

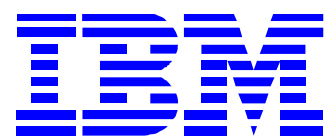
# **WebSphere Application Server v8.5.5**

## **Cross Component Trace (XCT)**

(2013. 07. )

IBM SWG WebSphere Technical Sales

이정운 과장(juwlee@kr.ibm.com)



## 0) Cross Component Trace (XCT) 간략 소개

안녕하세요 freeman 입니다.

이번 강좌에서 언급드릴 내용은 Cross Component Trace (XCT) 기능입니다. XCT 기능은 이전 버전의 강좌에서("제대로 맛보는 IBM WAS v8.5 - 8. HPEL 모드 로깅") 언급한 HPEL 모드 로깅과 함께 사용될 수 있는 기능으로 같은 요청을 책임지는 여러 개의 스레드나 프로세스에 의해 생성 되어진 로그나 트레이스의 연관관계를 XCT 를 활용하여 손쉽게 확인이 가능한 기능입니다. 이 기능을 활용하면 복잡한 애플리케이션 문제를 쉽게 디버깅하기 하거나 HTTP 나 JMS 요청과 응답에 대해서 자세한 정보를 확인하는 것이 가능합니다.

말이 조금 어려울 수도 있는데 쉽게 이야기 하자면 XCT 기능을 활용하면 하단의 로그에서 나타나는 것처럼 requestID 가 추가되어 여러 개의 스레드나 프로세스에 의해 생성 되어진 로그라고 해도 연관관계를 쉽게 확인할 수 있으며 당연히 이를 통해서 문제 분석도 빠르게 가능합니다.

```
[3/18/11 14:50:17:391 EDT] 00000018 W UOW= source=com.ibm.somelogger.QuickLogTest org= prod= component=
thread=[WebContainer : 1] requestID=AAP+k9s6JZ9-AAAAAAAAAA
hello world
```

또한, 해당 기능을 확장하여 쉽게 애플리케이션과 연관된 스레드의 모든 로그에 JEE 애플리케이션 이름을 추가('appName') 하거나 개발자들이 새로운 LogRecordContext API 를 활용하여 본인 고유의 확장을 추가하는 것도 가능합니다.

```
logViewer.sh -includeExtensions appName=ACMESHovels -format advanced
...
[12/10/11 10:52:01:500 EST] 000001c6 1 UOW= source=com.acme.SomeLogger thread=[WebContainer : 6] org= prod= component= appName=[ACMESHovels]
This is a trace entry from the MyShovels application
```

이를 통해서 애플리케이션을 수정하거나 변경할 필요 없이 간단한 옵션설정이나 트레이스 만으로 보다 빠르게 문제 분석을 수행하실 수 있습니다.

그럼 좀 더 자세한 사항은 이제부터 진행되는 강좌를 참고해주시기 바라면서 지금부터 강좌를 진행하도록 하겠습니다.

## 1) XCT 설정

1. XCT 를 설정하기 위해서는 기본적으로 HPEL 로깅 모드를 사용해야 합니다. 관리콘솔에서 서버 > 해당 서버명 > 로깅 및 추적 메뉴를 클릭하면 하단과 같은 화면을 확인할 수 있으며 여기서 HPEL 모드로 전환을 선택하면 HPEL 로깅 모드로 변환됩니다.

2. 다음과 같이 HPEL 로깅 모드로 변경된 것을 확인하고 해당 WAS 서버를 재시작하면 실제 반영됩니다.

3. 재시작후 관리콘솔을 통해서 서버 > 해당 서버 > 로깅 및 추적 > HPEL 로그 구성 메뉴를 클릭하면 하단과 같은 HPEL 로그 설정정보를 확인할 수 있습니다.

4. 참고적으로 HPEL 모드에 대한 오래된 로그 기록 제거 정책은 하단과 같이 3가지가 있습니다.

애플리케이션 서버 > server1 > server1 > HPEL 로그 구성

이 페이지를 사용하여 HPEL(High Performance Extensible Logging) 로그 옵션을 구성하십시오. 디렉토리에 있음) 또는 HPEL 로그 및 추적 보기 링크를 사용하여 볼 수 있습니다.

구성 **런타임**

#### 일반 특성

##### \* 디렉토리 경로

☒ 로그 기록 버퍼링 사용

☒ 다음 위치에서 새 로그 파일을 매일 시작: 오전 12시

##### 로그 기록 제거 정책

☒ 가장 오래된 기록 정리 시작

로그 크기가 최대 크기에 다다를 때

가장 오래된 기록이 유효 기간 한계에 도달할 때

유효 기간 또는 크기 제한에 도달할 때

최대 로그 크기

50

MB

5. 로그 기록 제거 정책과 마찬가지로 공간 부족 조치도 하단과 같은 3가지 중에 선택하실 수 있습니다.

#### \* 공간 부족 조치

로그 중지

서버 중지

이전 레코드 제거

로그 중지

취소

6. HPEL 로깅 모드로 전환이 완료되었다면 로그 세부사항 레벨 변경 메뉴를 클릭하여 세부 메뉴로 들어 갑니다. 세부 메뉴화면에서 “로그 및 추적 상관 사용” 을 선택하고 “로그 및 추적 레코드에 요청 ID 포함 및 상관 로그 레코드 작성” 항목을 선택하여 XCT 기능을 enable 합니다.

애플리케이션 서버

#### 애플리케이션 서버 > XCTTestSrv > 로그 세부사항 레벨 변경

Java 로깅에 의해 처리되는 이벤트를 제어하려면 로그 레벨을 사용하십시오. 개별 컴포넌트에 대한 로그 세부사항 레벨을 지정하려면 컴포넌트를 클릭하고 사전 정의된 컴포넌트 그룹에 대한 로그 세부사항을 지정하려면 그룹을 클릭하십시오. 로그 세부사항 레벨을 선택하려면 컴포넌트 또는 그룹 이름을 클릭하십시오. 로그 세부사항 레벨은 누적됩니다. 목록의 맨 위 근처에 있는 레벨은 모든 후속 레벨을 포함합니다.

구성

#### 일반 특성

##### 로그 세부사항 레벨 변경

☐ 잠재적으로 중요한 데이터의 로깅 및 추적 사용 안함(경고: 서버에서 이 옵션을 적용할 경우 로그 세부사항 레벨 설정이 수정될 수 있음)

컴포넌트를 선택하고 로그 세부사항 레벨을 지정하십시오. 여기에 지정된 로그 세부사항 레벨은 전체 서버에 적용됩니다. 개별 컴포넌트에 대한 로그 세부사항 레벨을 지정하려면 컴포넌트 및 그룹을 펼치고, 사전 정의된 컴포넌트 그룹에 대한 로그 세부사항 레벨을 지정하려면 그룹을 클릭하십시오. 로그 세부사항 레벨을 선택하려면 컴포넌트 또는 그룹 이름을 클릭하십시오. 로그 세부사항 레벨은 누적됩니다.

\*=info

☒ 컴포넌트 및 그룹

##### 상관

로그 및 추적 상관을 사용하여 두 개 이상의 스레드, 프로세스 또는 서버에서 서비스하는 항목을 동일한 작업 단위에 속한 것으로 식별되도록 하십시오.

☒ 로그 및 추적 상관 사용

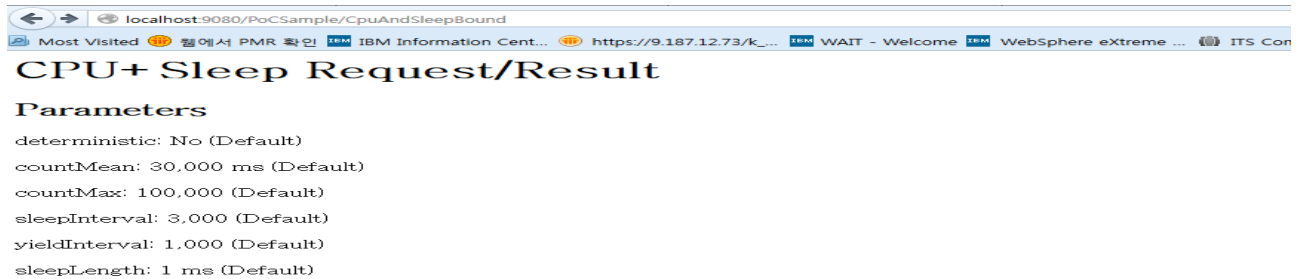
☐ 로그 및 추적 레코드에 요청 ID 포함

☒ 로그 및 추적 레코드에 요청 ID 포함 및 상관 로그 레코드 작성

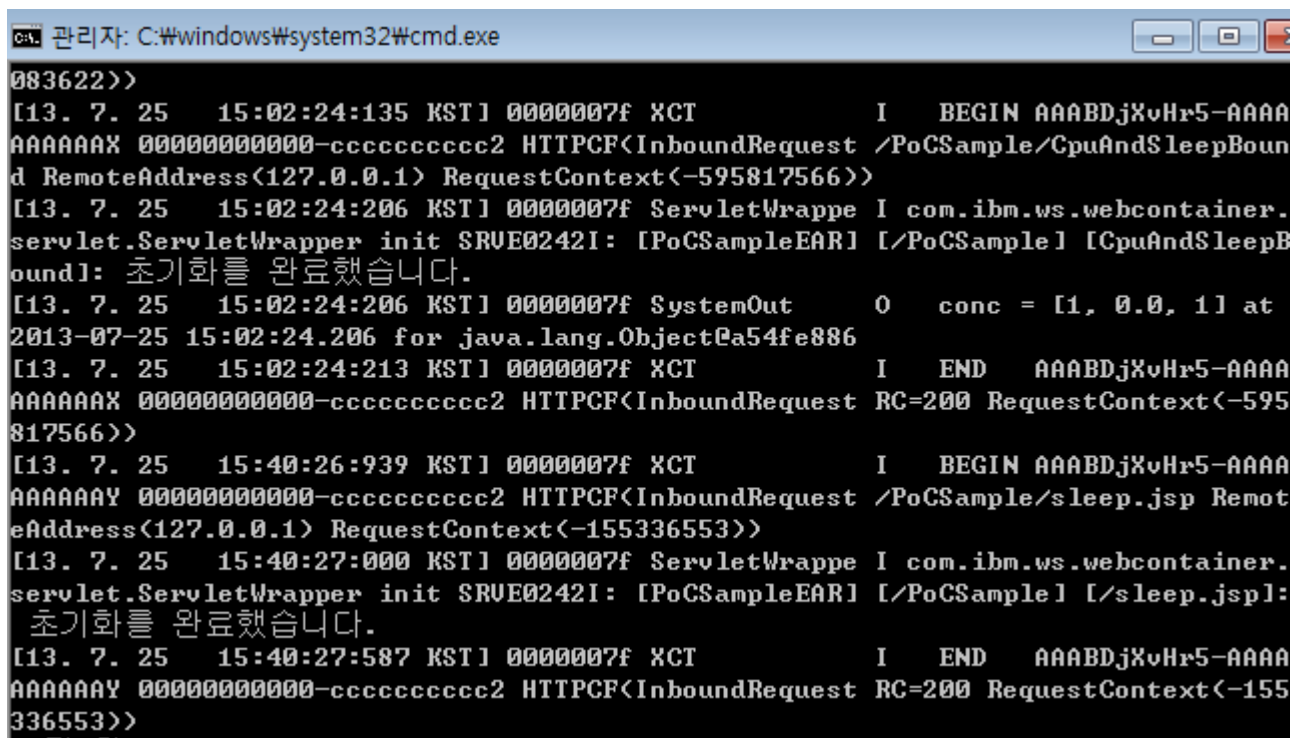
☐ 로그 및 추적 레코드에 요청 ID 포함, 상관 로그 레코드 작성 및 데이터 스냅샷 캡처

## 2) XCT 테스트

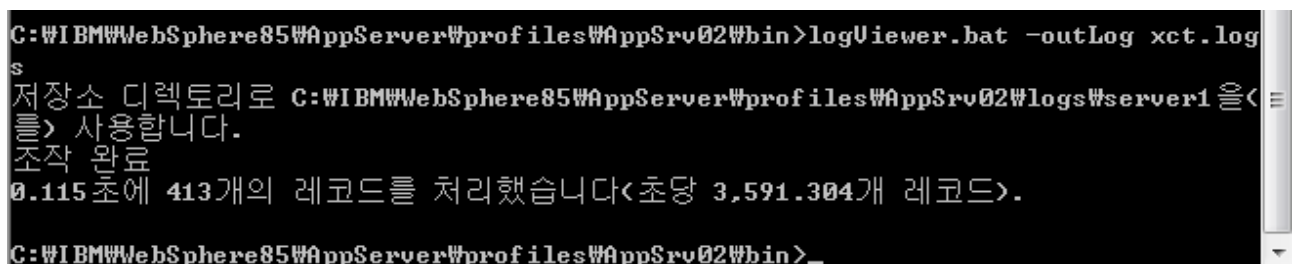
1. XCT 를 테스트하기 위해 샘플 애플리케이션을 하나 요청해서 수행합니다.



2. logViewer.bat 스크립트를 이용해서 log 를 확인해보면 이전에 봐왔던 log 와 다르게 XCT 가 enable 되어있는 것을 확인할 수 있습니다.



3. 다만 해당 방식으로는 분석이 어려우므로 "logViewer.bat -outLog xct.logs" 를 수행하여 log 를 text 형태의 파일로 생성합니다.



4. text 형태의 파일로 log 를 만들었으므로 기존에 가진 다양한 방식으로 좀 더 쉽게 분석이 가능합니다.

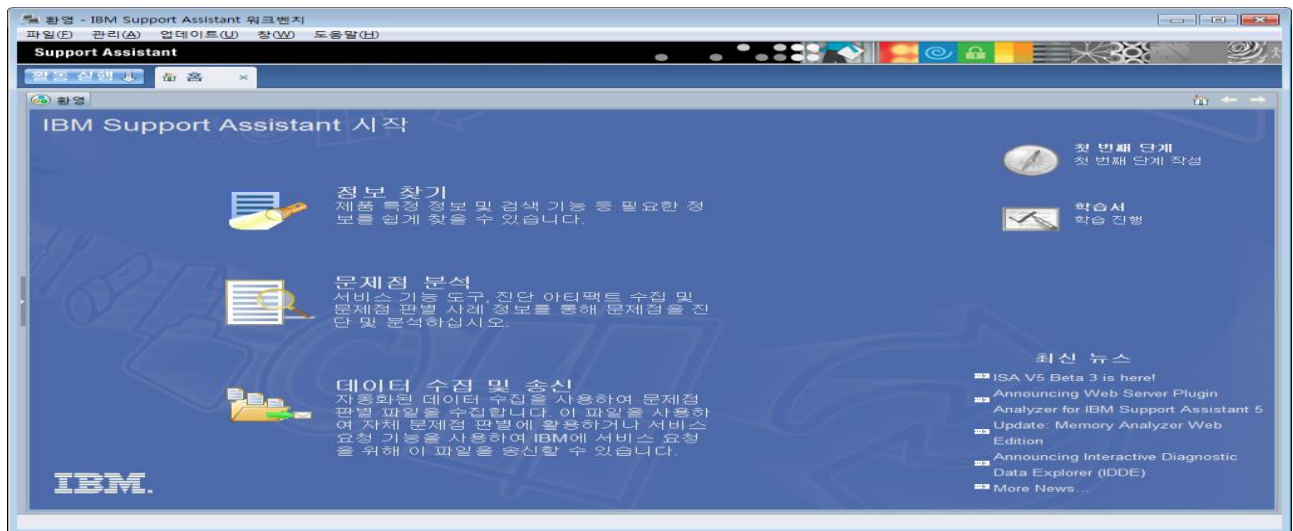
```

http_plugin.log 30_Problem&solving.html xct.logs sleep.jsp
0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170
520 [13. 7. 25 15:59:46:227 KST] 00000082 XCT I END AAABDjXvHr5-AAAAAAAAAAv 000000000000-cccccccccc2 HTTPCF(InboundRequest RC=200 RequestContext(1110070675)) J
521 [13. 7. 25 15:59:50:831 KST] 00000082 XCT I BEGIN AAABDjXvHr5-AAAAAAAAAAw 000000000000-cccccccccc2 HTTPCF(InboundRequest /ibm/console/configSpecDetail.do Rem
522 [13. 7. 25 15:59:50:840 KST] 00000082 ManagerAdmin I TRAS0018I: 추적 상태가 변경되었습니다. 새 추적 상태는 *info입니다. J
523 [13. 7. 25 15:59:50:843 KST] 00000082 ManagerAdmin I TRAS0040I: 구성된 추적 상태에는 현재 서버에 등록된 트거와 일치하지 않는 다음의 스택이 포함되었습니다. J
524 [13. 7. 25 15:59:51:039 KST] 00000082 XCT I END AAABDjXvHr5-AAAAAAAAAAx 000000000000-cccccccccc2 HTTPCF(InboundRequest RC=200 RequestContext(1110070675)) J
525 [13. 7. 25 15:59:51:110 KST] 00000082 XCT I BEGIN AAABDjXvHr5-AAAAAAAAAAx 000000000000-cccccccccc2 HTTPCF(InboundRequest /ibm/console/com.ibm.ws.console.proct
526 [13. 7. 25 15:59:51:112 KST] 00000082 XCT I END AAABDjXvHr5-AAAAAAAAAAx 000000000000-cccccccccc2 HTTPCF(InboundRequest RC=200 RequestContext(1110070675)) J
527 [13. 7. 25 15:59:51:137 KST] 00000082 XCT I BEGIN AAABDjXvHr5-AAAAAAAAAAy 000000000000-cccccccccc2 HTTPCF(InboundRequest /ibm/console/getTraceStrings Remote?
528 [13. 7. 25 15:59:51:143 KST] 00000080 XCT I BEGIN AAABDjXvHr5-AAAAAAAAAAz 000000000000-cccccccccc2 HTTPCF(InboundRequest /ibm/console/secure/javascriptToSess
529 [13. 7. 25 15:59:51:179 KST] 00000082 XCT I END AAABDjXvHr5-AAAAAAAAAAz 000000000000-cccccccccc2 HTTPCF(InboundRequest RC=200 RequestContext(1110070675)) J

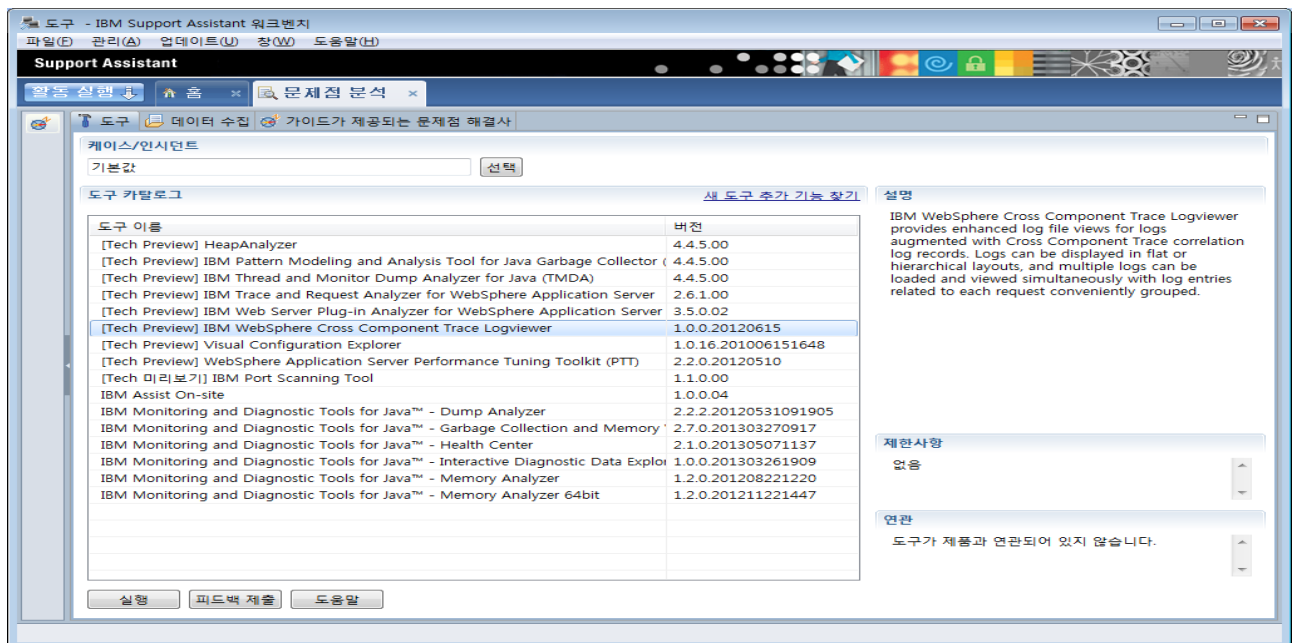
```

5. 또한, 해당 log 파일을 좀 더 손쉽게 분석하기 위해서 IBM 은 IBM Support Assistant 라는 무상 통합 문제 분석 도구를 제공합니다.

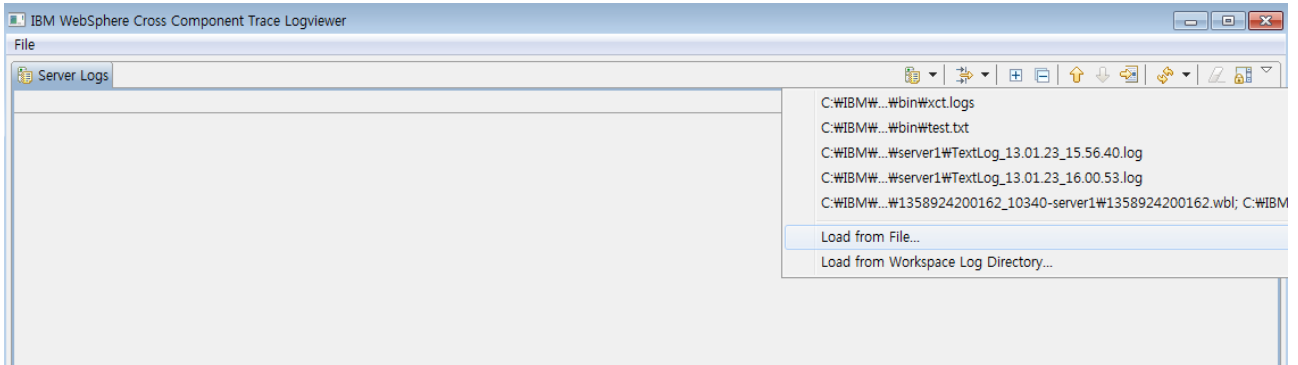
(<http://www-01.ibm.com/software/support/isa/workbench.html>)



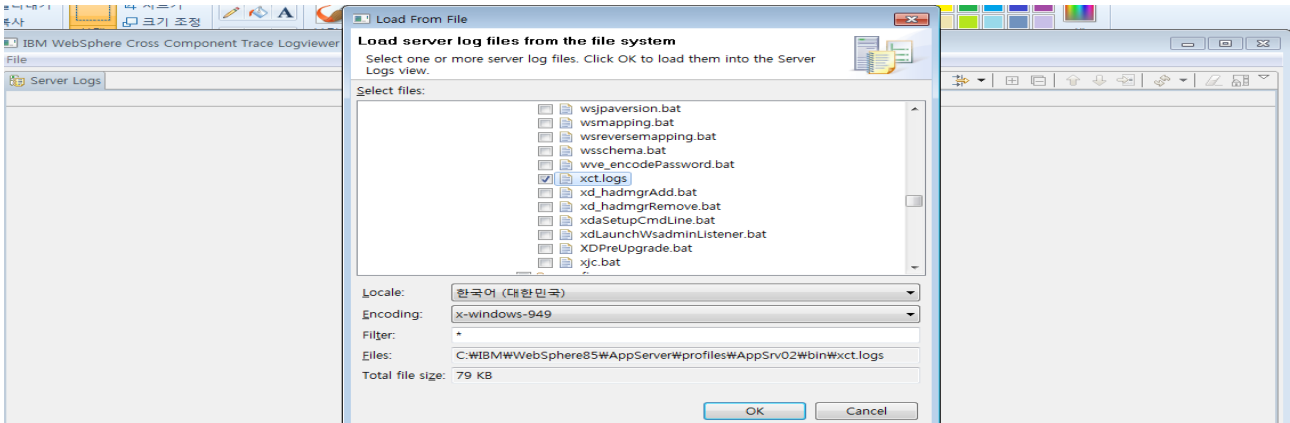
6. 해당 툴에서 IBM WebSphere Cross Component Trace Logviewer 라는 도구를 사용합니다.  
(ISA 의 가장 큰 장점은 원하는 분석 도구를 선택적으로 설치할 수 있으며 통합관리가 가능합니다.)



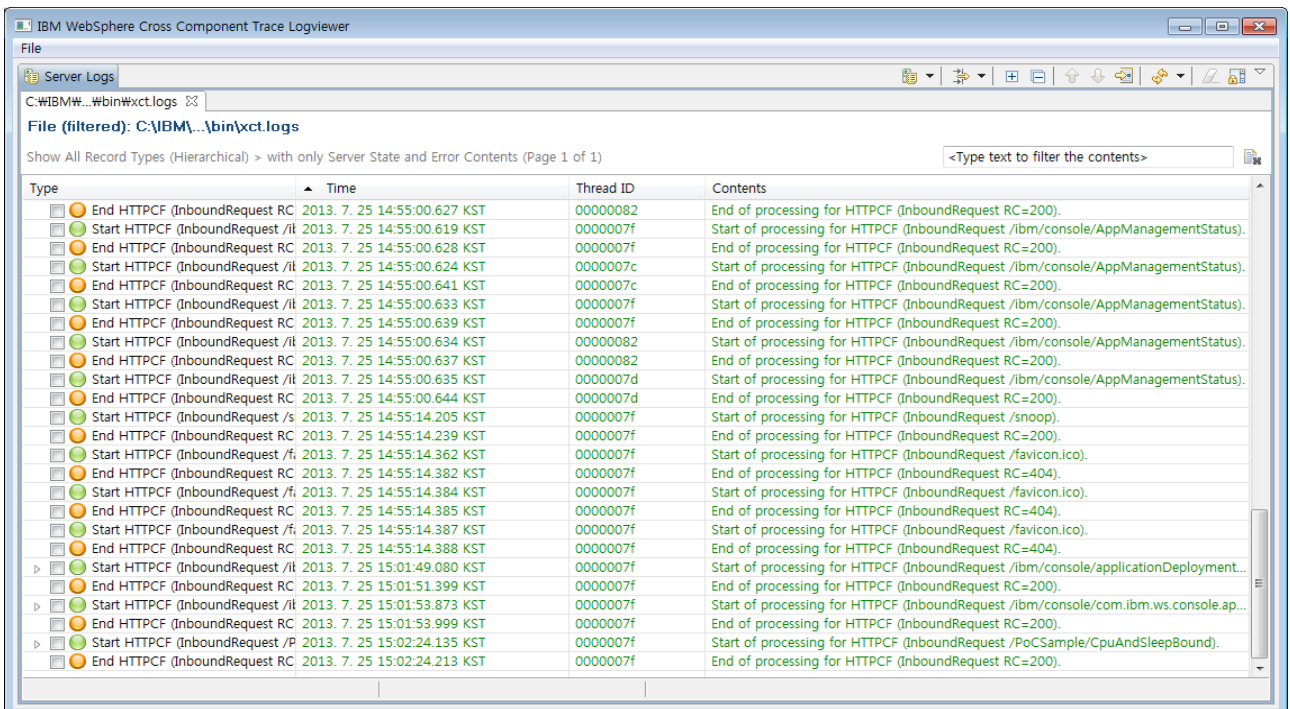
7. 해당 도구가 실행되면 "Load from File" 메뉴를 선택합니다.







8. "Load from File" 메뉴에서 방금 생성된 log 파일을 선택합니다.



9. 해당 log 파일을 선택하면 IBM WebSphere Cross Component Trace Logviewer 가 log 를 분석하기 편하게 표시해 줍니다.



10. 예를 들어 방금 호출한 요청에 대한 정확한 로그와(인바운드 요청 이름도 표시 /PoCSample/CpuAndSleepBound) XCT 의 연관 ID 로 그룹핑 되어 들어온 요청이 완료되고 나갈 때 까지의 정확한 log 정보를 하나의 그룹 형태로 확인 및 분석 가능합니다.

 Start HTTPCF (InboundRequest /P	2013. 7. 25 15:02:24.135 KST	0000007f	Start of processing for HTTPCF (InboundRequest /PoCSample/CpuAndSleepBound).
 Log message	2013. 7. 25 15:02:24.206 KST	0000007f	com.ibm.ws.webcontainer.servlet.ServletWrapper init SRVE0242I: [PoCSampleEAR] [/P.
 Log message	2013. 7. 25 15:02:24.206 KST	0000007f	conc = [1, 0.0, 1] at 2013-07-25 15:02:24.206 for java.lang.Object@a54fe886
 End HTTPCF (InboundRequest RC	2013. 7. 25 15:02:24.213 KST	0000007f	End of processing for HTTPCF (InboundRequest RC=200).



### 3) 결론

이번 강좌에서는 IBM WAS v8.5.5 의 로깅 기능에서 추가된 XCT 를 살펴보는 시간을 가져봤습니다. 해당 기능을 적절하게 활용하면 애플리케이션에서 별도 로깅 작업 없이도 WAS 차원에서 로그나 트레이스의 연관관계를 맵핑 및 로깅이 가능하므로 복잡한 애플리케이션에 대한 문제 분석을 좀 더 쉽게 하실 수 있습니다.

그럼 여기서 이만 이번 강좌는 마무리하고 다음 강좌에서 뵙도록 하겠습니다. Go Go !!!

## 9) 참고 자료

1. 이 가이드는 IBM WAS v8.5.5 최초 사용자를 위한 기본 가이드 입니다.
2. IBM WAS 자체에 아직 익숙하지 않으신 분들은 가급적 기본강좌인 '하나씩 쉽게 따라 해보는 IBM WAS v7' 강좌와 '제대로 맛보는 IBM WAS v8.5' 강좌를 먼저 읽고 이 강좌를 읽으시는 것이 이해에 훨씬 도움이 됩니다.  
([http://www.websphere.pe.kr/xe/?mid=was\\_info\\_re&page=3&document\\_srl=800](http://www.websphere.pe.kr/xe/?mid=was_info_re&page=3&document_srl=800)  
[http://www.websphere.pe.kr/xe/?mid=was\\_info\\_re&page=2&document\\_srl=134863](http://www.websphere.pe.kr/xe/?mid=was_info_re&page=2&document_srl=134863))
3. 가급적 IBM WAS v8.5 InfoCenter 의 해당 카테고리를 한 번 읽어보고 난 후에 작업하시기 바랍니다.
4. InfoCenter – WebSphere Application Server v8.5  
(<http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/index.jsp>)
5. InfoCenter – Cross Component Trace (XCT)  
([http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.express.doc/ae/ctrb\\_XCTOverview.html?resultof=%22%78%63%74%22%20](http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.express.doc/ae/ctrb_XCTOverview.html?resultof=%22%78%63%74%22%20))