



SupportPac CH0A CICS Transaction Gateway Extended Request Monitor

*Installation and User Guide
Version 1.1.0 – September 2014*

Author

Richard Ruston
rustonr@uk.ibm.com

IBM United Kingdom Limited
Hursley Park, Winchester, SO21 2JN. UK

© Copyright IBM Corporation 2012, 2014. All Rights Reserved.

Overview

This SupportPac provides a sample request monitoring utility for use with the CICS Transaction Gateway (CICS TG). It is designed to help identify possible performance related issues and to alert a system administrator in the case of failing or slow requests.

The utility provides the capability to log a single line message for any request which exceeds a specified response time or which fails to complete with an ECI_NO_ERROR return code. It also provides a regular statistics summary and can be requested to log any request currently in flight which has exceeded the specified response time.

The output of the exit is configurable and kept to a minimum so that the performance impact is as low as possible. Transaction tracking tokens are logged where available. In addition, logging can be dynamically enabled and disabled though CICS TG Request monitoring exit control.

Output can be sent to standard HFS, Windows or UNIX filing systems, or to MVS DD destinations. When used with standard filing system dynamic archiving support can be enabled using the inbuilt Java logging function.

The exit has been designed to work with versions of CICS TG V8.1 and later, and provides additional functionality when running under CICS TG V9.0 and later.

Specifically, when run on a CICS TG V9.0 or later system, the RequestDetails exit point is exploited, and requests are updated with the DSS selected CICS server and Origindata when that exit point is called. Additionally, if XCT data is provided by WebSphere (V8.5 and later), this information is also available.

Contents

This SupportPac contains the following artefacts:

- License Directory - Directory containing license information
- CH0A.pdf - This file
- CH0A.jar - JAR file for use with CICS TG

Installation

Preparation

To prepare for utility for installation you will need to download and unpack the SupportPac as follows:

1. Copy the file CH0A.zip to a temporary directory on your chosen system and uncompress using unzip.
2. Copy the CH0A.jar to a directory on your system, for example on z/OS you could use the HFS: /u/cicstg/tmp or on Windows the directory C:\temp.

Note: FTP transfers should use binary mode.

3. To install the SupportPac refer to the relevant instructions for using a Request Monitoring exit with CICS TG.

Installation

To install the utility for usage with CICS TG perform the following instructions:

1. Add CH0A.jar to the CLASSPATH used to start your Gateway daemon.

For example on Windows:

```
SET CLASSPATH=%CLASSPATH%;C:\temp\CH0A.jar
```

For example on z/OS update your CLASSPATH statement in your STDENV member as follows:

```
CLASSPATH=/u/cicstg/tmp/CH0A.jar
```

2. Update your Gateway daemon configuration file (ctg.ini) to name the monitoring exit in the SECTION GATEWAY. For example:

```
requestexits=com.ibm.ctg.samples.requestexit.ExtendedMonitor
```

Configuring Logging Destination

To enable the use of a log file rather than the default stdout destination set the `EM.out` system property.

For example on AIX start the Gateway with the `-j` flag as follows:

```
ctgstart -j-DEM.out=/tmp/ch0a.log
```

If using the `ctgd` daemon on UNIX or Linux set the following variable in the `ctgd.conf` configuration file:

```
CTGD_PARAMS=-j-DEM.out=/tmp/ch0a.log
```

If using a Windows service set the following startup parameter:

```
ctgservice -R -A-j-DEM.out=c:\temp\ch0a.log
```

To log to JES on z/OS add the following DD statement to the Gateway job:

```
//TXNLOG DD SYSOUT=*
```

Then set the output destination in jzos format in the `CTGSTART_OPTS` variable in your `STDENV` as follows:

```
CTGSTART_OPTS=-j-DEM.out=//DD:TXNLOG
```

To log to an MVS dataset set the `CTGSTART_OPTS` variable as follows:

```
CTGSTART_OPTS=-j-DEM.out=//'XX.YY.ZZ'
```

Log Archiving

When using HFS or zFS file systems on z/OS, or UNIX or Windows file systems, it is possible to use Java file logging to dynamically rotate log files so that a set number of log files are used based on a pre-defined maximum log file size. This is controlled by the following 2 Java system properties:

`EM.log.number` - Number of log files

`EM.log.size` - Maximum log file size (KB)

For example on z/OS to limit the number of log files to 3 and the maximum size to 2MB set the following additional start options:

```
CTGSTART_OPTS=-j-DEM.log.number=3 -j-DEM.log.size=2048
```

Configuration

The following Java system properties control functioning of the user exit and can be configured as start options. If an option is not set to an acceptable value it will be ignore and the default used. The properties can be set the same way as for the log configuration properties.

EM.log.Irts

Acceptable Values: off
Default value: on
Action: Disables logging of transactions which exceed the time specified in EM.Irttimeout.

EM.log.reqerrors

Acceptable Values: off
Default value: on
Action: Disables logging of transactions being returned to the client API with a return code which is not ECI_NO_ERROR.

EM.log.intstats

Acceptable Values: off
Default value: on
Action: Disables logging of periodic statistical data.

EM.log.extdata

Acceptable Values: off
Default value: on
Action: Disables logging of Origindata and XCT data.

EM.Irttimeout

Acceptable Values: Time in milliseconds
Default value: 5000
Action: Sets the threshold after which a transaction is considered to be long running.

EM.intervaltimeout

Acceptable Values: Time in milliseconds
Default value: 60000
Action: Sets the duration of an interval for statistics logging. Statistics will be logged at the end of every interval.
Note: This is not related to the CICS TG statistics interval.

EM.reapertimeout

Acceptable Values: Time in milliseconds
Default value: 600000
Action: Sets the time interval between sweeping the request table for orphaned requests, which are then removed from the table.

EM.orphantimeout

Acceptable Values: Time in milliseconds
Default value: 300000
Action: Sets the time that a request is allowed to be outstanding (with no response) before being removed from the request table. Requests which have been outstanding for longer than this value will be removed from the table the next time reaper processing takes place.

Exit Commands

The exit provides three commands which can be passed in at runtime using the standard CICS TG administration interfaces.

CH0A_START

Action: Enables logging to file and allows requests to be added to the work queue for processing.

CH0A_STOP

Action: Disables logging to file and stops requests being added to the work queue for processing. The background thread continues to run.

NOTE: Since requests are no longer passed to the background thread for processing, any requests in flight at the time the command is issued will be subsequently detected as long running, and will be eventually reaped, as the responses will not be detected.

CH0A_LRTDUMP

Action: Scans the internal request table and logs a line per request which has been in flight longer than the time specified by EM.lrttimeout

Exit Output

The exit writes various different types of information to the log file, shown below. Note that the examples below come from a variety of different sources and configurations, and some information, such as Origindata and XCT data is only available in certain configurations.

Startup Information

If the exit is correctly configured, the following message will be written to the CICS TG Gateway daemon log:

```
12/17/12 10:21:32:978 [0] CTG8402I Request Monitoring Exit
com.ibm.ctg.samples.requestexit.ExtendedMonitor is enabled
```

Startup information is written to the log defined for the exit:

```
12/17/12 10:21:32:973 CH0A: Starting ExtendedMonitor V1.1.0 on
z/OS, JVM encoding: IBM1047
12/17/12 10:21:32:975 CH0A: [LRT Timeout: 5000ms]
[Interval Timeout: 60000ms] [Reaper Timeout: 600000ms]
[Orphan Timeout: 300000ms]
12/17/12 10:21:32:975 CH0A: [Log Long Running Transactions: true]
[Log Request Errors: true] [Log Interval Statistics: true]
```

If the version of CICS TG being used supports the RequestDetails exit point, the following message is logged:

```
12/17/12 10:21:32:976 CH0A: Detected CICS TG supports
RequestDetails event
```

Statistics Information

The exit will log a summary of statistical information at the end of each interval as specified by `EM.intervaltimeout`. Note: this information is generated internally by the exit and is not related to statistics information provided by CICS TG.

```
12/17/12 10:51:33:490 CH0A: !INT-STATS! : [In: 0] [Out: 0]
[Inflight: 0] [Total: 0] [Avg Time: 0] [Max Time: 0]
12/17/12 10:52:33:567 CH0A: !INT-STATS! : [In: 599] [Out: 596]
[Inflight: 3] [Total: 596] [Avg Time: 131] [Max Time: 677]
12/17/12 10:53:34:411 CH0A: !INT-STATS! : [In: 18261] [Out: 18264]
[Inflight: 0] [Total: 18860] [Avg Time: 240] [Max Time: 5033]
```

Long Running Request Information

When a request completes (i.e. it is passed in by CICS TG at the ResponseExit point), the exit will log a line if the request took longer to complete than the time specified in EM.lrttimeout. Note that the 'server' field will be the DSS selected CICS server if available.

```
12/17/12 10:52:57:197 CH0A: !LRT-ALERT! :  
[Client IP: 9.20.196.204] [Program: EC02] [TransId: null]  
[Server: IPICNONC] [Rc: ECI_NO_ERROR]  
[Start: Mon Dec 17 10:52:51 GMT 2012]  
[End: Mon Dec 17 10:52:56 GMT 2012] [Time: 5033ms]  
[Origin Data: ApplidQualifier: CTG900  
Q, Applid: CTG900, startStck: 0xCAA1A77895737A00, clientIpAddr:  
9.20.196.204, clientPort: 49555, userCorrelator: null]  
[XCT Root: UNKNOWN] [XCT Parent: UNKNOWN] [XCT Current: UNKNOWN]
```

Failing Request Information

When a request completes with a return code other than ECI_NO_ERROR (i.e. any failing request), the exit will write a line to the log as follows:

```
12/05/12 11:50:44:779 CH0A: !RQF-ALERT! :  
[Client IP: 9.20.196.204] [Program: EC01] [TransId: null]  
[Server: NONCE] [Rc: ECI_ERR_CICS_DIED]  
[Start: Wed Dec 05 11:49:45 GMT 2012]  
[End: Wed Dec 05 11:50:43 GMT 2012] [Time: 58793ms]  
[Origin Data: UNKNOWN] [XCT Root: UNKNOWN] [XCT Parent: UNKNOWN]  
[XCT Current: UNKNOWN]
```

Stuck Request Information

When the CH0A_LRTDUMP command is received by the exit, the request table is processed and a line will be logged for any in flight request which has been in flight for longer than the time specified in EM.lrttimeout:

```
12/18/12 16:15:07:034 CH0A: !STK-ALERT! : [Client IP: 127.0.0.1]  
[Program: EC01] [TransId: null] [Server: NONCE] [Rc: null]  
[Start: Tue Dec 18 16:14:49 GMT 2012] [End: UNKNOWN]  
[Time: 17329ms]  
[Origin Data: ApplidQualifier: RUZCTGV9, Applid: RUZCTG01,  
startStck: 0xCAA3314D11A70100, clientIpAddr: 127.0.0.1,  
clientPort: 50920, userCorrelator: XCT AABs/DC26tB-AAAAAAAAAAB  
AABs/DC26tB-AAAAAAAAAAC]  
[XCT Root: AABs/DC26tB-AAAAAAAAAAB]  
[XCT Parent: AABs/DC26tB-AAAAAAAAAAB]  
[XCT Current: AABs/DC26tB-AAAAAAAAAAC]
```


Other Information

The exit will also log various other messages, these include when requests are reaped, commands are received and errors encountered.

Modifying the Exit Code

The supplied CH0A.jar contains both the compiled byte code and the Java source enabling further customisation if required. To modify and compile the code you will need the following:

- IBM SDK V6 (on any supported platform)
- ibmjzos.jar - Available in the Java 6 IBM SDK for z/OS in the lib/ext directory.
- ctgclient.jar – Available with the CICS TG in the classes directory