記事
Toshihiko Minamoto · 2021年8月31日 12m read

Webをデバッグする - パート2

最初の記事では、Caché
Webアプリケーションのテストとデバッグを外部ツールを用いて行うことについて説明しました。
2回目となるこの記事では、Cachéツールの使用について説明します。

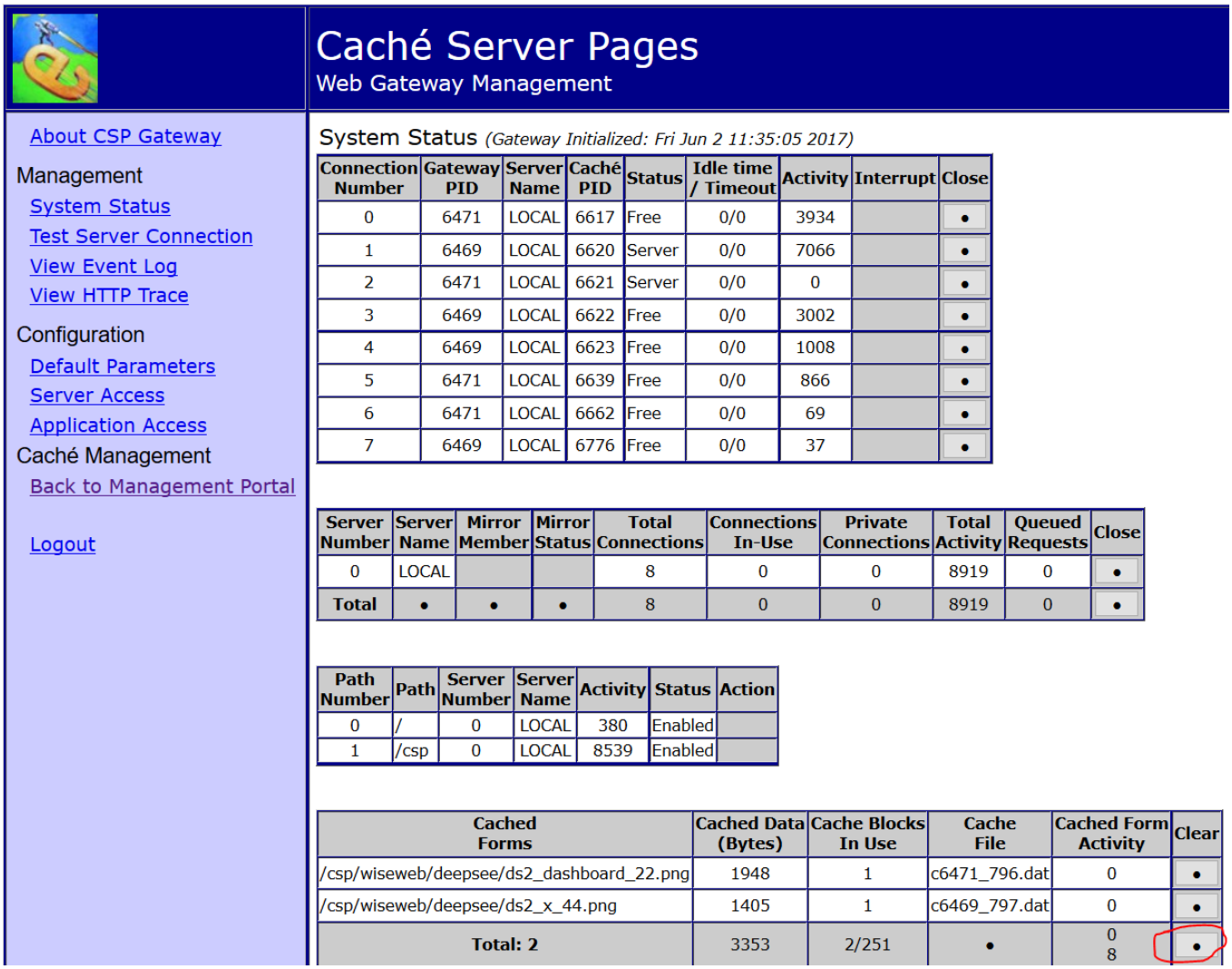
以下について説明します。

- CSP GatewayとWebappの構成
- CSP Gatewayのロギング
- CSP Gatewayのトレース
- ISCLOG
- カスタムロギング
- セッションイベント
- デバイスへの出力

CSP GatewayとWebappの構成

まず初めに、フロントエンドアプリケーションをデバッグする場合、特にそれを開発している場合は、キャッシュは必要ありません。 本番システムでは役立ちますが、開発中には不要です。
Webアプリケーションのロギングを無効にするには、システム管理ポータル → メニュー →
Webアプリケーションの管理 → <あなたのWebアプリ>に移動して、`Serve Files Timeout`を0に設定します。
そして［保存］ボタンを押します。

次に、webアプリケーションの既存のキャッシュを消去する必要があります。
これを行うには、システム管理ポータル → システム管理 → 構成 → CSP Gatewayの管理 →
システムステータスに移動します。 「Cached Forms」テーブルを探し、その最後の行にある「Total」の消去ボタン（ドット）を押してWebアプリケーションのキャッシュを消去します。



CSP Gatewayのロギング

CSP Gatewayに関しては、受信リクエストのロギングがサポートされています（ドキュメント）。
［デフォルトのパラメーター］タブで、希望するログレベル（v9aなど）を指定して、変更を保存します。 v9aは、すべてのHTTPリクエストをGatewayホームディレクトリのhttp.logに記録します（ほかのオプションについてはドキュメントをご覧ください）。 リクエストは次のようにキャプチャされます。

```
GET /forms/form/info/Form.Test.Person HTTP/1.1
Host: localhost:57772
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101 Firefox/53.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:57772/formsui/index.html
Cookie: CSPSESSIONID-SP-57772-UP-
forms-=001000000000yxiocLLb8bbc9SVXQJC5WMU831n2sENf4OGeGa; CSPWSERVERID=144zwBTP
Dnt: 1
Connection: keep-alive
Cache-Control: max-age=0
```

また、パフォーマンスを記録するオプションもあります。
結果はファイルに出力されるか、［イベントログの表示］タブで閲覧することができます。

CSP Gatewayのトレース

そして最後に、CSP
Gatewayの［HTTPトレースの表示］タブでリクエストとレスポンスをトレースすることができます。
トレースを有効にすると、リクエストのキャプチャがすぐに開始します。
デバッグを終えたら、忘れずに無効にしてください。 デバッグセッションは次のように行われます。

注意: トレースのやり方がわかっている場合は、エラーを簡単に理解し、問題を再現することができます。
ロギングは統計収集、パフォーマンスプロファイリングなどに使用します。

また、ほとんどのWebサーバーにも、ロギングとパフォーマンス追跡ツールが備わっています。

ISCLOG

CSP Gatewayは、ネットワークの問題の特定とパフォーマンスの追跡には役立ちますが、Caché内で起きていることをログするにほかのツールが必要となります。
そのようなツールの中でも汎用性に優れているのがISCLOGです。 ドキュメントをご覧ください。

これは、現在のリクエスト処理に関する情報を保存できるグローバル変数です。
ロギングを開始するには、以下を実行します。

```
set ^%ISCLOG = 2
```

ロギングを終了するには、以下を実行します。

```
set ^%ISCLOG = 0
```

リクエストは次のようにロギングされます。

```
^%ISCLOG=0
^%ISCLOG("Data")=24
^%ISCLOG("Data",1)=$lb(2,"CSPServer","Header from CSP Size:3744 CMD:h IdSource:3","46
64","FORMS","2017-06-07 10:49:21.341","%SYS.cspServer2","","")
^%ISCLOG("Data",1,0)="^h30000 "_$c(14,0,0)_"A"
^%ISCLOG("Data",2)=$lb(2,"CSPServer","[UpdateURL] Looking up: //localhost/forms/form/
info path found: //localhost/forms/ Appl= "_$c(2,1,3,4)_"@"_$c(3,4,1,2,1,9,1)_"/forms
/"_$c(2,1,3,4,1,2,1,2,1,2,4,3,4,1,2,1,9,1,7,1)_":%All"_$c(8,1)_"/forms"_$c(7,1)_"FORM
S"_$c(2,1,2,1,3,4,1,2,1,2,1,3,4,1,2,1,4,4,132,3,3,4,2,3,4,2,2,1,4,4,16,14,2,4,3,4,1,3
,4,1,2,1,3,4,1,2,1,16,1)_"Form.REST.Main"_$c(2,4,2,4),"4664","FORMS","2017-06-07 10:4
9:21.342","%CSP.Request.1","124","L3DfNILTaE")
^%ISCLOG("Data",3)=$lb(2,"CSPServer","[UpdateURL] Found cls: Form.REST.Main nocharset
convert:  charset:UTF-8 convert charset:UTF8","4664","FORMS","2017-06-07 10:49:21.342
","%CSP.Request.1","124","L3DfNILTaE")
^%ISCLOG("Data",4)=$lb(2,"CSPServer","[HTML] Determined request type","4664","FORMS",
"2017-06-07 10:49:21.342","%SYS.cspServer2","124","L3DfNILTaE")
^%ISCLOG("Data",4,0)=$lb("usesession",1,"i%Class","Form.REST.Main","i%Service","REST"
,"NOLOCKITEM","","i%GatewayError","")
^%ISCLOG("Data",5)=$lb(2,"CSPSession","[%LoadData] Loading CSP session, nosave=0","46
64","FORMS","2017-06-07 10:49:21.342","%CSP.Session.1","","L3DfNILTaE")
^%ISCLOG("Data",5,0)=$lb(900,,0,5567742244,$c(149)_"Ù"_$c(3)_"ó»à"_$c(127)_",½"_$c(14
9,10)_"\"_$c(18)_"v"_$c(128,135)_"3Vô"_$c(11)_"*"_$c(154)_"PÏG¥"_$c(140,157,145,10,13
1)_"*",2,"FORMS","001000000000L3DfNILTaE1cDBJNjyQdyLwKq4wCXP82ld8gic",,0,"ru","L3DfNI
LTaE",2,1,"/forms/",$lb("UnknownUser","%All,%Developer","%All,%Developer",64,-5590387
37),"","","","2017-06-07 10:48:51","2017-06-07 10:49:04","Basic ZGV2OjEyMw==","Mozill
a/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101 Firefox/53.0","","",0,"/forms/
```

```
","","","","",4,"","","","","http://localhost:57772/formsui/index.html")
^%ISCLOG("Data",6)=$lb(2,"CSPServer","[CSPDispatch]Requested GET /forms/form/info","4
664","FORMS","2017-06-07 10:49:21.342","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",7)=$lb(2,"CSPServer","[CSPDispatch] ** Start processing request newSe
s=0","4664","FORMS","2017-06-07 10:49:21.342","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",7,0)="/forms/form/info"
^%ISCLOG("Data",8)=$lb(2,"CSPServer","[CSPDispatch] Service type is REST has-soapacti
on=0 nosave=0","4664","FORMS","2017-06-07 10:49:21.342","%SYS.cspServer","124","L3DfN
ILTaE")
^%ISCLOG("Data",9)=$lb(2,"CSPServer","[CSPDispatch]About to run page: Form.REST.Main"
,"4664","FORMS","2017-06-07 10:49:21.342","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",9,0)=$lb("UnknownUser","%All,%Developer","%All,%Developer",64,-559038
737)
^%ISCLOG("Data",10)=$lb(2,"CSPServer","[callPage] url=/forms/form/info ; Appl: /forms
/ newsession=0","4664","FORMS","2017-06-07 10:49:21.342","%SYS.cspServer","124","L3Df
NILTaE")
^%ISCLOG("Data",11)=$lb(2,"CSPServer","[callPage]Imported security context ; User: Un
knownUser ; Roles: %All,%Developer","4664","FORMS","2017-06-07 10:49:21.342","%SYS.cs
pServer","124","L3DfNILTaE")
^%ISCLOG("Data",12)=$lb(2,"CSPServer","[OutputCSPGatewayData]: chd=1;","4664","FORMS"
,"2017-06-07 10:49:21.431","%CSP.Response.1","","L3DfNILTaE")
^%ISCLOG("Data",13)=$lb(2,"CSPResponse","[WriteHTTPHeaderCookies] Session cookie: CSP
SESSIONID-SP-57772-UP-forms-=001000000000L3DfNILTaE1cDBJNjyQdyLwKq4wCXP82ld8gic; path
=/forms/;  httpOnly;","4664","FORMS","2017-06-07 10:49:21.431","%CSP.Response.1","124
","L3DfNILTaE")
^%ISCLOG("Data",14)=$lb(2,"CSPServer","[callPage] Return Status","4664","FORMS","2017
-06-07 10:49:21.431","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",14,0)=1
^%ISCLOG("Data",15)=$lb(2,"CSPServer","[OutputCSPGatewayData]: chd=1;","4664","FORMS"
,"2017-06-07 10:49:21.431","%CSP.Response.1","","L3DfNILTaE")
^%ISCLOG("Data",16)=$lb(2,"CSPServer","[Cleanup]Page EndSession=0; needToGetALicense=
-1; nosave=0; loginredirect=0; sessionContext=1","4664","FORMS","2017-06-07 10:49:21.
431","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",17)=$lb(2,"CSPSession","[Cleanup] EndSession=0 nosave=0","4664","FORM
S","2017-06-07 10:49:21.431","%SYS.cspServer","124","L3DfNILTaE")
^%ISCLOG("Data",18)=$lb(2,"CSPSession","[%SaveData] Saved: ","4664","FORMS","2017-06-
07 10:49:21.431","%CSP.Session.1","","L3DfNILTaE")
^%ISCLOG("Data",18,0)=$lb(900,,0,5567742261,$c(149)_"Ù"_$c(3)_"ó»à"_$c(127)_",½"_$c(1
49,10)_"\"_$c(18)_"v"_$c(128,135)_"3Vô"_$c(11)_"*"_$c(154)_"PÏG¥"_$c(140,157,145,10,1
31)_"*",2,"FORMS","001000000000L3DfNILTaE1cDBJNjyQdyLwKq4wCXP82ld8gic",,0,"ru","L3DfN
ILTaE",2,1,"/forms/",$lb("UnknownUser","%All,%Developer","%All,%Developer",64,-559038
737),"","","","2017-06-07 10:48:51","2017-06-07 10:49:21","Basic ZGV2OjEyMw==","Mozil
la/5.0 (Windows NT 10.0; WOW64; rv:53.0) Gecko/20100101 Firefox/53.0","","",0,"/forms
/","","","",5,"","","","","http://localhost:57772/formsui/index.html")
^%ISCLOG("Data",19)=$lb(2,"CSPServer","[Cleanup] Restoring roles before running destr
uctor","4664","FORMS","2017-06-07 10:49:21.431","%SYS.cspServer","","L3DfNILTaE")
^%ISCLOG("Data",19,0)=$lb("UnknownUser","%All,%Developer","%All,%Developer",64,-55903
8737)
^%ISCLOG("Data",20)=$lb(2,"CSPServer","[Cleanup] End","4664","FORMS","2017-06-07 10:4
9:21.431","%SYS.cspServer","","L3DfNILTaE")
^%ISCLOG("Data",20,0)=""
^%ISCLOG("Data",21)=$lb(2,"GatewayRequest","[CSPGWClientRequest] GWID: ed-pc:57772; R
equest: sys_get_system_metricsTimeout: 5","11112","%SYS","2017-06-07 10:49:23.141","%
SYS.cspServer3","","")
^%ISCLOG("Data",22)=$lb(2,"GatewayRequest","[CSPGWClientRequest] GWID: 127.0.0.1:5777
2; Request: sys_get_system_metricsTimeout: 5","11112","%SYS","2017-06-07 10:49:23.141
","%SYS.cspServer3","","")
^%ISCLOG("Data",23)=$lb(2,"GatewayRequest","[SendSimpleCmd:Server:Failed] WebServer:
127.0.0.1:57772; Gateway Server Request Failed","11112","%SYS","2017-06-07 10:49:23.1
```

```
41","%CSP.Mgr.GatewayMgrImpl.1","","")
^%ISCLOG("Data",23,0)=0
^%ISCLOG("Data",24)=$lb(2,"GatewayRequest","[GetMetrics]","11112","%SYS","2017-06-07
10:49:23.141","%CSP.Mgr.GatewayMgrImpl.1","","")
^%ISCLOG("Data",24,0)=""
```

また、以下のような簡単なスクリプトを使用して、グローバルをファイルに出力することができます。

```
set p="c:\temp\isclog.txt"
open p:"NW"
use p zw ^%ISCLOG
close p
```

カスタムロギング

デフォルトのロギングツールも非常に優れていますが、いくつかの問題があります。

- 内容が一般的であり、アプリケーションが認識されない
- より詳細なオプションを使うと、パフォーマンスに影響する
- 構造化されていないため、情報の抽出が困難な場合がある

そのため、より具体的なケースを網羅するために、独自のカスタムロギングシステムを作成することができます。以下は、%requestオブジェクトの一部を記録する永続クラスのサンプルです。

```
/// Incoming request
Class Log.Request Extends %Persistent
{
/// A string indicating HTTP method used for this request.
Property method As %String;
/// A string containing the URL up to and including the page name
/// and extension, but not including the query string.
Property url As %String(MAXLEN = "");
/// A string indicating the type of browser from which the request
/// originated, as determined from the HTTP_USER_AGENT header.
Property userAgent As %String(MAXLEN = "");
/// A string indicating the MIME Content-Type of the request.
Property contentType As %String(MAXLEN = "");
/// Character set this request was send in, if not specified in the HTTP headers
/// it defaults to the character set of the page it is being submitted to.
Property charSet As %String(MAXLEN = "");
/// A %CSP.Stream containing the content submitted
/// with this request.
Property content As %Stream.GlobalBinary;
/// True if the communication between the browser and the web server was using
/// the secure https protocol. False for a normal http connection.
Property secure As %Boolean;
Property cgiEnvs As array Of %String(MAXLEN = "", SQLPROJECTION = "table/column");
Property data As array Of %String(MAXLEN = "", SQLPROJECTION = "table/column");
ClassMethod add() As %Status
{
    set request = ..%New()
    quit request.%Save()
}
Method %OnNew() As %Status [ Private, ServerOnly = 1 ]
{
    #dim %request As %CSP.Request
```

```
    #dim sc As %Status = $$$OK
    quit:'$isObject($g(%request)) $$$ERROR($$$GeneralError, "Not a web context")
    set ..charSet = %request.CharSet
    if $isObject(%request.Content) {
        do ..content.CopyFromAndSave(%request.Content)
    } else {
        set ..content = ""
    }
    set ..contentType = %request.ContentType
    set ..method = %request.Method
    set ..secure = %request.Secure
    set ..url = %request.URL
    set ..userAgent = %request.UserAgent
    set cgi = ""
    for {
        set cgi=$order(%request.CgiEnvs(cgi))
        quit:cgi=""
        do ..cgiEnvs.SetAt(%request.CgiEnvs(cgi), cgi)
    }
    // Only gets first data if more than one data with the same name is present
    set data = ""
    for {
        set data=$order(%request.Data(data))
        quit:data=""
        do ..data.SetAt(%request.Get(data), data)
    }
    quit sc
}
}
```

Log.Request テーブルに新しいレコードを追加するには、コードに次の呼び出しを追加します。

```
do ##class(Log.Request).add()
```

これは非常に基本的なサンプルであり、必要に応じてコメント、変数、またはほかの様々なものを記録するように拡張することができ、そうすることが推奨されます。
このアプローチの主なメリットは、記録されたデータに対してSQLクエリを実行できることにあります。
独自のロギングシステムの構築に関する詳細は、こちらの記事をご覧ください。

セッションイベント

イベントクラスは、%CSP.Session
オブジェクトの寿命中に呼び出されるインターフェースを定義するクラスです。 これを使用するには、
%CSP.SessionEventsをサブクラス化して、実行するメソッドコードを実装する必要があります。
次に、CSPアプリケーション構成内で、イベントクラスを作成したクラスに設定します。

次のコールバックを利用できます。

OnApplicationChange
OnEndRequest
OnEndSession
OnLogin
OnLogout
OnStartRequest
OnStartSession

- OnTimeout

たとえば上記のカスタムロギングは、これらのメソッドから呼び出すことができます。

デバイスへの出力

最も単純なオプションの1つは、すべてのオブジェクトをレスポンスとして出力するCSPユーティリティメソッドです。 これをコードの任意の場所に追加するだけで出力できます。

```
set %response.ContentType = "html"
do ##class(%CSP.Utils).DisplayAllObjects()
return $$$OK
```

まとめ

Webアプリケーションのデバッグにはさまざまなツールを使用することができます。
手元のタスクに最適なものを選択するようにしてください。

皆さんは、CachéからWebアプリケーションをデバッグする際には、どのようなヒントやトリックを使用していますか？

#デバッグ #フロントエンド #Caché

ソースURL:
https://jp.community.intersystems.com/post/web%E3%82%92%E3%83%87%E3%83%90%E3%83%83%E3%82%B0%E3%81%99%E3%82%8B-%E3%83%91%E3%83%BC%E3%83%882