|  |
| --- |
|  |
| **인젠트 ECM NewFace Toy Project**  **운영자 가이드 및 매뉴얼 이미지시스템 구축**  **RL-01** |
| |  |  | | --- | --- | | **업 무 명** | 이미지 구축 | | **작 성 일** | 2023. 06. 29 | | **버 전** | 1.0 | |

C:\Users\TY\AppData\Local\Microsoft\Windows\INetCache\Content.Word\inzent_logo_W.PNG

**Copyright Notice**

**Copyright ⓒ 2023 INZENT Co., Ltd. All Rights Reserved.**

**대한민국 서울 영등포구 국제금융로2길 유화증권빌딩 8층**

**Restricted Rights Legend**

**INZENT software and documents are made available under the terms of the INZENT License Agreement and may only be used or copied in accordance with the terms of this agreement. No part of this document may be transmitted, copied, deployed, or reproduced in any form or by any means, electronic, mechanical, or optical, without the prior written consent of INZENT Co., Ltd.**

**사용설명서의 내용과 여기에 설명된 프로그램은 INZENT Co., Ltd.와 사용권 계약 하에서만 사용이 가능하며, 사용권 계약을 준수하는 경우에만 사용 또는 복제할 수 있습니다. 이 사용설명서의 전부 또는 일부분을 INZENT의 사전 서면 동의 없이 전자, 기계, 녹음 등의 수단을 사용하여 전송, 복제, 배포, 2차적 저작물 작성 등의 행위를 하여서는 안 됩니다.**

**안내서 정보**

**발행일: 2023-06-29**

**소프트웨어 버전: XTORM6.0**

안내서 버전: XTORM 운영자 가이드 및 매뉴얼 1.0.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **문서 개정 이력표(Project Based)** | | | | |
| **Date** | **Version** | **Description** | **Writer** | **Etc.** |
| 2023. 06. 29 | 1.0 | XTORM 운영자 매뉴얼 작성 | ㈜인젠트 장재욱 | 최초 작성 |
| 2023. 06. 30 | 2.0 | SPRING BOOT 오류시 대처 방법 추가 | ㈜인젠트 장재욱 | 내용 추가 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

목 차

[1. XTORM (ECM 엔진) 주요 구성 5](#_Toc139026603)

[1.1 XTORM 서비스 구성 5](#_Toc139026604)

[1.2 XTORM 엔진 설치 디렉토리 구성 6](#_Toc139026605)

[1.3 XTORM 엔진 서비스 제어 6](#_Toc139026606)

[1.4 XTORM 엔진 실행 컨트롤러 9](#_Toc139026607)

[1.5 XTORM 엔진 서비스 구성 파일 12](#_Toc139026608)

[2. XTORM (ECM 엔진) 모니터링 14](#_Toc139026609)

[2.1 XTORM 엔진 서비스 로그 모니터링 14](#_Toc139026610)

[2.2 XTORM 엔진 API 문제 해결 15](#_Toc139026611)

[2.3 spring boot 문제 해결 17](#_Toc139026612)

[3. XTORM (ECM 엔진) 관리 정보 18](#_Toc139026613)

[3.1 프로젝트 사이트 정보 18](#_Toc139026614)

[3.2 서버 정보 18](#_Toc139026615)

[3.3 아카이브 · 볼륨 구성 정보 19](#_Toc139026616)

[3.4 엔진 정보 19](#_Toc139026617)

# XTORM (ECM 엔진) 주요 구성

본 장에서는 XTORM (ECM 엔진)에 대한 설치 고려사항 및 엔진 주요 구성에 대해 설명한다.

## XTORM 서비스 구성

- 이미지 엔진 구축 시 고려사항

XTORM 서비스를 구성하기 위해서는 다음과 같은 사항을 고려하여 구성해야 한다.

[표2-1] 엔진 설치 고려사항

|  |  |
| --- | --- |
| **항목** | **설명** |
| 서버 | * 동시 사용량 및 저장 용량 등을 고려하여 서버의 최소 사양 등을 고려해야 하며, XTORM 서비스를 구동하기 위한 JVM(Java Virtual Machine)이 기본적으로 탑재 되어야 한다. |
| Database | * XTORM은 서비스가 구동되기 위한 내부정보, 파일을 저장하기 위한 스토리지의 논리 및 물리 볼륨 정보 및 파일의 복제/이관 등을 관리하기 위한 Lifecycle 정보, 그리고 파일의 인덱스 및 위치정보 등을 데이터베이스에 저장한다. * XTORM Data Service를 다수로 운영하고자 한다면 데이터베이스 스키마도 그 개수에 맞추어 생성되어야 한다. |
| 엔지니어 관리 페이지 | * XTORM를 관리 및 모니터링 하기 위해서는 엔지니어 관리 페이지가 필요하다. 관리자만 사용하기 때문에 고성능의 엔진은 필요치 않으며 대부분 XTORM이 설치된 서버(혹은 다른 서버)에 tomcat을 탑재하여 사용한다. |
| 스토리지 | * 파일을 저장하기 위하여 필요한 물리적인 볼륨이며, 추후 확장성을 고려하여 마운트 해야 한다. 개별적인 물리적인 볼륨 크기는 스토리지 업체에서 제안하는 크기 혹은 백업 정책에 따라 정한다. * XTORM은 물리적인 볼륨이 큰 경우에 내부적으로 논리적으로 분할하여 사용하여 볼륨을 유연성 및 연속성을 보장한다. |
| Lifecycle  (보존 주기) | * 파일이 최초로 저장된 이후에 특정한 조건에 의하여 다른 스토리지로 복제/이관 작업 등이 필요한 경우에 이를 미리 정의하여 파일 입력 시 파일의 속성으로 지정해야 한다. |
| ElementId | * XTORM은 내부적으로 입력되는 파일에 ElementId라는 16자리의 키를 부여하여 관리한다. 이는 단일 XTORM Data Service 내에서 파일 단위로 Unique 한 값이며 파일을 다운로드, 수정, 삭제 등의 작업을 수행할 때 사용되는 키이다. |

## XTORM 엔진 설치 디렉토리 구성

[표2-2] 엔진 설치 디렉토리 구성

|  |  |  |
| --- | --- | --- |
| **서버** | **설치위치** | **설명** |
| **XTORM** | **/home/pj394053/xtorm** | XTORM홈디렉토리 |
| **/home/pj394053/xtorm/bin** | XTORM 서비스 control shell |
| **/home/pj394053/xtorm/lib** | XTORM library |
| **/home/pj394053/xtorm/log** | log파일 |
| **/home/pj394053/xtorm/msg** | 메시지 파일 |
| **/home/pj394053/xtorm/servers/** | XTORM 서비스 루트 디렉토리 |
| **/home/pj394053/xtorm/servers/50003** | XTORM Data 서비스 디렉토리 |

## XTORM 엔진 서비스 제어

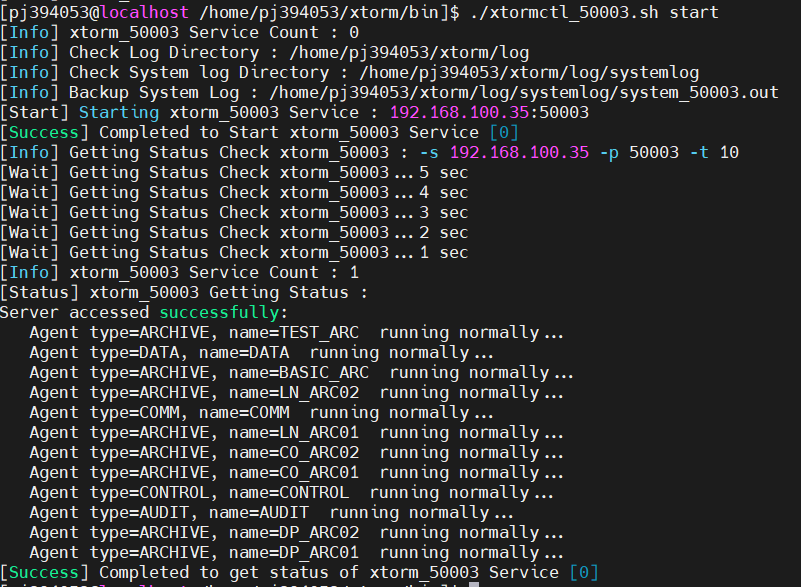
- XTORM 서비스 시작 및 종료

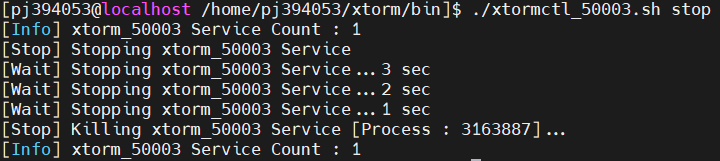
모든 XTORM 서비스들은 아래의 명령에 의해 시작 및 종료된다.

[표2-3] 엔진 설치 고려사항

|  |  |  |  |
| --- | --- | --- | --- |
| **서버** | **구분** | **명령** | **비고** |
| **Data Agent** | **시작** | **xstart -> /home/pj394053/xtorm/bin/xtormctl\_50003.sh start** | 사이트 특성에 따라 실행 쉘 경로 상이 할 수 있음 |
| **상태 체크** | **xstatus -> /home/pj394053/xtorm/bin/xtormctl\_50003.sh status** |
| **종료** | **xstop -> /home/pj394053/xtorm/bin/xtormctl\_50003.sh stop** |
| **프로세스 확인** | **ps –ef | grep xtorm | grep –v grep** |

[그림2-1] 엔진 기동 및 중지 화면 – **예시**



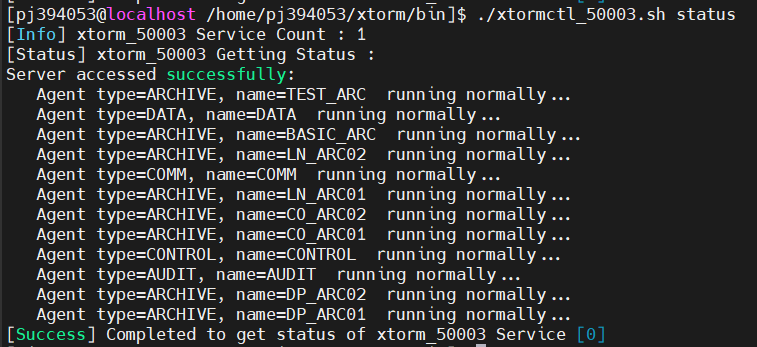


* **XTORM 엔진 서비스 기동 및 중지 화면**
* **고객사 특성에 따라 상이 할 수 있음.**

**- XTORM엔진 서비스 기동 및 중지**

START 혹은 STOP 명령어로 기동한다. 실행 및 중지 후 숫자 “0”이 Return 될 경우 정상작동 한 것이다.

명령어는 “[표2-3] 엔진 설치 고려사항” 참고

[그림2-2] 엔진 Status 화면 – **예시**

* **XTORM 엔진 서비스 상태 체크 화면**
* **고객사 특성에 따라 상이 할 수 있음.**

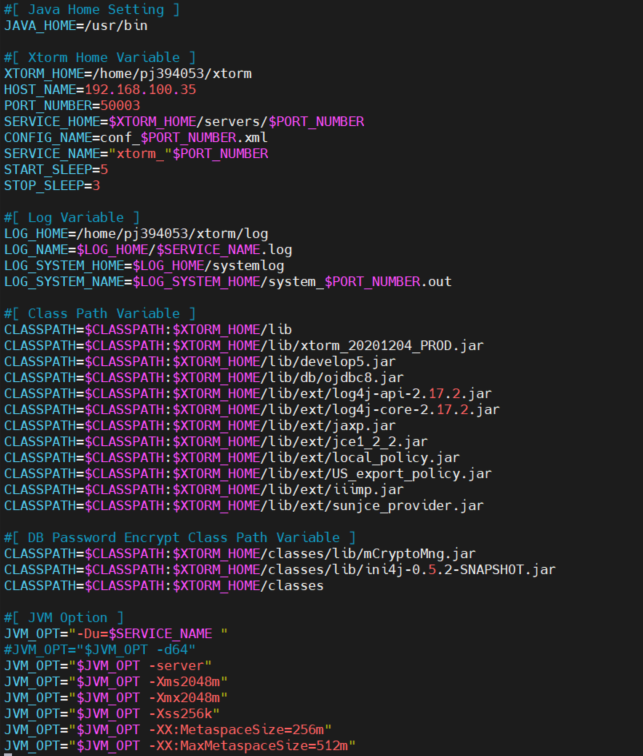
**- XTORM엔진 서비스 상태 체크**

STATUS 명령어로 Data 서비스에 상태를 체크한다. 위의 캡쳐 화면과 같이 모든 Agent 항목이 running normally……와 같이 화면에 출력이 되어야 정상적으로 XTORM 엔진 에이전트가 가동된 것이다.

명령어는 “[표2-3] 엔진 설치 고려사항” 참고

## XTORM 엔진 실행 컨트롤러

[그림2-3] 엔진 실행 컨트롤러 화면 – **예시**



* **엔진 서비스 xtormctl 파일 설정(JAVA HOME, 엔진 경로, CLASSPATH, JVM OPTION…)**
* **고객사 특성에 따라 상이 할 수 있음.**

**[설명]**

JAVA\_HOME과 XTORM 서비스 홈을 지정.

XTORM에 필요한 클래스 파일과 라이브러리 파일의 경로를 지정.

Du -50003: 사용하는 포트를 지정

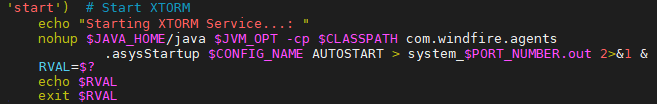
d64: XTORM 비트 수를 지정 (※해당 옵션은 JDK 11 이상에서 제거)

Xms1024m: 최소 Heap 사이즈를 지정, Xmx2048m: 최대 Heap 사이즈를 지정

Xss300k: 최대 Stack 사이즈를 지정

XX:MetaspaceSize: Metaspace Default Size 설정 (※해당 옵션은 JDK 1.8 이상에서 필수)

XX:MaxMetaspaceSize: Metaspace Max Size 설정 (※해당 옵션은 JDK 1.8 이상에서 필수)

****[그림2-4] 엔진 시작 스크립트 화면 – **예시**

* **고객사 특성에 따라 상이 할 수 있음.**

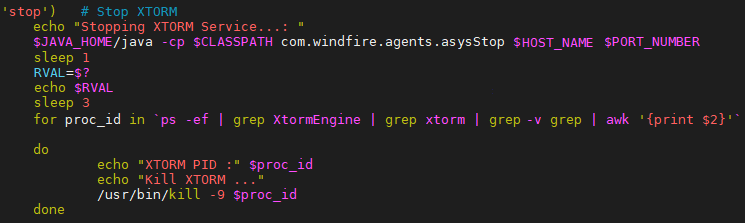
**[설명]**

Xtorm 엔진 수행하기 위한 스크립트이다. 명령어 sh xtormctl\_50003 start를 입력하면 구동된다.

엔진 구동 시 자동적으로 Systemout 로그가 생성된다.

자세한 명령어는 “[표2-3] 엔진 설치 고려사항” 참고

[그림2-5] 엔진 중지 스크립트 화면 – 예시

* **고객사 특성에 따라 상이 할 수 있음.**

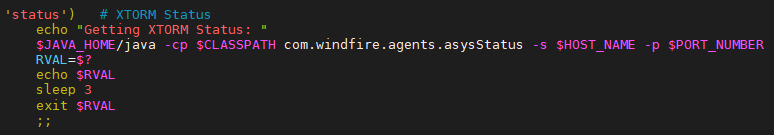
**[설명]**

Xtorm 엔진 정지를 위한 스크립트이다. 명령어sh xtormctl\_50003 stop를 입력하면 구동된다.

엔진이 정상적으로 중지 되지 않을 시, 프로세스 ID를 검색하여 Kill 명령어를 활용하여 중지시킨다.

자세한 명령어는 “[표2-3] 엔진 설치 고려사항” 참고

[그림2-6] 엔진 상태 스크립트 화면

****

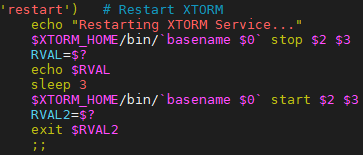
* **고객사 특성에 따라 상이 할 수 있음.**

**[설명]**

Xtorm 엔진 상태 확인을 위한 스크립트이다. 명령어sh xtormctl\_50003 status를 입력하면 구동된다.

명령어는 “[표2-3] 엔진 설치 고려사항” 참고

[그림2-7] 엔진 재 시작 스크립트 화면

****

* **고객사 특성에 따라 상이 할 수 있음.**

**[설명]**

Xtorm 엔진 재 시작을 위한 스크립트이다. 명령어sh xtormctl\_50003 restart를 입력하면 구동된다.

## XTORM 엔진 서비스 구성 파일

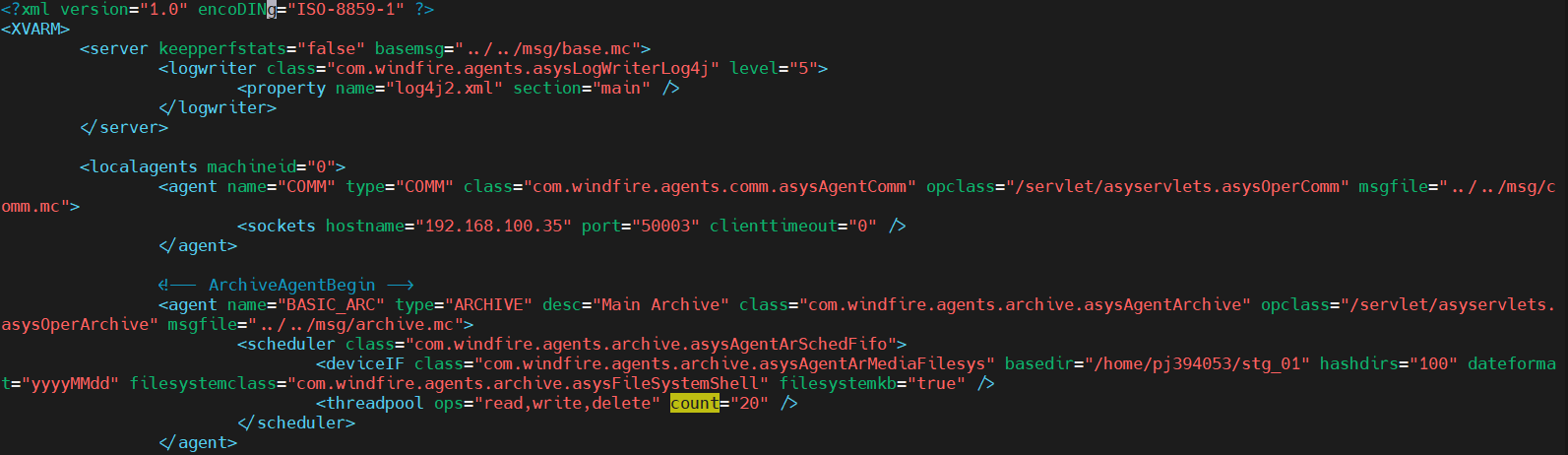
- XTORM Data 엔진 서비스 구성 파일

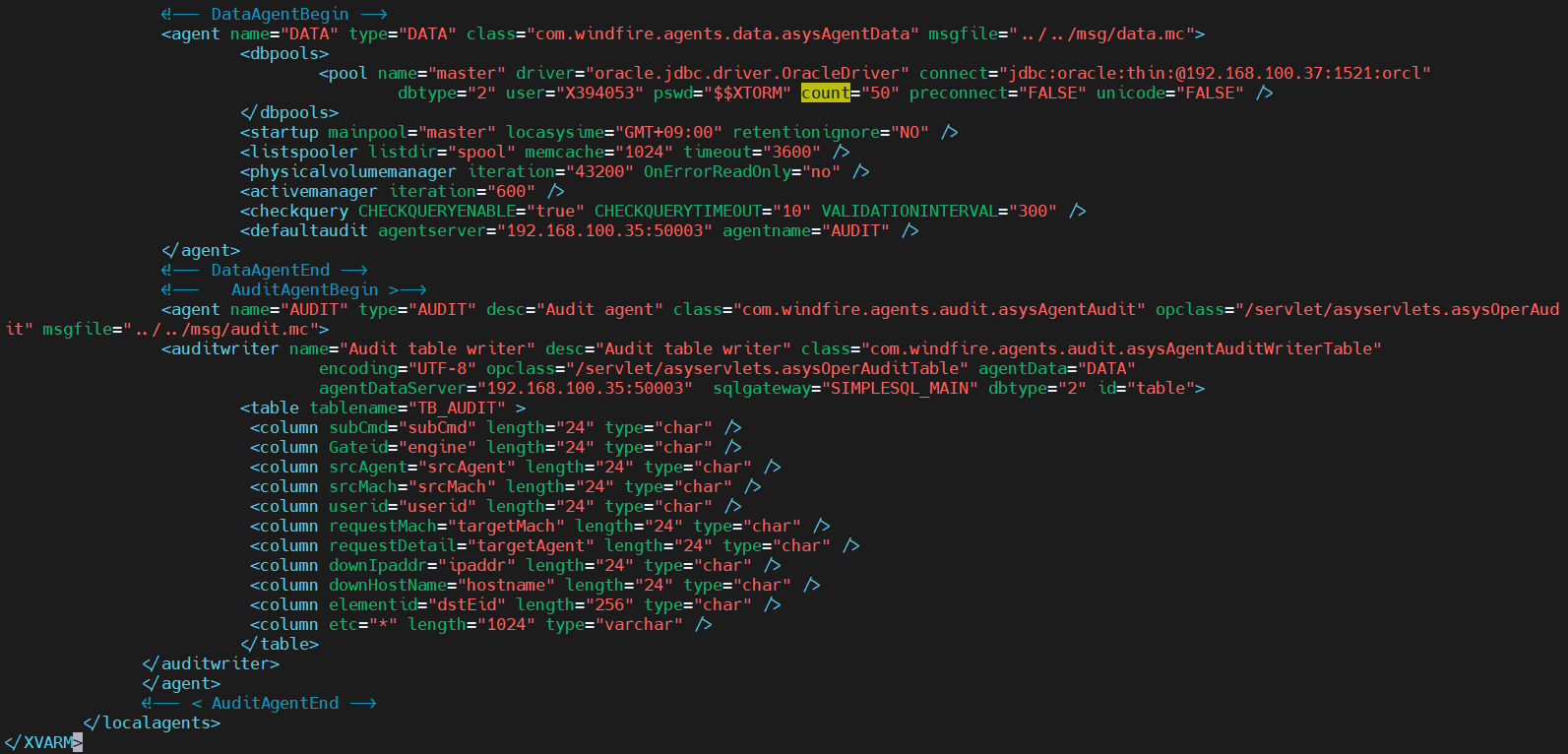
XTORM Data 엔진 환경 설정과 변경 및 추가 시 사용하는 엔진 환경 구성에 사용되는xml파일입니다.

로그 설정, 포트 설정, DB 설정, 아카이브 설정을 하는 XTORM환경 파일입니다.

명령어는 “[표2-2] 엔진 설치 디렉토리 구성” 참고

[그림2-8] 엔진 서비스 파일 구성 화면





**로그 설정 부분**

XTORM로그는 엔진 자체 로그 방식을 사용하며. 로그 레벨을 구분할 수 있도록 지원하며 전체 레벨 수준은 다음과 같이 6 가지로 구분한다.

운영중인 XTORM의 로그 레벨을 높게 잡을 경우 과도한 로그가 쌓일 수 있으므로 “2” 이하를 권장한다.

**[설명<logWriter>]**

“0” : ERROR - 에러내용만 출력

“1” : Normal Warning - 일반적인 경고 내용까지 포함하여 출력

“2“ : Detail Warning - 자세한 경고 내용까지 포함하여 출력

“3” : Normal Information - 일반적인 정보 내용까지 포함하여 출력

“4” : Detail Information - 자세한 정보 내용까지 포함하여 출력

“5” : DEBUG - 디버깅 용도로 모든 내용을 포함하여 출력

# XTORM (ECM 엔진) 모니터링

본 장에서는 XTORM(ECM 엔진)에 대한 로그 모니터링 및 API 문제 해결에 대해 설명한다.

## XTORM 엔진 서비스 로그 모니터링

**-** XTORM 서비스 로그 파일

에러 및 시스템 오류 등이 발생한 경우, XTORM 엔진의 로그를 통해 원인을 파악한다.

log파일은 10MB단위로 새로운 파일을 생성하게 된다.

10MB가 넘으면 기존에 있던 파일은 xtorm\_50003.log\_<년월일시분초> 로 명칭을 바꾸고 새로운 xtorm.log 파일이 생기게 된다.

Log파일이 커서 vi 편집기로 Open을 할 수 없을 경우에는 more xtorm\_50003.log 명령어를 이용해 Log 내용을 확인할 수 있다.

- XTORM System out 파일

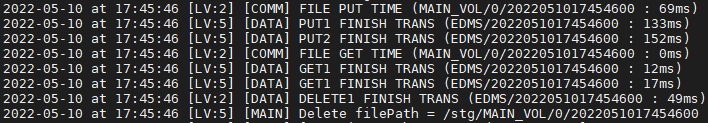
XTORM 기동 시 발생하는 오류에 대해서는 system\_<port>.out에서 확인 할 수 있다.

[표3-1] 로그 파일 위치

|  |  |  |
| --- | --- | --- |
| **서버** | **구분** | **로그파일 위치** |
| **XTORM 서버** | **Data/Archive Agent** | **/home/pj394053/xtorm/log/xtorm\_50003.log** |
| **System log** | **/home/pj394053/xtorm/log/systemlog/system\_50003.out** |

- 로그 레벨 5의 경우

[그림3-1] 로그 레벨 5 화면



- 로그 레벨 2의 경우

[그림3-2] 로그 레벨 2 화면

****

**[설명]**

FILE GET TIME:

클라이언트가 XTORM 엔진으로 조회가 일어날 경우 해당 로그가 Write 됩니다.

FILE PUT TIME:

클라이언트가 XTORM 엔진으로 파일을 등록할 경우 해당 로그가 Write 됩니다.

DELETE1 FINISH TRANS

클라이언트가 XTORM 엔진으로 등록된 파일을 삭제 요청 할 경우 해당 로그가 Write 됩니다.

## XTORM 엔진 API 문제 해결

- XTORM API 에러 코드

XTORM 엔진에서 저장하는 에러 로그는 대부분 API를 통한 거래에 의해 발생된 것이기 때문에 해당 API의 상세 오류 메시지로 에러 내용이 반환됩니다.

[표3-2] XTORM API 에러 코드

|  |  |  |
| --- | --- | --- |
| **코드** | **내용** | **비고** |
| 0 | 정상 | 정상 |
| 3 | 엔진 발생 오류 | API 상세 로그 메시지 확인 |
| 5 | 파라미터 오류 |
| 10 | 커넥션 연결 오류 |
| 11 | 커넥션 타임아웃 오류 |

**API 리턴 코드는 위와 같다. API 호출 후 0 이 아닌 경우 전부 오류 이며 API 사용시, getLastError()를 통해 상세 메시지 받아서 오류 확인 해야 한다.**

- XTORM API 에러 메시지

XTORM 엔진에서 발생하는 에러 로그는 엔진 자체 로그부터 DB에서 발생한 오류 등 매우 다양하므로 정확한 진단을 위해서는 XTORM 엔지니어 혹은 유지보수 담당자에게 문의해야 한다.   
XTORM 엔진에서 발생하는 오류 중 가장 빈번히 발생하는 유형은 다음과 같다.

[표3-3] XTORM API 에러 유형

|  |  |  |
| --- | --- | --- |
| **에러 메시지** | **내용** | **조치 방안** |
| [308] [COMM] Communications agent no longer listening on an incoming socket (Socket closed). | COMM 에이전트 기동 종료로 소켓 닫힘 | XTORM 에이전트 재 기동 필요 |
| [6717] [DATA] Error (Error creating local file in file transfer: "FILE\_PATH"  (No such file or directory (errno:2))) retrieving content to a client file from archive "ARCHIVE". Other archives will be tried. | 파일 등록 경로 비 정상 (경로 부적절 또는 해당 경로의 쓰기 권한 문제) | 등록 파일 경로 및 권한 여부 확인 필요 |
| [6714][DATA]Element "elementid" not found in the database | 데이터베이스에 해당 “Elementid” 존재하지 않음 | ElementId 확인 및 Xtorm DB 확인 필요 |
| [6716] [DATA] There are no available archives that can access content for element "elementid", last local path "FILE\_PATH". | 파일 다운로드 실패 | 다운로드 경로 및 쓰기 권한, Archive 목록 확인, 로그 확인 후 조치 필요 |
| [6428] [DATA] A connection to the database has been lost. Closing connection and retrying transaction... | 트랜잭션 도중 DB 연결 끊김 또는 DeadLock 발생 | DB 상태 확인 필요 |

## Spring boot 문제 해결

- Spring boot 에러

웹 화면을 이용하다 보면 발생하는 몇가지 에러에 대한 정리를 다음과 같이 했다.

[표3-4] Spring boot 에러 유형

|  |  |  |
| --- | --- | --- |
| **에러 메시지** | **내용** | **조치 방안** |
| [Request processing failed; nested exception is org.springframework.dao.DataIntegrityViolationException:  ### Error updating database. Cause: java.sql.SQLIntegrityConstraintViolationException: ORA-01400: NULL을 업무 테이블 안에 삽입할 수 없습니다 | 업무에 이미지 등록 시 사원코드, 고객명, 고객 주민번호를 입력하지 않았을 때 발생하는 오류 | 이미지를 등록할 때 사원코드, 고객명, 고객 주민번호 모두 입력 |
| [Request processing failed; nested exception is org.springframework.dao.DataIntegrityViolationException:  ### Error updating database. Cause:  java.sql.SQLIntegrityConstraintViolationException: ORA-01400: NULL을 EMPLOYEE\_TB안에 삽입할 수 없습니다 | EMPLOYEE\_TB의 사번과 이름이 들어가지 않았을 때 발생하는 오류 | 사용자 등록 시 사번과 이름은 무조건 입력 |

## XTORM Migration(이관) 관리 정보

XTORM에 입력된 파일은 XTORM Migration Component에 의하여 저장소 간, 서버간 파일을 자동으로 복제/이동 관리를 한다. 이는 입력 시에 부여하는 Lifecycle Id(혹은 Content Class Id)에 의해 저장 룰이 지정되며 특정시간 혹은 특정 조건에 따라 XTORM Scheduler Component통해 작업이 수행된다.

[그림2-1] Lifecycle 개념도 예시



**[Migration설명]**

[그림2-1]의 예시와 같이 “이미지 저장 룰” 이라는 Lifecycle이 특정 파일에 지정되었을 때, 최초 입력 시 이미지 저장소에 저장된다. 특정 기간 후에 Archive 저장소에 복제되고, 30일 경과 시 이미지 저장소에서 지워지고 Archive 저장소에 복제된다.

Archive저장소는 같은 이미지 서버만 아니라 원격지 이미지 서버로 구성이 가능하다.

**[복제(백업)/이동 설명]**

XTORM 에 저장된 파일은 XTORM Data Service가 통합관리하고 있는 저장소(archive)간에 파일을 복제/이동할 수 있다. 특정한 상태를 갖는 XTORM Content State를 만들어 원하는 저장소를 지정하고 이를 Lifecycle에 설정하는 것이다.

**[Lifecycle 기준]**

기간 베이스 도는 컨텐츠의 중요도에 따라 기준 설정

[그림2-2] Lifecycle 관리 정책 예시

**컨텐츠 관리 정책  
Content Class**

**최초 입력 시**

**1~2시간 내**

**2시간 경과 시**

**1차 스토리지 생성**

**1, 2차 스토리지**

**1차 스토리지 삭제**

[그림2-3] Lifecycle 관리 흐름 예시

**1차 스토리지**

**1차 스토리지 생성지**

**2차 스토리지**

**복사**

**1~2시간 내**

**최초 입력 시**

**2시간 경과 시**

**1차 스토리지 유지**

**2차 스토리지 생성**

**1차 스토리지 삭제**

**2차 스토리지 유지**

* 업무별 Content 분류 정보

[표2-8] Content Class 정보

|  |  |  |
| --- | --- | --- |
| **Content Class** | **업무명** | **설명** |
| CO\_CC | 공통 | 공통 업무에 파일 및 정보 등록 관리 |
| DP\_CC | 수신 | 수신 업무에 파일 및 정보 등록 관리 |
| LN\_CC | 여신 | 여신 업무에 파일 및 정보 등록 관리 |

- 업무별 Content 라이프사이클 정보

[표2-9] Content State 정보

|  |  |  |
| --- | --- | --- |
| **Content Class** | **목적** | **설명** |
| CO\_CC | 초기 생성 | 초기 생성시 아카이브에 파일 생성(1차 스토리지 보관) |
| 복제 | 1 ~ 2시간 보관 (1차, 2차 스토리지 보관) |
| 이관 | 2시간 경과 시 (2차 스토리지 보관) |
| DP\_CC | 초기 생성 | 초기 생성시 아카이브에 파일 생성 (1차, 2차 스토리지 보관) |
| 이관 | 1시간 경과 시 (2차 스토리지 보관) |
| LN\_CC | 초기 생성 | 초기 생성 시 아카이브에 파일 생성 (1차 스토리지 보관) |
| 이관 | 1시간 경과 시 (2차 스토리지 보관) |

## XTORM Scheduler 관리 정보

**-** Scheduler 설정 정보

[표2-10] Scheduler 정보

|  |  |  |  |
| --- | --- | --- | --- |
| **Scheduler ID** | **유형** | **DATA 서버** | **스케줄러 설정 설명** |
| DELETE\_BAT |  | XTORM | 매일 18:30분 시작 |

# XTORM (ECM 엔진) 관리 정보

본 장에서는 XTORM(ECM 엔진)을 활용한 프로젝트 내에서 XTORM을 관리하기 위한 정보를 설명한다..

## 프로젝트 사이트 정보

[표4-1] 프로젝트 사이트 정보

|  |  |
| --- | --- |
| **항목** | **내용** |
| 프로젝트 명 | ECM 사업팀 Toy Project |
| 프로젝트 수행 기간 | 2023.05.08 ~ 2023.06.30 |
| 담당자 | 장현식 수석, 성기명 수석 |
| 엔지니어 | 장재욱 매니저 / 010-5349-2526 / jaewook21121@inzent.com |
| 엔진 버전 및 JAR | XTORM 6.0 / xtorm\_20201204\_PROD.jar |

## 서버 정보

[표4-2] 서버 정보

|  |  |  |
| --- | --- | --- |
| **서버** | **항목** | **내용** |
| 서버 | OS | Rocky Linux release 8.7 |
| IP | 192.168.100.35 |
| User (sudo 권한 여부) | pj394053 (N) |
| Password | - |
| JDK Version | openJdk 11.0.18 |
| 접속 방안 | Vpn 접속 후 ssh 접속 |

## 아카이브 · 볼륨 구성 정보

- 서버 아카이브 · 볼륨 구성 정보

[표4-3] 운영서버 아카이브 · 볼륨 구성 정보

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **업무** | **스토리지** | **아카이브** | **볼륨** | **용량** | **비고** |
| 공통 | - | CO\_ARC01 | CO\_VOL01 | 420G |  |
| 공통 | - | CO\_ARC02 | CO\_VOL02 | 420G |  |
| 수신 | - | DP\_ARC01 | DP\_VOL01 | 420G |  |
| 수신 | - | DP\_ARC02 | DP\_VOL02 | 420G |  |
| 여신 | - | LN\_ARC01 | LN\_VOL01 | 420G |  |
| 여신 | - | LN\_ARC02 | LN\_VOL02 | 420G |  |

## 엔진 정보

[표4-4] 엔진 운영 서버 정보

|  |  |  |
| --- | --- | --- |
| **서버** | **항목** | **내용** |
|  | IP ( Domain ) | 192.168.100.35 |
| PORT | 50003 |
| 엔진 경로 | /home/pj394053/xtorm |

- API 배포 정보

XTORM API를 통해 엔진에 접근하는 모든 WAS 및 Client 프로그램에 대한 정보를 기술한다.

[표4-11] API 배포 정보

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **서버(hostname)** | **역할** | **API버전** | **사용메소드** | **비고** |
| pj394053 | WAS | xtormapi\_20210723.jar | create, download, delete |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |