用します。

※3 ユーザー名は IIS ログフォーマットの場合のみ取得可能です。

4.5. URL とクエリ文字列の区切り指定

log2f は基本的にリクエストを '?' か ';' で区切って URL とパラメータ(クエリ文字列)に分割します。例えば、

http://example.com/some?y=2013&m=4

のようなリクエストは、

http://example.com/some

までを URL、

y=2013&m=4

をパラメータとして解釈します。

'?' と ';' の両方を含んでいる場合は先に出現した箇所で区切ります。

log2f の設定ファイルを編集することにより、'?' と ';' 以外の文字で分割を行うことも可能です。

iim configuration assistant にて設定を作成後、

install-path¥log2f.conf

をテキストエディタで編集します。

log2f.conf の内容は以下のような Windows の INI 形式のファイルです。

```
[OUTPUT]
DIR=C:¥iim_work¥cs¥pdbout
[TRX1]
#iimca#31298|31298|HTTPX
DIR=C:¥iim_work¥cs ¥httpOUT¥31298¥31298
SITE=31298
SYSTEM=31298
INTLEN=15
FILTER=C:¥IIM¥CS¥log2f¥noimage.flt
LOGTYPE=C
SUBNAME=httpx
CHRESP=1

[TRX2]
...
```

処理対象フォルダ毎に "[TRXn]" (n は 1 からの連番) というセクションがあります。

'?' と ';' 以外の区切り文字を指定するには対象となるログが配置されるフォルダが "DIR=" キーに記述されたセクションに "EXTSEP=c" (c は区切りとなる半角の 1 文字) を追加します。

例えば、コロン(:) で区切りたい場合は以下の例([TRX2]の上)のように指定します。


```
[OUTPUT]
DIR=C:¥iim_work¥cs¥pdbout
[TRX1]
#iimca#31298|31298|HTTPX
DIR=C:¥iim_work¥cs ¥httpOUT¥31298¥31298
SITE=31298
SYSTEM=31298
INTLEN=15
FILTER=C:¥IIM¥CS¥log2f¥noimage.flt
LOGTYPE=C
SUBNAME=httpx
CHRESP=1
EXTSEP=:

[TRX2]
...
```

ただし、"EXTSEP="キーで指定した文字よりも'?'と';'のほうが優先されます。例えば、"EXTSEP="キーに':'を指定していても、

http://example.com/any:some?y=2013&m=4

のようなリクエストでは'?'が優先され、

http://example.com/any:some

までを URL、

y=2013&m=4

をパラメータとして解釈します。'?'を含まない、

http://example.com/any:y=2013&m=4

のようなリクエストは、

http://example.com/any

までを URL、

y=2013&m=4

をパラメータとして解釈します。

4.6. WEB ページ集約機能

log2f の WEB ページ集約機能は、ページ単位のアクセス状況やレスポンスの解析を行います。

これらの解析結果は、CS-MAGIC の一部クエリーで 사용할ことが可能です。

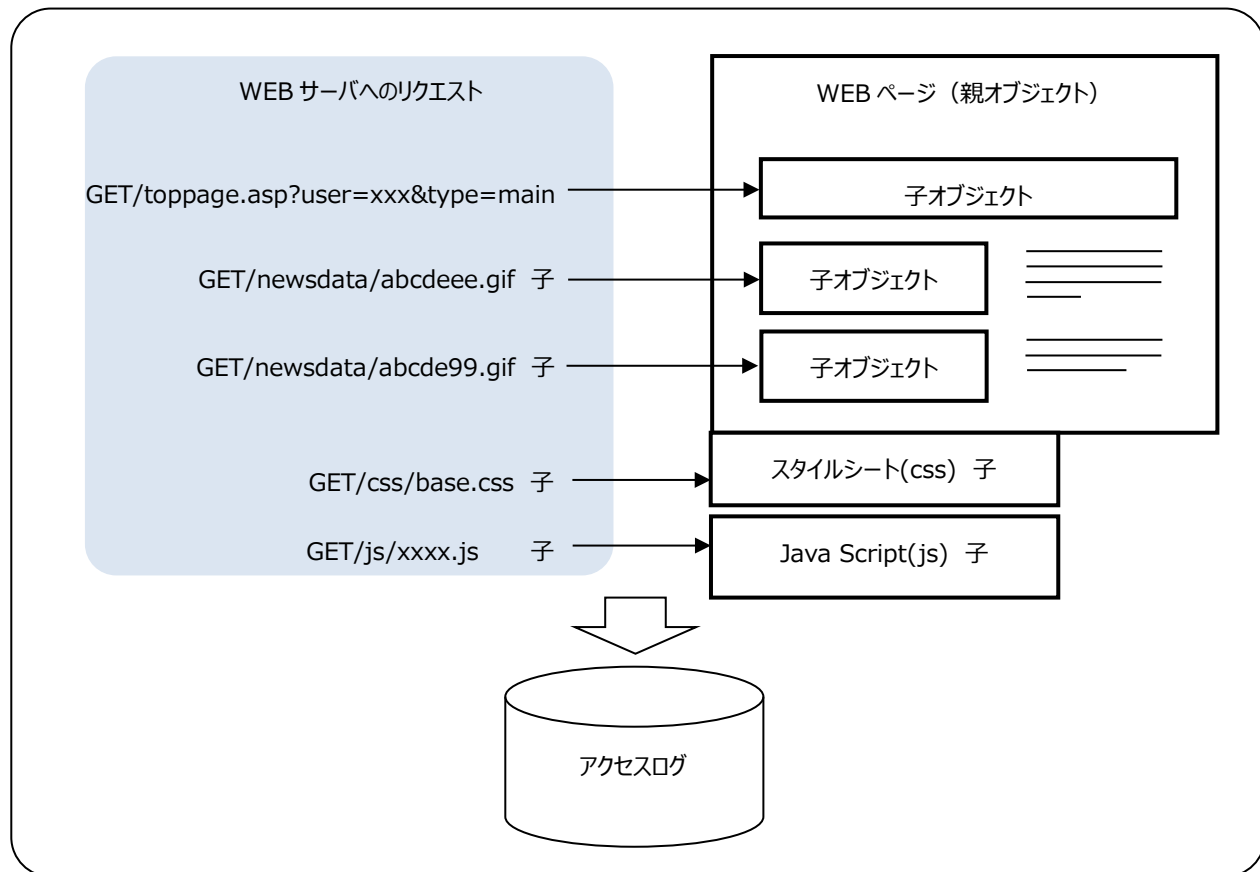
本機能を利用するには、後述する『ページ構成要素定義ファイル』を WEB サーバのページ構成に沿って定義していただく必要があります。

注意！

WEB ページ集約機能を有効にすると、ページアクセス解析結果のレコードが大量に生成される場合があります。
中間フラットファイル格納フォルダやフラットファイルインポート先フォルダのディスク使用量が増加しますので、
空き容量に十分注意した上でご利用ください。

4.6.1. ページアクセス

WEB ページへのアクセスは、複数の URL リクエストで構成されています。WEB クライアントが WEB ページにアクセスする場合、最初におおまかなレイアウト情報を含んだ親オブジェクト（index.html、index.php など）を取得します。WEB クライアントはこの親オブジェクトを解析し、ページ内で参照される子オブジェクトの URL を取得し、WEB サーバに対してリクエストを発行します。子オブジェクトが親オブジェクトと同じ WEB サーバ内に配置されている場合、アクセスログには、親オブジェクトと一緒に子オブジェクトへのアクセスが記録されます。



log2f は外部定義されたページ構成要素の定義情報を利用してアクセスログからページ単位のアクセス記録を解析し、WEB ページ単位のレコードをフラットファイルに出力します。レコードの項目については「CS-MAGIC 使用者の手引き 9.13.3. HTTP ページアクセス情報（詳細）（表名：WLOGPG）」をご参照ください。

注意！

log2f はページ構成要素の定義に従ってページアクセスの解析を行います。定義情報が実際の WEB アプリケーションにマッチしない場合や、NAT のように複数クライアントからのアクセスがひとつにまとめられている場合、正しくページ解析が行われえない可能性があります。

以下の構成の場合、アクセスログが分散してしまい、ページアクセス解析が正しく行われえない可能性があります。

- 複数の WEB サーバで負荷分散をしている
- 同一クライアントからの一連のアクセスが異なる WEB サーバに分散される

<アクセスログのグルーピング>

WEB ページへのアクセスは WEB サーバによってアクセスログに記録されます。log2f はアクセスログを時系列順に並べ、アクセス元のクライアント毎にグルーピングを行います。

ページ 1	
172.16.xx.xx - - [14/Jun/2006:08:29:21 +0900] "POST /webapp/main HTTP/1.1" 200 100 1189612	← 親オブジェクトへのアクセス
172.16.xx.xx - - [14/Jun/2006:08:29:22 +0900] "GET /webapp/picture/rb3.gif HTTP/1.1" 304 - 1695	← 子オブジェクトへのアクセス
172.16.xx.xx - - [14/Jun/2006:08:29:22 +0900] "GET /webapp/picture/rb2.gif HTTP/1.1" 304 - 1569	
ページ 2	
172.16.xx.xx - - [14/Jun/2006:08:29:31 +0900] "POST /webapp/picture/title2.gif HTTP/1.1" 304 - 1660	
172.16.xx.xx - - [14/Jun/2006:08:29:31 +0900] "GET /webapp/picture/back.gif HTTP/1.1" 304 - 1556	
172.16.xx.xx - - [14/Jun/2006:08:30:26 +0900] "GET /webapp/main HTTP/1.1" 200 2802 44382	
172.16.xx.xx - - [14/Jun/2006:08:31:29 +0900] "GET /webapp/main HTTP/1.1" 200 5814 46114	
ページ 3	
172.16.xx.xx - - [14/Jun/2006:08:31:29 +0900] "GET /webapp/picture/list2.gif HTTP/1.1" 304 - 1952	
172.16.xx.xx - - [14/Jun/2006:08:31:29 +0900] "POST /webapp/main HTTP/1.1" 200 100 1189612	
172.16.xx.xx - - [14/Jun/2006:08:31:31 +0900] "GET /webapp/picture/rb3.gif HTTP/1.1" 304 - 1695	
172.16.xx.xx - - [14/Jun/2006:08:31:31 +0900] "GET /webapp/picture/rb2.gif HTTP/1.1" 304 - 1569	

注意！

同一クライアントから複数ページに同時にアクセスしていないことを前提としています。

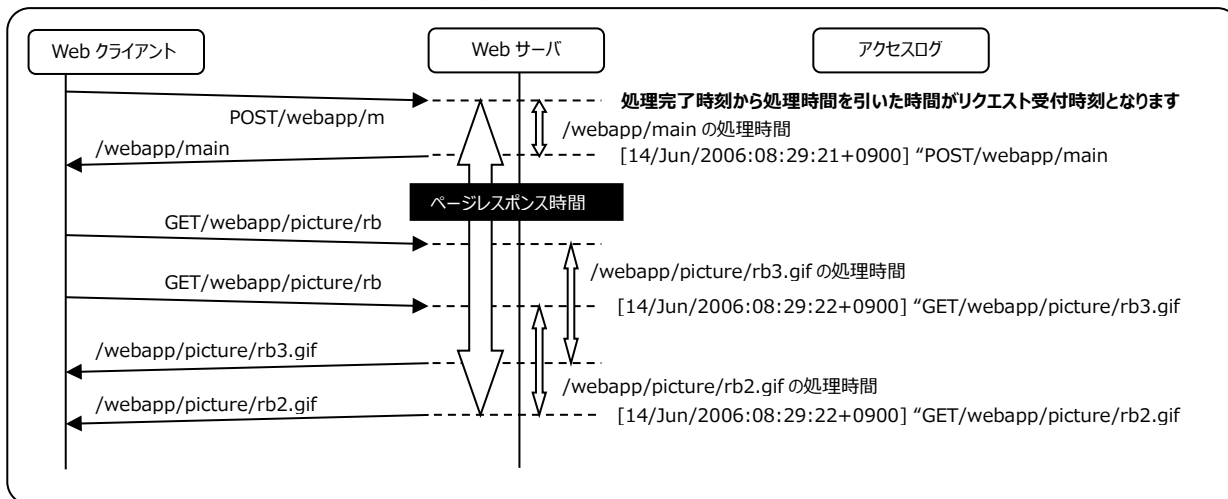
<ページレスポンス>

log2f が取り扱うページレスポンスは、「最初のリクエストを受け付けてから、一連のリクエストに対する最後の処理が完了するまで」をサーバ上で計測した経過時間です。実際にユーザが体感するレスポンスとは異なり、クライアント⇄サーバ間のページデータの伝送時間や、クライアント上で WEB ブラウザがオブジェクトをレンダリングする時間を含みません。

アクセスログには、リクエストの処理完了時刻と、WEB サーバ内での処理時間が記憶されています。log2f では以下の計算式を用いてページのレスポンス時間を計測します。

ページのレスポンス時間 = 最後のリクエストの処理完了時刻 - 最初のリクエストの受付時刻

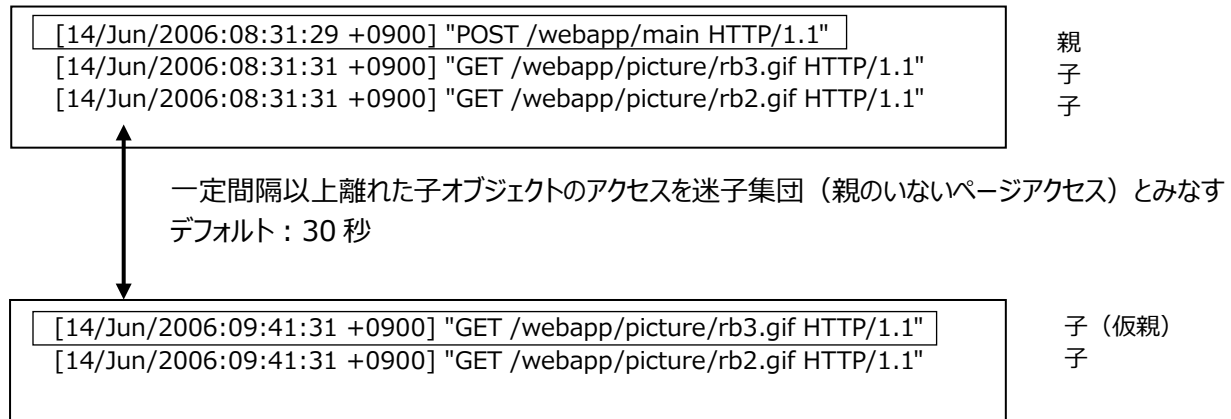
- 最初のリクエストの受付時刻 : アクセスログに記録された親オブジェクトの処理完了時刻から親オブジェクトの処理時間を引いた時間。
- 最後のリクエストの処理完了時刻 : ページに含まれる子オブジェクトのうち、最後にアクセスログに記録された処理完了時刻。



<アクセスログの欠損への対応>

クライアントが WEB ページへアクセスする場合、親オブジェクトのアクセスが行われた直後に子オブジェクトへのアクセスが行われます。何らかの理由により親オブジェクトへのアクセスログが失われると、本来は異なるページとして扱うべき子オブジェクトが、直前のページアクセスの続きとして判断され、異常に長いページアクセスとして報告される可能性があります。

log2f は、このようなアクセスログの欠損に対応するため、直前のアクセスから一定秒数以上離れた子オブジェクトの集団（迷子集団）を別のページアクセスとしてグルーピングしています。この場合、親が不明となりますので、集団の先頭にある子オブジェクトを一時的に仮親とみなし、ページアクセスの解析を行います。



このようにして検出された迷子集団は、log2f が作成する通常の詳細レコード（WLOGPG）に出力されます。迷子集団に限り、直前のアクセスからの経過時間（秒）を WLOGPG レコードの DF フィールドに出力します。

4.6.2. ページ構成要素定義ファイル

ページ単位のアクセスを解析する場合、WEB サーバに行われたリクエストが親オブジェクトであるかどうか、判断する必要があります。ページ構成要素の定義ファイル（cpcheck.txt）は、1 つ 1 つの URL リクエストが親と子のどちらであるかの判別方法を定義するファイルです。log2f はページ構成要素定義ファイルの定義に従い、アクセスログにあるリクエストの親子判別を行いページ単位のグルーピングを行います。

ページ構成要素の定義は、以下の処理対象単位に個別指定することができます。

処理対象範囲
①サイト/システム/ログ出力アプリケーション毎
②サイト/システム毎
③サイト毎
④上記以外の全て

(1)ページ構成要素定義ファイルの配置

log2f 全体で使用するデフォルトの cpcheck.txt は、log2f インストールフォルダ直下（通常は C:\IIM\CS\log2f）に配置されています。すべてのアクセスログを対象に定義を行う場合はデフォルトのページ構成要素定義ファイルを編集してください。

①サイト/システム/ログ出力アプリケーション毎に定義を行う場合

「log2f インストールフォルダ¥サイト名¥システム名¥ログ出力アプリケーション名」フォルダに cpcheck.txt を配置します。

②サイト/システム毎に定義を行う場合

「log2f インストールフォルダ¥サイト名¥システム名」フォルダに cpcheck.txt を配置します。

③サイト毎にまとめて定義を行う場合

「log2f インストールフォルダ¥サイト名」フォルダに cpcheck.txt を配置します。

log2f はより細かい定義（サイト名¥システム名¥ログ出力アプリケーション名）を先に検索し、それが見つからない場合に、順次、より広い範囲の定義（サイト名¥システム名, サイト名）を探します。いずれも見つからない場合、log2f 直下に置かれた共通定義を参照します。

ページ構成要素定義ファイルの適用優先順位
1.log2f インストールフォルダ¥サイト名¥システム名¥ログ出力アプリケーション名¥cpcheck.txt
2.log2f インストールフォルダ¥サイト名¥システム名¥cpcheck.txt
3.log2f インストールフォルダ¥サイト名¥cpcheck.txt
4.log2f インストールフォルダ¥cpcheck.txt

(2) <ページ構成要素定義ファイルの配置>

ページ構成要素定義ファイル（cpcheck.txt）には、アクセスログの各レコードが、ページの親オブジェクトに対するアクセスか、子オブジェクトに対するアクセスかを振り分けるためのルールを記述します。

ページ構成要素定義ファイルは、前部にオプション定義、後部に親子定義を記述します。

オプション定義が必要ない場合は、親子定義のみ記述することが可能です。

①コメント行

空行や、先頭が「#」ではじまる行は、コメント行として読み飛ばされます。

②区切り文字

パラメータの各項目はタブ区切りで指定します。

例) パラメータ A (タブ) パラメータ B

パラメータ C (タブ) パラメータ D

③オプション定義

親子判別に関連する動作オプションを変更したい場合に指定します。

オプション定義は、必ず親子定義の前に記述してください。

■大文字小文字の判別

opt case_sensitive URL の大文字小文字を区別します。(デフォルト)

opt ignore_case URL の大文字小文字を区別しません。

■迷子判定

opt lost_sec n 同一クライアントから n 秒数以上ページアクセスがない場合に、そこでページアクセスが途切れたものとして扱います。[省略時: 30 (秒)]

n 秒以上離れた子アクセスが見つかった場合、そのアクセスは仮親とみなされ、ページアクセス集計では親アクセスとして扱われます。

迷子判定を行わないようにする場合、明示的に「0」を指定してください。

指定可能な値は整数のみです。

④親子定義

アクセスログから親オブジェクト、子オブジェクトを判別するための条件を記述します。オプション定義を使用する場合、必ずオプション定義の後部に親子定義を記述してください。

親子定義は、1 行でひとつの判定条件を記述することが可能です。判定条件は上から順に検査され、最初にマッチした親子定義が適用されます。

親子定義の書式 ([]内は省略可)

親子フラグ[判別箇所 判別手法 パラメータ]

(A) (B) (C)

(A)親子フラグ

親オブジェクトと子オブジェクト、どちらの定義を行うかを指定します。

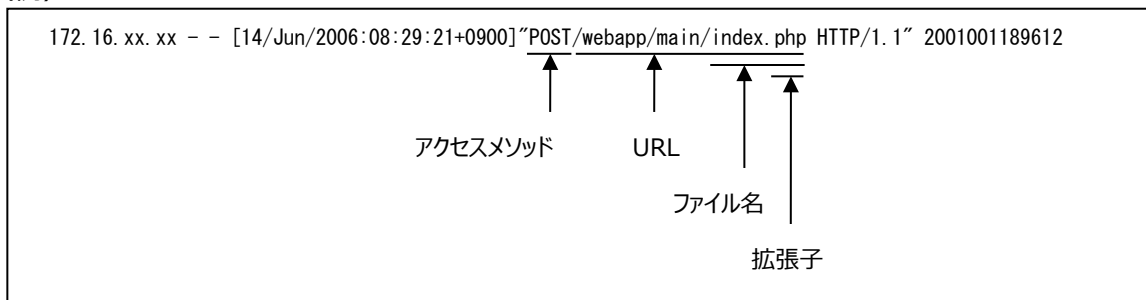
P 親オブジェクト

C 子オブジェクト

(B)判別箇所

アクセスログのどの部分を使用して判別するかを指定します。アクセスログのリクエスト文字列から特定の部位を抽出して親子判定の条件とすることが可能です。

(例)



METHOD	<p>アクセスメソッドを判定条件とします。</p> <p>1 行で指定できるメソッドは 1 つだけです。複数のアクセスメソッドを指定したい場合は METHOD 行を分けて指定してください。</p> <p>例) P METHOD MATCH POST P METHOD MATCH HEAD</p>
PATH	<p>URL 全体を判定条件とします。</p> <p>1 行で指定できる URL は 1 つだけです。複数の URL を指定したい場合は PATH 行を分けて指定してください。</p> <p>例) P PATH WILDCARD/main/login*.php</p>
FILENAME	<p>URL のファイル名を判定条件とします。URL に区切り文字 (/) が含まれる場合、最後の / の後ろの文字列が判定対象となります。 / で終わる URL を対象とする場合は判別手法を「MATCH」、パラメータを空にしてください。1 行で指定できるファイル名は 1 つだけです。複数のファイル名を指定したい場合は FILENAME 行を分けて指定してください。</p> <p>例) P FILENAME MATCH / で終わる URL を対象とする場合</p>
EXTENSION	<p>URL の拡張子を判定条件とします。パラメータにはドット (.) で始まる拡張子を指定します。拡張子のない URL を対象とする場合、パラメータ部分を空にしてください。</p> <p>1 行で指定できる拡張子は 1 つだけです。複数の拡張子を指定したい場合は、EXTENSION 行を分けて指定してください。</p> <p>例) P EXTENSION MATCH .php C EXTENSION MATCH .css C EXTENSION MATCH .gif</p>

(C)判別箇所

判定に使用する手法を指定します。

MATCH	単純一致。文字列が一致するかどうかで判定が行われます。
WILDCARD	ワイルドカードによる一致。使用できる記号は、次の 2 種類です。 * 任意の数の文字列 ? 任意の一文字

注意！

親子判別定義は、上から順に適用されます。そのため、すべての判定にもれた場合に、親オブジェクト、もしくは子オブジェクトのどちらとみなすかを指定することが必要です。

すべての判定に該当しないアクセスを親オブジェクトとみなす場合は P のみの行を、子オブジェクトとみなす場合は C のみの行を末尾に指定してください。

メモ！

log2f のフィルタファイルの定義はページアクセス解析に影響を与えません。フィルタファイルで除外したアクセスもすべてページアクセス解析の対象となります。

P または C の指定が末尾にない場合は、親子判別定義のどれにもマッチしないアクセスログとみなし、ページアクセス解析の対象外となります。特別に除外する場合を除き、どちらかの指定を行ってください。

<デフォルトの親子定義>

初期導入時はページ構成要素定義ファイル内の定義が全てコメントアウトされており、WEB ページ集約機能が OFF の状態になっています。

4.7. ログイングの指定

log2f の実行ログはテキストファイル（log2f ディレクトリ内の"log2f.log"）、イベントログ、およびメッセージボックスに記録することが可能です。

ログイングの指定は log2f ディレクトリ内の log2f.ini ファイルにて行います。このファイルの記述方法および情報の出力レベルについては、別紙マニュアル「Log Utility 使用者の手引き 8. ログ情報出力レベルの設定」を参照してください。

第5章 添付資料 Sun ONE Web Server における アクセスログフォーマット指定について

Sun ONE Web Server では、アクセスログのカスタマイズを行う場合、項目名の指定が apache とは異なりますが、logscn/log2f では apache の項目に変換して指定する必要があります。下記に対応表を記述します。

項目名	apache	Sun ONE Web Server
アクセス時刻	%t	%SYSDATE%
URL	%r or %U	%Req->reqpb.clf-request% or %Req->reqpb.uri%
メソッド	%r or %m	%Req->reqpb.mechod%
クエリ文字列	%r or %q	%Req->reqpb.query%
クライアントホスト	%a or %h	%Ses->client.ip%
送信バイト数	%B or %b	%Req->srvhdrs.content-length
レスポンスコード	%s	%Req->srvhdrs.clf-status%
応答時間	%D	%duration%

上記以外の項目は logscn/log2f で取り扱わないため、すべて %z を指定してください。