

XTORM 환경 구성 가이드

Table of contents

| | |
|--|----|
| 개요 | 3 |
| 기본구성 | 3 |
| 루트 엘리먼트 - Xtorm | 3 |
| 로컬 에이전트 | 4 |
| Transmit 에이전트 (유형: COMM) | 7 |
| 아카이브 에이전트 (유형 : ARCHIVE) | 9 |
| asysAgentArMediaFilesys | 11 |
| asysAgentArMediaEMCCentera | 13 |
| asysAgentArMediaEMCCenteraBatch | 14 |
| asysAgentArMediaEMCCenteraBatchCache | 15 |
| asysAgentArMediaEMCCenteraBatchCacheShred | 18 |
| asysAgentArMediaEMCCenteraBatchCacheDirect | 19 |
| asysAgentArMediaDav | 20 |
| asysAgentArMediaWebDavBatchCacheDirect | 22 |
| asysAgentArMediaWebDavBatchCache | 22 |
| asysAgentArMediaHTTP | 24 |
| asysAgentArMediaHCP | 26 |
| asysAgentArMediaDataChian | 29 |
| asysAgentArMediaAvamar | 30 |
| asysAgentArMediaSnapLock | 32 |
| EMC Centera | 34 |
| HIS HCP | 37 |
| 필터 사용법 | 39 |
| 서버 | 42 |
| Log4설정 | 45 |
| Audit 에이전트 (유형: AUDIT) | 46 |
| Cache Cleanup 에이전트 (유형: CACHECLEAN) | 54 |
| IMPORT 에이전트 (유형: IMPORT) | 56 |
| 정보/데이터 에이전트 (유형: DATA) | 70 |
| DB패스워드 암호화 가이드 | 81 |
| 마이그레이션 에이전트 (유형: MIGRATION) | 83 |
| 스케줄러 에이전트 (유형: SCHEDULER) | 86 |
| 중첩 엘리먼트 카테고리 – 서버 리스트 | 88 |
| Configuration File 예제 | 89 |
| 프로로그 | 96 |

개요

본 문서는 Xtorm 서버 및 에이전트들에 대한 기본 구성 방법을 설명합니다.

분산환경에서 Xtorm 서비스를 구성하기 위해 단위 서버, 또는 에이전트들이 갖추어야 하는 환경파일의 정의에 대하여 설명합니다.

이러한 기본 구성은 Xtorm 서버가 설치되어 있는 홈 디렉토리 밑에 있는 "conf.xml" 파일에 XML 형식으로 저장됩니다. 환경 파일명은 관리자에 의해 임의로 지정될 수 있습니다.

기본 설치 경로는 다음과 같습니다.

- UNIX: "/xtorm"

"conf.xml" 파일의 기본 설치 디렉토리:

- UNIX: "servers/portnumber"

결론적으로, 기본 경로를 그대로 사용할 경우, 서버명과 환경구성 파일은 다음 위치에서 찾을 수 있습니다.

- UNIX: "/xtorm/servers/2102"

기본구성

다음은 환경파일을 구성하는 XML 태그(tag) 별로 각각의 속성이 의미하는 내용을 정리 하였습니다.

경우에 따라서 환경파일을 구성하는 XML 태그(tag)와 하위의 속성명을 이해하기 쉽도록 ‘. ’ 로 구분 하였습니다.

루트 엘리먼트 - Xtorm

루트 엘리먼트는 핵심 구성 요소입니다. 루트 엘리먼트 아래에 Xtorm 전체에 걸친 구성을 정의합니다. 다음과 같이 크게 3 가지의 내부 카테고리 구성됩니다: 서버(server), 로컬 에이전트(local agents), 서버 리스트(server list).

Xtorm

현재 서버(컴퓨터)에 설치되어 있는 Xtorm을 정의합니다.

예:

<Xtorm>

로컬 에이전트

"LocalAgents"는 현재의 Xtorm 서버에 설치되어 있는 모든 에이전트에 대한 일반적인 정보를 정의합니다.

속성

localagents

서버에서 운영 중인 각각의 에이전트 및 에이전트 기본정보를 정의

localagents.timediff

엔진 로케일(Locale) 설정, "-Duser.timezone=GMT+09:00" 자바 실행 옵션과 동일한 기능을 수행하며 분단위로 설정합니다.

예: "540"분

localagents.machineid

별도의 프로세스로 동작하는 엔진 인스턴스 별로 Elementid가 중복되지 않게 생성하도록 하기 위하여 고유의 아이디를 지정합니다. 이때 아이디값은 한자리 숫자로 지정되어야 하며 Elementid의 맨 첫자를 대체하여 사용됩니다.

예: 2 + machineid 로 Elementid 시작 숫자임.
"1" 일 경우, Elementid 는 3으로 시작 되도록 채번됨.

agent

각각의 에이전트에 대한 개별 정보 정의합니다. 각각의 에이전트에 대하여 복수로 지정이 가능합니다.

agent.name

서버 시작 시에 시작될 에이전트의 이름. COMM 및 데이터 에이전트의 경우 각각 반드시 COMM 및 DATA 이름을 사용해야 하며, 이 두 에이전트는 conf.xml 파일 안에 하나씩만 존재해야 합니다. 아카이브 에이전트 이름은 서버 전체에 걸쳐 고유해야 하며 IMPORT 에이전트 이름은 한 서버 안에서 고유해야 합니다.

agent.type

에이전트 유형. 웹 운영 어플리케이션은 각각의 에이전트 유형을 보고 화면에서 아이콘으로 구분 표시합니다.

ARCHIVE : 아카이브 에이전트 유형

| | |
|--------------|-----------------------|
| AUDIT : | Audit 스케줄러 에이전트 유형 |
| CACHECLEAN : | Cache Cleanup 에이전트 유형 |
| COMM : | Transmit 에이전트 유형 |
| DATA : | 데이터 에이전트 유형 |
| IMPORT : | Import 에이전트 유형 |
| MIGRATION : | 마이그레이션 에이전트 유형 |
| SCHEDULER : | 스케줄러 에이전트 유형 |

agent.desc

에이전트에 대한 일반적인 설명. ARCHIVE, IMPORT, CACHECLEAN, MIGRATION 및 SCHEDULER 에이전트에만 유효함

agent.class

각각의 에이전트가 시작될 때 사용될 자바 클래스 이름

```

ARCHIVE : com.windfire.agents.archive.asysAgentArchive
AUDIT : com.windfire.agents.audit.asysAgentAudit
CACHECLEAN : com.windfire.agents.cache.asysAgentCache
COMM : com.windfire.agents.comm.asysAgentComm
DATA : com.windfire.agents.data.asysAgentData
IMPORT : com.windfire.agents.importagent.asysAgentImport
MIGRATION : com.windfire.agents.migration.asysAgentMig
SCHEDULER : com.windfire.agents.scheduler.asysAgentSched

```

agent.opclass

웹 운영 어플리케이션에서 각각의 에이전트를 표시할 서블릿 이름

```

ARCHIVE : "/servlet/asysservlets.asysOperArchive"
AUDIT : "/servlet/asysservlets.asysOperAudit"
CACHECLEAN : "/servlet/asysservlets.asysOperCache"
COMM : "/servlet/asysservlets.asysOperComm"
IMPORT : "/servlet/asysservlets.asysOperImport"
MIGRATION : "/servlet/asysservlets.asysOperMig"
SCHEDULER : "/servlet/asysservlets.asysOperSched"

```

참고: 데이터 에이전트의 경우 관련된 서블릿은 없습니다. 데이터 에이전트는 게이트웨이, 보안 관리자, 포스트 프로세서 등으로 구성되어 있습니다. 또한, 각각의 하위 요소는 자체 서블릿을 가지고 있습니다.

agent.msgfile

각각의 에이전트가 사용할 메시지 기본 파일을 지정합니다.

```

ARCHIVE : "msg/archive.mc"
AUDIT : "msg/audit.mc"
CACHECLEAN : "msg/cache.mc"
COMM : "msg/comm..mc"
DATA : "msg/data.mc"
IMPORT : "msg/import.mc"
MIGRATION : "msg/migration.mc"
SCHEDULER : "msg/scheduler.mc"

```

agent.minhealth

이는 Xtorm의 Fail Over 기능으로 각각의 에이전트가 유효한지 여부를 나타내는 최소 값을 지정합니다.

ARCHIVE:

- 1 OK
- 5 실패, 미디어에 접속할 수 없음

AUDIT:

- 1 OK
- 3 최소 하나 작성기를 시작할 수 없음
- 5 작성기를 모두 시작할 수 없음

CACHECLEAN:

- 1 OK
- 5 실패, 캐시 리스트 에러(잘못된 캐쉬 ID)

COMM:

- 1 OK
- 5 실패, 리스닝(Listening) 응답 없음

CONTROL:

- 1 OK
- 4 로그 작성기 에러
- 5 실패, 에이전트 시작 에러

DATA:

- 1 OK
- 5 실패, 데이터베이스 연결 에러

IMPORT:

- 1 OK
- 5 실패, 스케줄링 실행 쓰레드 실패
- 5 실패, 스케줄링 실행 쓰레드를 찾을 수 없음

MIGRATION:

- 1 OK
- 5 실패, 마이그레이션 리스트 에러

SCHEDULER:

- 1 OK
- 2 쓰레드 풀 관리자에 의해 관리되는 하나 또는 하나 이상의 쓰레드가 실패한 부분적인 실패
- 5 Import 게이트웨이가 실패함
- 5 Import 게이트웨이가 없음
- 5 Bundler 쓰레드가 실패함
- 5 Bundler 게이트웨이가 없음
- 5 Reimport 쓰레드가 실패함
- 5 Reimport 게이트웨이가 없음
- 5 쓰레드 풀 관리자 쓰레드가 실패함
- 5 쓰레드 풀 관리자에 의해 관리되는 모든 쓰레드가 실패함
- 5 쓰레드 관리자가 없음

agent.debug

에이전트 단위로 'debugger'를 설정할 수 있도록 지원 합니다. 해당 파라미터에 debugger ID를 지정하면 해당 'debugger'에 디버깅 정보를 기록합니다.

(대상 에이전트 유형: ARCHIVE)

Transmit 에이전트 (유형: COMM)**속성****sockets**

현재의 서버에서 Transmit 에이전트에 의해 사용될 통신 소켓 정의

hostname

호스트명이나 IP로 소켓을 Open 할 수 있도록 지원

port

Transmit 에이전트가 에이전트 상호간 또는 어플리케이션 간에 통신하는 통신 포트 번호

clienttimeout

일정 시간이 경과하도록 클라이언트 연결이 아무런 작업을 하지 않는 경우(IDLE), 자동으로 이 클라이언트의 연결을 닫도록 하는 파라미터.

이 기능은 클라이언트에 전적으로 투명하게 적용되며, 일단 서버 측에 타임아웃이 발생하여 클라이언트 접속이 끊겼다 해도 해당 클라이언트 어플리케이션이 재 시도를 하면 자동으로 연결이 재개되어 업무의 연속성을 유지할 수 있음.

기본 값: "0" (분 단위)

sotimeout

소켓을 통한 read() 작업 수행 시 응답이 없을 경우, java.net.SocketTimeoutException 을 발생시켜 예외상황으로 인한 무한정 대기 상황을 회피하기 위한 파라미터.

기본값: "0" (분단위)

maxbuff

클라이언트가 서버에 접속하게 되면 요청한 작업이 모두 처리될때까지 작업큐에 머무르게 되는데 서버가 감당할 수 있는 처리량에는 한계가 있기 때문에 안정적으로 서비스를 제공할 수 있도록 작업큐에 대기하는 클라이언트의 요청 개수를 제한하는 기능을 수행.

기본값: "5000"

virtualhost

대표 IP 접속을 허용하기 위해 대표 호스트 명을 지정합니다.

virtualip

대표 IP 접속을 허용하기 위해 대표 IP 주소를 지정합니다.

virtualport

대표 IP 접속을 허용하기 위해 대표 서비스 포트 번호를 지정합니다.

debug

디버깅 용도로 Transmit 에이전트를 통해 처리되는 전문 내용을 캡처하여 실행 경로 하위에 저장할지의 여부를 지정합니다.

기본값: "NO"

filepermission

등록 및 조회할 경로나 파일에 대한 권한이 없을 경우 권한을 체크하여 권한을 부여함.(소유계정의 파일 및 디렉토리만 가능함)

기본값: "NO"

checkfilesize

실제 등록요청한 파일사이즈와 저장된 파일사이즈 비교 하여 동일할 경우에만 성공 리턴.

기본값: "FALSE"

license

라이센스 파일에 대한 경로 지정(미지정시 설치경로)

기본값: ""

filepermission

조회 및 파일 등록시 해당 파일 및 폴더에 권한 체크 하여 권한 없을 경우 권한 추가(계정 소유자만 가능함 .)

값: "YES"

속성**throttle**

목적지의 Xtorm 서버의 정보와, 이 서버와 목적지 서버 간에 사용할 네트워크 전송 비율을 정의합니다. 속도 조절 기능은 파일 전송에만 적용됩니다. 데이터베이스와의 일반 통신, 결과 리스트 또는 파일 전송이 포함되지 않은 모든 통신에는 이 속도조절 기능이 영향을 미치지 않습니다. 여러 라인에 걸쳐 속도 조절 기능을 추가할 수 있습니다. 예를 들어, 각각 다른 컴퓨터 상에서 운영중인 3 개의 Xtorm 서버 즉, DataServer, LocalStorage 및 RemoteStorage가 있다고 가정합니다. 엘리먼트가 생성될 때 실제 파일은 LocalStorage에 저장됩니다. LocalStorage 서버와 RemoteStorage 서버 간의 마이그레이션에 속도 조절 기능을 추가하고 싶으면 LocalStorage 서버의 환경 구성 파일의 COMM 에이전트 부분에 throttle 부분을 추가 하십시오. 목적지 서버는 RemoteStorage가 될 것이며 사용 예는 다음과 같습니다

* Xtorm은 컴포넌트 간에 네트워크 속도 조절(Throttling) 기능

Xtorm의 HPI 및 마이그레이션과 같이 파일을 다루는 모든 컴포넌트에 의해 발생하는 파일 전송에 관련된 모든 네트워크 트래픽에 관련된 기능입니다. Xtorm이 Xtorm 자체 모듈로부터 독립적으로 속도 조절 기능을 제공하기 위하여 통신 컴포넌트에 이 기능이 추가되었습니다. 이러한 방법으로 알라이언스시스템 또는 알라이언스시스템의 파트너에 의해 개발될 미래의 어플리케이션은 속도 조절 기능의 장점을 활용할 수 있습니다.

컴포넌트 간에 기본 통신 방법을 보면 거기에는 보내는 컴포넌트와 받는 컴포넌트가 있습니다. 속도 조절 기능은 보내는 측의 Xtorm 구성 파일에 정의됩니다. 목적지별로 속도 조절기능을 지정하기 위하여 Xtorm 구성 파일의 통신 에이전트 부분에 새로운 부분이 추가 되었습니다. 통신을 보내는 Xtorm 서버/컴포넌트는 목적지 또는 목적지 서버에 지정된 최대 비율에 따라 정보를 전송합니다. 동시에 여러 라인에 걸쳐 속도 조절 기능을 추가할 수 있으며 각각 라인별로 전송 비율을 달리할 수 있습니다. 속도조절 기능을 사용하는 실제 알고리즘은 10 초 동안 평균 회전 비율을 추적하여 특정 주어진 시간에는 이 비율이 maxrate 값 보다 실제적으로 높을 수 있지만 결과적으로는 장기간 동안의 평균 전송은 원하는 비율대로 이루어질 것입니다. 다음은 새로 추가된 구성 부분의 속성에 대한 설명입니다.

예: <throttle target="RemoteStorage:2104" maxrate="300"/>

target

정보를 받을 목적지 Xform 서버를 정의합니다. 목적지는 주로 컴퓨터:포트번호(MACHINE:PORT)로 표현됩니다. 컴퓨터는 컴퓨터의 네트워크 이름 또는 IP 주소로 지정할 수 있으며 포트번호는 이 서버의 Xform 이 사용하는 리스닝 포트번호를 의미합니다. 이 서버와 연결되어 있는 모든 목적지 Xform 서버와의 통신에 속도 조절 기능을 사용하고자 할 경우에는 "*" (Asterisk)를 사용할 수 있습니다.

maxrate

초 당 전송될 최대 킬로바이트(Kb) 수를 정의합니다 (Kb/sec). 이 숫자는 네트워크 성능 및 실제적인 여러 번의 테스트를 통하여 계산되어야 합니다.

예:

```
<agent name="COMM" type="COMM"
class="com.windfire.agents.comm.asysAgentComm" opclass="/serMet/asyserMets.asysOperComm"
msgfile="../../msg/comm.mc" >
  <sockets hostname="10.1.61.99" port="3102" debug="NO" clienttimeout="10"
FILEPERMISSION="NO" LICENSE="/home/xtorm/lic"/>
  <throttle target="RemoteStorage:2104" maxrate="300"/>
</agent>
```

아카이브 에이전트(유형: **ARCHIVE**)

속성

scheduler

아카이브 에이전트의 특정 기능과 관련된 정보를 정의합니다.

scheduler.class

장치를 읽고 쓰는 중에 사용될 알고리즘을 포함하고 있는 자바 클래스 이름. 현재는 First In First Out (FIFO) 방식만을 지원하지만 향후 다른 방식 지원 예정

기본 값: "com.windfire.agents.archive.asysAgentArSchedFifo"

scheduler.iterationop

EMC 또는 Webdav용 아카이브의 경우와 같이 임시 폴더(캐시)를 운용하는 경우, GET 임시 폴더 하위의 파일들을 삭제해 주어야 하는데, 이 파라미터에 지정된 시간 단위로 GET 임시폴더 하위의 파일들을 지우는 작업을 수행하게 됩니다. 이때 GET 임시폴더의 파일 중 하위의 rstime 파라미터에 지정된 시간 보다 Last Access Time 이 경과된 파일에 대해서만 삭제 작업을 수행합니다. PUT 임시폴더의 경우는 EMC CENTERA나 Webdav으로 전송을 마친 뒤 바로 삭제되므로 해당 옵션과는 상관이 없습니다. 지정되지 않을 경우, 해당 작업이 이루어지지 않게 되므로 EMC CENTERA나 Webdav 등의 스토리지 장치를 사용할 경우에는 반드시 사용하셔야 합니다.

일반 파일시스템에는 해당되지 않습니다.

기본값: "60"초

scheduler.rstime

위의 iterationop 옵션이 활성화된 상태에서 유효하게 작동하며 임시 폴더(캐쉬)에서 유지되는 시간을 의미 합니다.

기본값: "60"초

scheduler.qtimeout

트랜잭션이 타임아웃 되기 전까지 큐에 남아있을 수 있는 시간을 초단위로 지정. 만일, 이 파라메타가 지정되어 0이 아닌 값을 갖게 될 경우, 큐에서 대기하고 있는 아카이브에 대한 모든 요청은 qtimeout 파라메타에 지정된 시간 동안만 큐에 남아 있을 수 있습니다. 시간이 만료되면, Xtorm은 에러를 발생시키고, 파일이 다른 아카이브에 존재할 경우 트랜잭션을 완수하기 위해 다른 아카이브로 대체하여 시도하게 됩니다. qtimeout 파라메타가 존재하지 않거나 0으로 설정되어 있을 경우는 타임아웃이 없습니다. 그리고, 요청된 작업은 큐에 무한정 남아있게 됩니다. 대기큐에 쌓여 있으면서 아카이브 쓰레드에 의해 핸들링되고 있지 않은 트랜잭션들은 모두 자동 취소됩니다. 아카이브 쓰레드에 의해 핸들링되는 트랜잭션은 장치 드라이버에 의해 이미 서비스 되고 있는 것을 의미하는데, 자바의 경우 쓰레드가 강제적으로 종료될 경우, 예기치 않은 불안정한 상태를 초래할 수 있기 때문에 아카이브 쓰레드에 의해 이미 핸들링되고 있는 트랜잭션은 취소하지 않습니다.

deviceIF

스토리지 매체와 통신할 때 사용할 매체 정보를 정의

deviceIF.class

파일 시스템에 파일을 쓰거나 읽을 경우에 사용할 자바 클래스 이름. 매체에 따라 각각의 클래스가 제공될 것입니다.

파일 시스템: "[com.windfire.agents.archive.asysAgentArMediaFilesys](#)"

(Containerizing, Streaming, Direct file transmission for bundle)

EMC Batch Cache Direct:

["com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCacheDirect"](#)

(Containerizing, Streaming, Cache, Direct file transmission for bundle)

EMC Batch Cache Shred:

["com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCacheShred"](#)

(Streaming, Cache)

WebDav: "[com.windfire.agents.archive.asysAgentArMediaDav](#)"

(Streaming)

WebDav Batch Cache:

["com.windfire.agents.archive.asysAgentArMediaWebDavBatchCache"](#)

(Containerizing, Cache)

WebDav Batch Cache Direct:

["com.windfire.agents.archive.asysAgentArMediaWebDavBatchCacheDirect"](#)

(Containerizing, Cache, Direct file transmission for bundle)

HTTP: "[com.windfire.agents.archive.asysAgentArMediaHTTP](#)"

(Containerizing, Streaming, Direct file transmission for bundle)

EMC Avamar: "[com.windfire.agents.archive.asysAgentArMediaAvamar](#)"

(Containerizing, Cache)

SnapLock: "[com.windfire.agents.archive.asysAgentArMediaSnapLock](#)"

(Containerizing, Cache)

threadpool

파일을 읽고, 쓰고, 삭제하는 일에 사용될 쓰레드 정보를 정의. 어떤 운영에 대하여 얼마나 많은 쓰레드를 제어할 지 여부에 따라 여러 쓰레드 풀을 사용할 수 있습니다.

threadpool.ops

count 속성에 있는 쓰레드에 의해 지원될 운영 내용. 특정 쓰레드 목적에 맞게 복수의 ops 및 Count 속성을 지정할 수 있습니다.

기본 값: “read,write,delete”

threadpool.ops

스토리지 장치와 인터페이스에 사용될 작업 쓰레드 수 정의. 이는 동시에 수행될 수 있는 요청 수를 결정합니다.

기본 값: “2”

asysAgentArMediaFilesys

asysAgentArMediaFilesys 클래스 유효 속성

묶음 처리, 스트리밍 서비스, 향상된 묶음 파일 처리 등의 기능을 지원합니다.

deviceIF.basedir

오브젝트를 읽거나 기록할 대상 디렉토리 경로를 지정.

deviceIF.retention

해당 저장소로 저장되는 문서의 보존주기를 의미함. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 ‘-1’로 지정할 수 있음. 지정하지 않으면 기본 값으로 설정됨.

기본 값: “-2” (보존주기 미지정)

deviceIF.dateformat

중분 백업이 가능하도록 하기 위해 사용되는 파라미터로 지정하게 되면 해쉬 디렉토리 상위에 지정된 Date 포맷의 폴더를 생성함. 지정하지 않으면 해당 디렉토리를 생성하지 않음.

예: “yyyyMMdd”, “yyyy-MM-dd” (대소문자 구분함)

deviceIF.batch

파일 시스템에서도 묶음 처리를 할 수 있도록 지원. 해당 기능이 활성화된 상태에서 배치API를 통해 입력하게 되면 대상 파일들이 묶음으로 저장됨. 묶음으로 입력된 파일에 대하여 조회 요청을 받게 되면 해당 묶음 파일에 대한 오프셋(Offset) 정보를 이용해 단위 파일에 대한 부분만 따로 분리하여 전송함.

기본값: “false”

deviceIF.filesystemclass

물리 볼륨 관리를 위해 볼륨 정보를 취득하기 위한 클래스의 종류를 지정. 볼륨 정보 취득을 위해 다음과 같이 두가지의 클래스를 지원, OS 셸 명령 사용 방식(asysFileSystemShell)의 경우 별도 설정이 필요 없지만, OS API를 호출하는 JNI 방식(asysFileSystem)의 경우 엔진 구동시 해당 라이브러리를 JVM 실행 옵션을 이용하여 다음과 같이 지정해야 함.

해당 파라미터를 지정할 경우 반드시 DATA 에이전트에 ‘physicalvolumemanager’를 활성화 시켜야 정상적으로 동작 합니다.

```
-Djava.library.path=[library path]
(solaris : -Djava.library.path=/home/Xtorm/lib/libFileSystem.so)
OS 쉘 사용 : "com.windfire.agents.archive.asysFileSystemShell"
OS API 사용 (JNI 방식 호출) : "com.windfire.agents.archive.asysFileSystem"
```

deviceIF.hashdirs

새로운 파일을 작성할 경우 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정하십시오. 0으로 입력하면 Hash Directory를 사용하지 않습니다.

기본 값: "10"

deviceIF.bgthreadpool

백그라운드 필터를 사용할 경우, 백그라운드로 운용될 쓰레드 풀의 개수를 지정합니다.

기본 값: "0"

threadpool.ops

count 속성에 있는 쓰레드에 의해 지원될 운영 내용. 특정 쓰레드 목적에 맞게 복수의 ops 및 count 속성을 지정할 수 있음.

기본 값: "read,write,delete"

threadpool.count

스토리지 장치와 인터페이스에 사용될 작업 쓰레드 수 정의. 이는 동시에 수행될 수 있는 요청 수를 결정합니다

기본 값: "2"

scheduler.qtimeout

트랜잭션이 타임아웃 되기 전까지 큐에 남아있을 수 있는 시간을 초단위로 지정.

만일, 이 파라미터가 지정되어 0이 아닌 값을 갖게 될 경우, 큐에서 대기하고 있는 아카이브에 대한 모든 요청은 qtimeout 파라미터에 지정된 시간 동안만 큐에 남아 있을 수 있음.

시간이 만료되면, Xtorm은 에러를 발생시키고, 파일이 다른 아카이브에 존재할 경우 트랜잭션을 완수하기 위해 다른 아카이브로 대체하여 시도하게 됩니다. qtimeout 파라미터가 존재하지 않거나 0으로 설정되어 있을 경우는 타임아웃이 없음. 그리고, 요청된 작업은 큐에 무한정 남아있게 됨. 대기 큐에 쌓여 있으면서 아카이브 쓰레드에 의해 핸들링되고 있지 않은 트랜잭션들은 모두 자동 취소 됨. 아카이브 쓰레드에 의해 핸들링되는 트랜잭션은 장치 드라이버에 의해 이미 서비스 되고 있는 것을 의미하는데, 자바의 경우 쓰레드가 강제적으로 종료될 경우, 여기치 않은 불안정한 상태를 초래할 수 있기 때문에 아카이브 쓰레드에 의해 이미 핸들링되고 있는 트랜잭션은 취소하지 않음.

구성 예)

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile=".../
../msg/data.mc" minhealth="1">
  <dbpools>
    <pool name="master" driver="oracle.jdbc.driver.OracleDriver"
      connect="jdbc:oracle:thin:@hdual:1521:orcl" user="Xtorm" pswd="Xtorm" count="20"
```

```

preconnect="FALSE" unicode="FALSE" dbtype="2" />
</dbpools>
<startup mainpool="master" checksum="NO" localtime="GMT+09:00"/>
<listspooler listdir="spool" timeout="6" />
<physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>
<defaultaudit agentserver="" agentname="" />
</agent>

<agent name="MAIN" type="ARCHIVE" desc="Local Disk"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asysservlets.asysOperArchive"
msgfile="msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaFilesys"
filesystemclass="com.windfire.agents.archive.asysFileSystemShell" basedir="c:/Xtorm/storage"
hashdirs="10" retention="240" dateformat="yyyyMMdd" batch="true"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>

```

SAN 구성 예)

```

<agent name="SAN" type="ARCHIVE" desc="Local Disk"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asysservlets.asysOperArchive"
msgfile="msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo" qtimeout="60">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaFilesys"
filesystemclass="com.windfire.agents.archive.asysFileSystemShell" basedir="c:/Xtorm/storage"
hashdirs="10" etention="240" dateformat="yyyyMMdd" batch="true"/>
    <threadpool ops="read,write,delete" count="50"/>
  </scheduler>
</agent>

```

asyAgentArMediaEMCCentera

asyAgentArMediaEMCCentera 클래스 유효 속성

스트리밍 서비스 기능을 지원합니다.

deviceIF.basedir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성합니다.

deviceIF.deletedsize

'GET' 폴더 하위에 저장되는 파일들의 총 용량을 의미하며 지정된 값이 넘으면 임시 파일들을 삭제함. 값을 지정하지 않을 경우 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야함.

deviceIF.deletedage

'GET' 폴더 하위에 저장되는 파일들이 삭제되지 않고 남아있는 시간을 의미하며 지정된 값을 넘는 파일들을 삭제함. 값을 지정하지 않을 경우 임시 파일들을 삭제하지 않아 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야함.
(단위: 초)

참고: deletedsize 옵션과 deletedage 옵션은 반드시 둘 다 지정 해야함.

deviceIF.retention

해당 아카이브로 저장되는 문서의 보존주기를 의미합니다. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정됩니다. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1'로 지정할 수 있습니다. 지정하지 않으면 기본 값으로 설정됩니다.

기본 값: "-2" (보존주기 미지정)

deviceIF.pooladdrlist

EMC Centera 장비의 연결 리스트 정보를 지정합니다. ', '로 구분하여 복수개의 서버를 지정할 수 있습니다. 접근 권한을 제어하기 위한 Centera pea 파일은 지정할 수 없습니다.

deviceIF.poolbuffersize

EMC Centera 관련 처리를 위해 할당되는 내부 버퍼의 크기를 바이트(byte) 단위로 지정합니다. 지정하지 않을 경우 기본값으로 설정되며 최소 1024 bytes (1Kb) ~ 최대 10485760 bytes (10Mb) 까지 지정 가능 합니다.

기본 값: "16*1024" bytes (16Kb)

deviceIF.timeout

EMC Centera 연결을 위한 TCP/IP 타임아웃 시간을 millisecond 단위로 지정합니다. 지정하지 않을 경우 기본값으로 설정되며 최대 600000 ms (10 minutes) 까지 지정 가능 합니다

기본 값: "120000" ms (2 minutes)

구성 예)

```
<agent name="EMC" type="ARCHIVE" desc="EMC"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/servlet/asysservlets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo" qtimeout="10" >
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaEMCCentera"
basedir="storage/EMC" pooladdrlist="192.168.0.50,192.168.0.51,192.168.0.52"
deletedsize="104857600" deletedage="86400" poolbuffersize="16"
timeout="120" retention="300"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaEMCCenteraBatch**asysAgentArMediaEMCCenteraBatch 클래스 유효 속성**

묶음 처리, 스트리밍 서비스 등의 기능을 지원합니다.

deviceIF.tempdir

묶음 처리 시 임시 파일을 지정된 경로 하위에 저장 합니다.

deviceIF.bufferstream

묶음 처리 시 해당 옵션을 활성화 시킬 경우 tempdir 파라미터에 지정된 경로 하위로 임시 파일을 생성하지 않고 메모리를 사용하여 처리합니다.

기본 값: "false"

다음의 파라미터 구성은 [asysAgentArMediaEMCCentera](#) 기능과 동일합니다.

deviceIF.basedir

deviceIF.deletedsize

deviceIF.deletedage

deviceIF.retention

deviceIF.pooladdrlist

deviceIF.poolbuffersize

deviceIF.timeout

구성 예)

```
<agent name="EMCBATCH" type="ARCHIVE" desc="EMCBATCH"
class="com.alerotech.agents.archive.asysAgentArchive" opclass="/servlet/scservlets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatch"
basedir="storage/EMCBATCH" pooladdrlist="192.168.0.50,192.168.0.51,192.168.0.52"
deletedsize="104857600" deletedage="86400" tempdir="storage/EMCtemp"
bufferstream="false" poolbuffersize="16" timeout="120" retention="300"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaEMCCenteraBatchCache

asysAgentArMediaEMCCenteraBatchCache 클래스 유효 속성

묶음 처리, 스트리밍 서비스, 캐싱 등의 기능을 지원합니다.

scheduler.iterationop

EMC 또는 Webdav용 아카이브의 경우와 같이 임시 폴더(캐쉬)를 운용하는 경우, GET 임시 폴더 하위의 파일들을 삭제해 주어야 하는데, 이 파라미터에 지정된 시간 단위로 GET 임시폴더 하위의 파일들을 지우는 작업을 수행함. 이때 GET 임시폴더의 파일 중 하위의 rtime 파라미터에 지정된 시간 보다 Last Access Time 이 경과된 파일에 대해서만 삭제 작업을 수행. PUT 임시폴더의 경우는 EMC CENTERA나 Webdav으로 전송을 마친 뒤 바로 삭제되므로 해당 옵션과는 상관이 없음. 지정되지 않을

경우, 해당 작업이 이루어지지 않게 되므로 EMC CENTERA나 Webdav 등의 스토리지 장치를 사용할 경우에는 반드시 지정해야 함. 일반 파일시스템에는 해당되지 않음.

적용 클래스:

[asysAgentArMediaEMCCenteraBatchCache](#)
[asysAgentArMediaEMCCenteraBatchCacheDirect](#)
[asysAgentArMediaEMCCenteraBatchCacheShred](#)
[asysAgentArMediaWebDavBatchCache](#)
[asysAgentArMediaWebDavBatchCacheDirect](#)
[asysAgentArMediaAvamar](#)

기본값: "60"초

scheduler.rstime

위의 iterationop 옵션이 활성화된 상태에서 유효하게 작동하며 임시 폴더(캐쉬)에서 유지되는 시간을 의미.

기본값: "60"초

deviceIF.basedir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성.

deviceIF.retention

해당 저장소로 저장되는 문서의 보존주기를 의미함. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1'로 지정할 수 있음. 지정하지 않으면 기본 값으로 설정됨.

기본 값: "-2" (보존주기 미지정)

deviceIF.tempdel

'PUT', 'GET' 폴더 하위에 저장된 파일의 삭제 여부를 지정합니다. 지정하지 않을 경우 기본값으로 설정됩니다.

기본 값: "true"

deviceIF.pooladdrlist

EMC Centera 장비의 연결 리스트 정보를 지정합니다. ', '로 구분하여 복수개의 서버를 지정할 수 있습니다. 접근 권한을 제어하기 위한 Centera pea 파일을 지정할 수 있습니다.

예: "10.1.1.1,10.1.1.2?D:/us2_profile1_rqecw.pea"

deviceIF.poolbuffersize

EMC Centera 관련 처리를 위해 할당되는 내부 버퍼의 크기를 바이트(byte) 단위로 지정합니다. 지정하지 않을 경우 기본값으로 설정되며 최소 1024 bytes (1Kb) ~ 최대 10485760 bytes (10Mb) 까지 지정 가능합니다.

기본 값: 16*1024 bytes (16Kb)

deviceIF.deletedsize

'GET' 폴더 하위에 저장되는 파일들의 총 용량을 의미하며 지정된 값이 넘으면 임시 파일들을 삭제함. 값을 지정하지 않을 경우 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야 함.

적용 클래스:

[asysAgentArMediaEMCCentera](#)
[asysAgentArMediaEMCCenteraBatch](#)

deviceIF.deletedage

‘GET 폴더 하우에 저장되는 파일들이 삭제되지 않고 남아있는 시간을 의미하며 지정된 값을 넘는 파일들을 삭제함. 값을 지정하지 않을 경우 임시 파일들을 삭제하지 않아 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야 함.
 (단위: 초)

주의) deletedsize 옵션과 deletedage 옵션은 반드시 둘 다 지정 해야 함.

deviceIF.pooladdrlist

EMC Centera 장비의 연결 리스트 정보를 저장함. ‘, 로 구분하여 복수개의 서버를 지정할 수 있고 접근 권한을 제어하기 위한 Centera pea 파일을 지정할 수 있음.

pea파일 저장가능 클래스:

[asysAgentArMediaEMCCenteraBatchCache](#)
[asysAgentArMediaEMCCenteraBatchCacheDirect](#)
[asysAgentArMediaEMCCenteraBatchCacheDirectShred](#)

예: “10.1.1.1,10.1.1.2?D:/us2_profile1_rqecw.pea”

deviceIF.timeout

EMC Centera 연결을 위한 TCP/IP 타임아웃 시간을 초단위로 지정.
 지정하지 않을 경우 기본값으로 설정되며 최대 600초 (10 분) 까지 지정 가능.

기본 값: “120”초. (2분)

deviceIF.c_avoidance

EMC Centera 저장 시 생성되는 Clip-ID의 충돌을 방지하기 위해 Centera 자체 확장 기능을 사용할지의 여부를 지정.

적용 클래스:

[asysAgentArMediaEMCCenteraBatchCache](#)
[asysAgentArMediaEMCCenteraBatchCacheDirect](#)
[asysAgentArMediaEMCCenteraBatchCacheDirectShred](#)

기본 값: “true”

deviceIF.blobpartialread

묶음처리(Containerizing)로 저장된 Blob에 대하여 조회 요청 시 묶음 전체를 캐쉬영역에 반환할지 또는 해당 파일 부분만 캐쉬영역에 반환 할지의 여부를 지정.

적용 클래스:

[asysAgentArMediaEMCCenteraBatchCache](#)
[asysAgentArMediaEMCCenteraBatchCacheDirect](#)
[asysAgentArMediaEMCCenteraBatchCacheDirectShred](#)

기본 값: “false”

구성 예)

```
<agent name="EMC_BATCH_CACHE" type="ARCHIVE" desc="EMC BATCH CACHE"
class="com.alerotech.agents.archive.asysAgentArchive" opclass="/servlet/scservlets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCache"
basedir="storage/EMCBATCH" tempdel="true" pooladdrlist="10.1.1.1,10.1.1.2?D:/
us2_profile1_rqecw.pea " poolbuffersize="16" timeout="120" retention="300"
c_avoidance="true" blobpartialread="true"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaEMCCenteraBatchCacheShred

asysAgentArMediaEMCCenteraBatchCacheShred 클래스 유효 속성

스트리밍 서비스, 캐싱 기능을 지원합니다.

파라미터 구성은 [asysAgentArMediaEMCCenteraBatchCache](#) 기능과 동일합니다.

deviceIF.deletedsize 기능 제거됨.

deviceIF.deletedage 기능 제거됨.

삭제 주기 **scheduler**에서 대체 함.

scheduler qtimeout="60" iterationop="600" rstime="600" 단위(초)
rstime 해당 시간만큼 지난 파일 삭제 대상, **iterationop** 삭제 스케줄 작동 주기

deviceIF.EMBEDDBLOB

센터라에 파일저장시 실제파일과 meta정보를 1개에 blob에 저장하는 기능임.
 센터라 object 수를 줄일수 있는 효과가 있음.
 값:"true"

deviceIF.IgnoreClipName

센터라 클립이름이 가 다른경우에도 조회 및 삭제가 가능한 옵션
 기본값:"true"

deviceIF.EMBEDDEDBLOBSIZE

blob 함께 저장할 파일 사이즈
 100k 기본 값이며, 1MB 까지 가능하나 100kb 이상일 경우 성능 문제 발생한다고함

기본값: "100" (단위 KB)

deviceIF.RETENTIONDELETE

센터라에 RETENTION 걸려 있는 파일 센터라에서 RETENTION 체크하지 않고 삭제 가능
 RETENTIONDELETE="YES" 일 경우 RETENTION 값 무시 하고 삭제 가능

주의 사항: 위 옵션 적용해도 DB(ASYSCONTENTFELEMENT)에서 RETENTION 값은 0으로 업데이트 해야
 삭제됨.

기본값: "no"

deviceIF.filesystemkb

물리블록 사용시kb로 표기

값: ture

구성 예)

```
<agent name="BCH_CACHE_SR" type="ARCHIVE" desc="EMC BATCH CACHE SHRED"
class="com.alerotech.agents.archive.asysAgentArchive" opclass="/servlet/scservets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF
      class="com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCacheShred"
      basedir="storage/EMCBATCH" tempdel="true" pooladdrlist="10.1.1.1,10.1.1.2?D:/
        us2_profile1_rqecw.pea" poolbuffersize="16" timeout="120" retention="300"
      c_avoidance="true" blobpartialread="true" EMBEDDEDLOB="true" EMBEDDEDBLOBSIZE="100"
      RETENTIONDELETE="no" FILESYSTEMKB="true"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaEMCCenteraBatchCacheDirect**asysAgentArMediaEMCCenteraBatchCacheDirect 클래스 유효 속성**

묶음 처리, 스트리밍 서비스, 캐싱, 향상된 묶음 처리 등의 기능을 지원합니다.

파라미터 구성은 [asysAgentArMediaEMCCenteraBatchCache](#) 기능과 동일합니다.

deviceIF.deletedsize 기능 제거됨.

deviceIF.deletedage 기능 제거됨. **scheduler**

삭제 주기|**scheduler**에서 대체 함.

scheduler qtimeout="60" iterationop="600" rstime="600" 단위(초)
rstime 해당 시간만큼 지난 파일 삭제 대상, **iterationop** 삭제 스케줄 작동 주기

deviceIF.IgnoreClipName

센터라 클립이름이 가 다른경우에도 조회 및 삭제가 가능한 옵션
 기본값:"true"

deviceIF.RETENTIONDELETE

센터라에 RETENTION 걸려 있는 파일 센터라에서 RETENTION 체크하지 않고 삭제 가능
 RETENTIONDELETE="YES" 일 경우 RETENTION 값 무시 하고 삭제 가능

주의 사항: 위 옵션 적용해도 DB(ASYSCONTENTFELEMENT)에서 RETENTION 값은 0으로 업데이트 해야
 삭제됨.

기본값: "no"

deviceIF.filesystemkb

물리블록 사용시kb로 표기

값: ture

구성 예)

```
<agent name="BCH_CACHE_DT" type="ARCHIVE" desc="EMC BATCH CACHE DIRECT"
class="com.alerotech.agents.archive.asysAgentArchive" opclass="/servlet/scservlets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF
class="com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCacheDirect"
basedir="storage/EMCBATCH" tempdel="true" pooladdrlist="10.1.1.1,10.1.1.2?D:/
us2_profile1_rqecw.pea" poolbuffersize="16" timeout="120" retention="300" c_avoidance="true"
blobpartialread="true" RETENTIONDELETE="no" FILESYSTEMKMB="true"/>
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaDav

asysAgentArMediaDav 클래스 유효 속성

스트리밍 서비스 기능을 지원합니다.

deviceIF.baseurl

웹 서버의 연결 리스트 정보를 지정합니다. ; 로 구분하여 복수개의 URL을 지정할 수 있습니다.

예: "http://10.13.10.60;http://10.13.10.61;http://10.13.10.62"

deviceIF.basedir

웹 프로토콜로 접속된 저장소 상의 파일 저장 경로를 지정합니다.

예: "fcfs_data/Xtorm/"

deviceIF.tempdir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성합니다.

deviceIF.hashdirs

웹 서버로 연결된 저장소 상의 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정하십시오. 0으로 입력하면 Hash Directory를 사용하지 않습니다.

기본 값: "10"

deviceIF.securetype

웹 서버에 접속하기 위한 사용자 인증 방식을 지정합니다.

- 1: NTLM 방식의 보안 인증 방식 (윈도우 도메인 인증 방식)
- 2: WebDav 서버 자체적으로 제공하는 사용자/패스워드 인증 방식

deviceIF.secureserver

사용자 인증 서비스를 수행하는 인증 서버의 주소를 지정합니다. 지정하지 않을 경우 사용자 인증을 수행하지 않습니다.

(NTLM 방식에만 유효함)

deviceIF.securedomain

사용자 인증을 위한 도메인명을 지정합니다. 지정하지 않을 경우 사용자 인증을 수행하지 않습니다.

(NTLM 방식에만 유효함)

deviceIF.secureuser

웹 서버에 접속하기 위한 사용자 아이디를 지정합니다. 지정하지 않을 경우 사용자 인증을 수행하지 않습니다.

deviceIF.securepswd

웹 서버에 접속하기 위한 사용자 패스워드를 지정합니다. 지정하지 않을 경우 사용자 인증을 수행하지 않습니다.

deviceIF.retries

웹 서버 연결에 실패한 경우 재시도 회수를 지정합니다.

기본 값: "100"

deviceIF.retrywait

웹 서버 연결 재시도를 위한 대기 시간을 지정합니다.

(단위: milliseconds)

기본 값: "5000" (5초)

deviceIF.failnodeafter

각 노드에 대하여 Fail-Over를 위한 옵션을 제공하기 위한 파라미터로서 각각의 노드에 대한 연결 시도 회수를 지정합니다. 지정된 회수를 초과하면 연결 불가 노드로 판단하여 다음 노드로 연결을 시도하게 되며 아래의failnodewait 파라미터에 지정된 시간 동안 접속을 재시도하지 않습니다. 만약, 이 값이0으로 설정되면 연결 시도가 실패하더라도 해당 노드는 연결 불가 노드로 지정되지 않게 되며 이후 정상적으로 재접속 시도를 수행합니다. 이 파라미터는 다수의 연결 노드 구성 시에만 사용되어야 합니다.

기본 값: "0"

deviceIF.failnodewait

각 노드에 대하여 Fail-Over를 위한 옵션을 제공하기 위한 파라미터로서 연결 불가 노드로 지정된 노드를 다시 재접속 하도록 하기 위해 재접속 대기 시간을 지정합니다. 만약, 이 값이0으로 설정되면 연결 불가 노드로 지정된 노드에 대해서 이후로 재접속 시도를 하지 않습니다.(권장되지 않음) 이 파라미터는 다수의 연결 노드 구성 시에만 사용되어야 합니다.

(단위: milliseconds)

기본 값: "0"

deviceIF.retention

해당 아카이브로 저장되는 문서의 보존주기를 의미합니다. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정됩니다. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1'로 지정할 수 있습니다. 지정하지 않으면 기본 값으로 설정됩니다.

기본 값: "-2" (보존주기 미지정)

구성 예)

```
<agent name="HCAP_ARC" type="ARCHIVE" desc="HCAP Storage"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asysserMets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaDav"
```

```
baseurl="http://10.13.10.60;http://10.13.10.61;http://10.13.10.62" basedir="fcfs_data/Xtorm/" hashdirs="500"
securetype="1"          secureuser="" securepswd="" securedomain="" secureserver=""
tempdir="/sw/xtorm/temp" retries="3" retrywait="5000" failnodeafter="3" failnodewait="5000"
retention="300"/>
    <threadpool ops="read,write,delete" count="50"/>
</scheduler>
</agent>
```

asysAgentArMediaWebDavBatchCacheDirect

asysAgentArMediaWebDavBatchCacheDirect 클래스 유효 속성

목록 처리, 캐싱, 향상된 목록 처리 등의 기능을 지원합니다.

파라미터 구성은 [asysAgentArMediaWebDavBatchCache](#) 기능과 동일합니다.

구성 예)

```
<agent name="WEBDAV_ARC" type="ARCHIVE" desc="WEBDAV archive for HCAP"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asysservlets.asysOperArchive"
msgfile=".../msg/archive.mc">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo" perfile="NO" timeout="60"
nodecheck="60">
        <deviceIF
class="com.windfire.agents.archive.asysAgentArMediaWebDavBatchCacheDirect" basedir="storage"
hashdirs="10" tempdel="true" retention="240" timeout="120">
            <dpool name="node1" addr="http://localhost:8080/webdav/testFolder"
user="admin" pswd="admin" count="10" preconnect="true" />
            <dpool name="node2" addr="http://HDUAL:8080/webdav/testFolder" user="admin"
pswd="admin" count="10" preconnect="true" />
        </deviceIF>
        <threadpool ops="read,write,delete" count="50" />
    </scheduler>
</agent>
```

asysAgentArMediaWebDavBatchCache

asysAgentArMediaWebDavBatchCache 클래스 유효 속성

목록 처리, 캐싱 등의 기능을 지원합니다.

deviceIF.hashdirs

웹 서버로 연결된 저장소 상의 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정하십시오. 0으로 입력하면 Hash Directory를 사용하지 않습니다.

기본 값: "10"

deviceIF.timeout

웹 서버 연결 시도를 위한 대기 시간을 지정합니다.
(단위 : seconds)

기본 값: "60" (1분)

deviceIF.perfile

폴더에만 보존주기를 지정할지, 하위의 파일 단위 까지 지정할지의 여부를 설정합니다.
("false"일 경우, 폴더레벨에만 지정)

기본 값: "false"

deviceIF.nodecheck

각 노드들의 연결 상태를 점검하는 동작 주기를 지정합니다.
(단위: 초)

기본 값: "60" 초

dpool

해당 노드의 캐백션 풀을 정의합니다.

dpool.name

해당 노드의 이름을 지정합니다.

dpool.addr

해당 노드로 접근하기 위한 접속 URL을 지정합니다.

dpool.user

해당 노드의 웹 서버로 접속하기 위한 사용자 아이디를 지정합니다.

dpool.pswd

해당 노드의 웹 서버로 접속하기 위한 사용자 비밀번호를 지정합니다.

dpool.count

해당 노드의 연결 pool 개수를 지정합니다.

dpool.preconnect

해당 노드의 연결을 엔진 구동 시점에 미리 수행할 지의 여부를 지정합니다.
("true"일 경우 해당 노드에 미리 연결을 수행함)

기본값: "false"

다음의 파라미터 구성은 [asysAgentArMediaEMCCenteraBatchCache](#) 기능과 동일합니다.

deviceIF.basedir

deviceIF.tempdel

deviceIF.retention

구성 예)

```
<agent name="WEBDAV_ARC" type="ARCHIVE" desc="WEBDAV archive for HCAP"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/servlet/asysservlets.asysOperArchive"
msgfile="../../msg/archive.mc">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo" perfile="NO" timeout="60"
nodecheck="60">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaWebDavBatchCache"
basedir="storage" hashdirs="10" tempdel="true" retention="240" timeout="120">
```

```

        <dpool name="node1" addr=http://localhost:8080/webdav/testFolder user="admin"
pswd="admin" count="10" preconnect="TRUE" />
        <dpool name="node2" addr="http://HDUAL:8080/webdav/testFolder" user="admin"
pswd="admin" count="10" preconnect="TRUE" />
    </deviceIF>
    <threadpool ops="read,write,delete" count="50" />
</scheduler>
</agent>

```

asysAgentArMediaHTTP

asysAgentArMediaHTTP 클래스 유효 속성

묶음 처리, 스트리밍 서비스, 향상된 묶음 처리 등의 기능을 지원합니다.

deviceIF.dateformat

중분 백업이 가능하도록 하기 위해 사용되는 파라미터로 지정하게 되면 해시 디렉토리 상위에 지정된 Date 포맷의 폴더를 생성. 지정하지 않으면 해당 디렉토리를 생성하지 않음.

예: "yyyyMMdd", "yyyy-MM-dd" (대소문자 구분함 scheduler.rstime)

deviceIF.batch

파일 시스템에서도 묶음 처리를 할 수 있도록 지원. 해당 기능이 활성화된 상태에서 배치 API를 통해 입력하게 되면 대상 파일들이 묶음으로 저장 됨. 묶음으로 입력된 파일에 대하여 조회 요청을 받게 되면 해당 묶음 파일에 대한 오프셋(Offset) 정보를 이용해 단위 파일에 대한 부분만 따로 분리하여 전송함.

기본값: "false"

deviceIF.deleteage

'GET' 폴더 하위에 저장되는 파일들이 삭제되지 않고 남아있는 시간을 의미하며 지정된 값을 넘는 파일들을 삭제함. 값을 지정하지 않을 경우 임시 파일들을 삭제하지 않아 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야 함.
(단위: 초)

deviceIF.retention

해당 저장소로 저장되는 문서의 보존주기를 의미함. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1'로 지정할 수 있음. 지정하지 않으면 기본 값으로 설정됨.

기본 값: "-2" (보존주기 미지정)

deviceIF.baseurl

웹 서버의 연결 리스트 정보를 지정. ';'로 구분하여 복수개의 URL을 지정 가능 함.

예: "http://10.13.10.60;http://10.13.10.61;http://10.13.10.62"

deviceIF.basedir

웹 프로토콜로 접속된 저장소 상의 파일 저장 경로를 지정.

예: "fcfs_data/Xtorm/", "rest/"

deviceIF.tempdir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성

deviceIF.hashdirs

웹 서버로 연결된 저장소 상의 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정 권장. 0으로 입력하면 Hash Directory를 사용하지 않음.

기본 값: "10"

deviceIF.securetype

웹 서버에 접속하기 위한 사용자 인증 방식을 지정.

- 1: NTLM 방식의 보안 인증 방식 (윈도우 도메인 인증 방식)
- 2: WebDav 서버 자체적으로 제공하는 사용자/패스워드 인증 방식

deviceIF.secureserver

사용자 인증 서비스를 수행하는 인증 서버의 주소를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

(NTLM 방식에만 유효함)

deviceIF.securedomain

사용자 인증을 위한 도메인 명을 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

(NTLM 방식에만 유효함)

deviceIF.secureuser

웹 서버에 접속하기 위한 사용자 아이디를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

deviceIF.securepswd

웹 서버에 접속하기 위한 사용자 패스워드를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

deviceIF.retrywait

단위 트랜잭션 실패(다른 노드가 이미 사용중인 경우) 시, 재시도 wait 시간 (단위:milli-seconds).

기본 값: "500" milli-seconds

deviceIF.timeout

단위 노드 연결을 위한 타임아웃 최대값 (단위:milli-seconds)

기본 값: "10000" milli-seconds

deviceIF.MAXTOTALCONNECTIONS

Connection Pooling 사용으로 인한 최대 접속 수(기본값: 30)

deviceIF.MAXHOSTCONNECTIONS

한 host에 접속할 수 있는 접속 수(기본값: 30)

deviceIF.IDLETIMEOUT

idle 인 접속 끊는 시간(초) (기본값: 120)

deviceIF.filesystemkb

물리볼륨 사용시 kb로 표기

값: true

파라미터 구성은 [asysAgentArMediaDav](#) 기능과 동일합니다.**구성 예)**

```

<agent name="1HS_ARC" type="ARCHIVE" desc="HCAP Storage"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asyserMets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaHTTP" basedir="temp"
hashdirs="10" tempdel="YES" retention="3600" timeout="120">
      <deviceIF class="com.windfire.agents.archive.asysAgentArMediaHTTP"
baseurl="http://10.1.21.77:8080"
        basedir="fcfs_data" hashdirs="1" securetype="1" secureuser="admin"
securepswd="admin" securedomain="shhan-PC" secureserver="" tempdir="c:/temp" TIMEOUT="1000"
retrywait="3000" failnodeafter="3" failnodewait="3000" retention=""
        MAXTOTALCONNECTIONS="100" MAXHOSTCONNECTIONS="100"
FILESYSTEMMKB = "true" />
      </deviceIF>
      <threadpool ops="read,write,delete" count="10"/>
    </scheduler>
  </agent>

```

asysAgentArMediaHCP**asysAgentArMediaHCP 클래스 유효 속성**

HCP 장비 7.2.x 이상 버전 사용

httpclient 4.x 버전 사용하는 신규 HCP 장비 아답터

목록 처리, 스트리밍 서비스, 향상된 목록 처리 등의 기능을 지원합니다.

deviceIF.dateformat

중분 백업이 가능하도록 하기 위해 사용되는 파라미터로 지정하게 되면 해쉬 디렉토리 상위에 지정된 Date 포맷의 폴더를 생성. 지정하지 않으면 해당 디렉토리를 생성하지 않음.

예: "yyyyMMdd", "yyyy-MM-dd" (대소문자 구분함 scheduler.rstime)

deviceIF.batch

파일 시스템에서도 목록 처리를 할 수 있도록 지원. 해당 기능이 활성화된 상태에서 배치API를 통해 입력하게 되면 대상 파일들이 목록으로 저장 됨. 목록으로 입력된 파일에 대하여 조회 요청을 받게 되면 해

당 묶음 파일에 대한 오프셋(Offset) 정보를 이용해 단위 파일에 대한 부분만 따로 분리하여 전송함.

기본값: "false"

deviceIF.deleteage

'GET' 폴더 하위에 저장되는 파일들이 삭제되지 않고 남아있는 시간을 의미하며 지정된 값을 넘는 파일들을 삭제함. 값을 지정하지 않을 경우 임시 파일들을 삭제하지 않아 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야 함.

(단위: 초)

deviceIF.retention

해당 저장소로 저장되는 문서의 보존주기를 의미함. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1' 로 지정할 수 있음. 지정하지 않으면 기본 값으로 설정됨.

기본 값: "-2" (보존주기 미지정)

deviceIF.baseurl

멀티텐던시 사용시 접속 주소 지정 (네임스페이스: nds1 테넌트: eXtorm-arc HCP도메인명: hcp.shbank.com)

Default 사용시 기존 http 주소 사용 ; 구분으로 주소 입력

예: 멀티텐던시: " http://eXtormtest.XtormTest.hisbmt.hcp.com"

Default : "http://10.13.10.60;http://10.13.10.61;http://10.13.10.62"

deviceIF.basedir

웹 프로토콜로 접속된 저장소 상의 파일 저장 경로를 지정.

멀티텐던시: "rest"

Default : "fcfs_data"

예: "fcfs_data", "rest"

deviceIF.tempdir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성

deviceIF.hashdirs

웹 서버로 연결된 저장소 상의 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정 권장. 0으로 입력하면 Hash Directory를 사용하지 않음.

기본 값: "10"

deviceIF.securetype

웹 서버에 접속하기 위한 사용자 인증 방식을 지정.

1: NTLM 방식의 보안 인증 방식(윈도우 도메인 인증 방식)

2: WebDav 서버 자체적으로 제공하는 사용자/패스워드 인증 방식

deviceIF.secureserver

사용자 인증 서비스를 수행하는 인증 서버의 주소를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

(NTLM 방식에만 유효함)

deviceIF.securedomain

사용자 인증을 위한 도메인 명을 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.
(NTLM 방식에만 유효함)
baseurl 주소에서 http 부분 제거

deviceIF.secureuser

웹 서버에 접속하기 위한 사용자 아이디를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

deviceIF.securepswd

웹 서버에 접속하기 위한 사용자 패스워드를 지정. 지정하지 않을 경우 사용자 인증을 수행하지 않음.

deviceIF.retrywait

단위 트랜잭션 실패(다른 노드가 이미 사용중인 경우) 시, 재시도 wait 시간 (단위:milli-seconds).

기본 값: "500" milli-seconds

deviceIF.timeout

단위 노드 연결을 위한 타임아웃 최대값 (단위:milli-seconds)

기본 값: "10000" milli-seconds

deviceIF.MAXTOTALCONNECTIONS

Connection Pooling 사용으로 인한 최대 접속 수(기본값: 30)

deviceIF.MAXHOSTCONNECTIONS

한 host에 접속할 수 있는 접속 수(기본값: 30)

deviceIF.IDLETIMEOUT

idle 인 접속 끊는 시간(초) (기본값: 120)

deviceIF.filesystemkb

물리블록 사용시 kb로 표기

값: ture

파라미터 구성은 [asysAgentArMediaHTTP](#) 기능과 동일합니다.

구성 예)

```
<agent name="HCP_ARC" type="ARCHIVE" desc="HCPStorage"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asyserMets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaHCP"
baseurl="http://eXtormtest.XtormTest.hisbmt.hcp.com"
basedir="rest" hashdirs="1000" securetype="1" secureuser="service"
securepswd="passwd9" securedomain="eXtormtest.XtormTest.hisbmt.hcp.com" secureserver=""
```

```

tempdir="c:/temp" TIMEOUT="5000" retrywait="1000" failnodeafter="0" failnodewait="0" retention=""
MAXTOTALCONNECTIONS="100" MAXHOSTCONNECTIONS="100" FILESYSTEMKMB="true" />
    <threadpool ops="read,write,delete" count="20"/>
</scheduler>
</agent>

```

asysAgentArMediaDataChian

asysAgentArMediaDataChian 클래스 유효 속성

KT블록체인 연동을 위한 어댑터

CLASSPATH에 lib/hcp/*.jar 전부 추가해줘야함.
 baseurl = 데이터체인 접속 URL 주소 (http 제외)
 basedir = 데이터체인 파일저장 uri
 SECRETCODE = 암호화 사용시 암호화키값
 이외 옵션은 HTTP 옵션과 동일함.

deviceIF.deleteage

‘GET’ 폴더 하위에 저장되는 파일들이 삭제되지 않고 남아있는 시간을 의미하며 지정된 값을 넘는 파일들을 삭제함. 값을 지정하지 않을 경우 임시 파일들을 삭제하지 않아 디스크 용량이 부족해 질 수 있으므로 반드시 설정 해야 함.
 (단위: 초)

deviceIF.baseurl

baseurl = 데이터체인 접속 URL 주소 (http 제외)

baseurl : “14.63.172.220:18080/dcedm”

deviceIF.basedir

데이터체인 파일저장 uri

예: “/file”

deviceIF.tempdir

지정된 경로 하위에 ‘GET’, ‘PUT’ 등의 임시 폴더를 생성

deviceIF.secretcode

암호화 사용시 암호화키값

deviceIF.retrywait

단위 트랜잭션 실패(다른 노드가 이미 사용중인 경우) 시, 재시도 wait 시간 (단위:milli-seconds).

기본 값: “500” milli-seconds

deviceIF.timeout

단위 노드 연결을 위한 타임아웃 최대값 (단위:milli-seconds)

기본 값: “10000” milli-seconds

deviceIF.MAXTOTALCONNECTIONS

Connection Pooling 사용으로 인한 최대 접속 수(기본값: 30)

deviceIF.MAXHOSTCONNECTIONS

한 host에 접속할 수 있는 접속 수(기본값: 30)

deviceIF.IDLETIMEOUT

idle 인 접속 끊는 시간(초) (기본값: 120)

파라미터 구성은 [asysAgentArMediaHTTP](#) 기능과 동일합니다.

구성 예)

```
<agent name="KTDATA_ARC" type="ARCHIVE" desc="HCAP Storage"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asyserMets.asysOperArchive"
msgfile="../../msg/archive.mc" minhealth="1">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaDataChain"
baseurl="14.63.172.220:18080/dcedm"
      basedir="/file" SECRETCODE="" tempdir="c:/temp" TIMEOUT="0"
DELETEDAGE="300" retrywait="3000" failnodeafter="3" failnodewait="3000" retention=""
      MAXTOTALCONNECTIONS="100" MAXHOSTCONNECTIONS="20"/>
    <threadpool ops="read,write,delete" count="5"/>
  </scheduler>
</agent>
```

**asysAgentArMediaAvamar****asysAgentArMediaAvamar 클래스 유효 속성**

목록 처리, 캐싱 등의 기능을 지원합니다.

deviceIF.basedir

지정된 경로 하위에 'GET', 'PUT' 등의 임시 폴더를 생성 합니다. 목록 처리 시 임시 파일 또한 지정된 경로 하위에서 처리합니다.

deviceIF.tempdel

'PUT', 'GET' 폴더 하위에 저장된 파일의 삭제 여부를 지정합니다. 지정하지 않을 경우 기본값으로 설정됩니다.

기본 값: "true"

deviceIF.retention

EMC Avamar 장비는 효율적인 저장소 관리에 무게를 두고 있기 때문에 보존주기의 개념이 다른 WORM 장비와 전혀 다르다는 점을 주의해야 합니다.

EMC Avamar 장비는 내부 프로세스를 활성화 시켜 보존기간이 설정된 파일을 보존주기 경과 시 자동으로 삭제할 수 있는데, 영구보존 설정된 파일일 경우 영구적으로 자동 삭제하지 않지만 일반 WORM 장비와 같이 완전적으로 삭제를 금지 하지는 않습니다. 즉, 별도의 Avamar 삭제 커맨드를 이용하여 명시적으로 파일을 삭제하게 되면 영구보존으로 설정된 파일이라도 삭제되게 됩니다.

기본적으로 Avamar 보존주기는 날짜 단위로 지정하기 때문에 보존주기를 지정할 경우에는 저장 시점으로부터 보존일자를 초단위로 환산하여 지정하여야 합니다 (예: 7일 -> $7 * 24 * 60 * 60 = 604800$)

* 지정 가능한 값

“-1” à Xtorm : 영구보존, 삭제 불가

Avamar : 영구보존, 삭제 가능

“0” à Xtorm : 지정안함, 삭제 가능

Avamar : 영구보존, 삭제 가능

양의정수 (초단위)

à Xtorm : 보존기간 (초),

Avamar : 보존기간 (지정된 초를 자동으로 일자로 환산함)

기본 값: “0” (Avamar 영구 보존, 삭제 가능)

deviceIF.pooladdrlist

EMC Avamar 클라이언트 노드의 연결 리스트 정보를 지정합니다. ‘,’ 로 구분하여 복수개의 노드를 지정할 수 있습니다.

예: “sut1,sut2,sut3”

deviceIF.hostloclist

EMC Avamar 클라이언트의 실행 경로 리스트를 지정합니다. ‘,’ 로 구분하여 복수개의 노드를 지정할 수 있으며 pooladdrlist 파라미터에 지정된 클라이언트 노드에 반드시 매핑되도록 지정해야 합니다.

예: “/opt/AVMCInt1,/opt/AVMCInt2,/opt/AVMCInt3”

deviceIF.bgthreadpool

백그라운드 필터를 사용할 경우, 백그라운드로 운용될 쓰레드 풀의 개수를 지정합니다.

기본 값: “0”

deviceIF.cmdshell

EMC Avamar CLi(Command Line Interface) 명령파일의 위치를 지정합니다. 절대경로 또는 상대경로를 모두 지정할 수 있습니다.

deviceIF.errlog

EMC Avamar CLi(Command Line Interface) 명령 수행 시 발생한 에러로그 파일을 저장할 위치를 지정합니다.

deviceIF.ostype

EMC Avamar 아답타가 구동되고 있는 운영체제(OS)를 지정합니다.

예: “1” – UNIX 계열, “2” – Windows 계열

구성 예)

```
<agent name="AVMARC" type="ARCHIVE" desc="AVAMAR backup"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asyserMets.asysOperArchive"
msgfile="../msg/archive.mc">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo" ITERATIONOP="300"
RSTIME="60">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaAvamar"
basedir="/storage/AVAMAR" pooladdrlist="sut1,sut2,sut3"
hostloclist="/opt/AVMCInt1,/opt/AVMCInt2,/opt/AVMCInt3" tempdel="true" cmdshell="../bin/avamarctl"
errlog="log/AVAMAR" retention="604800" />
    <threadpool ops="read,write,delete" count="2"/>
  </scheduler>
</agent>
```

asysAgentArMediaSnapLock

asysAgentArMediaSnapLock 클래스 유효 속성

목록 처리, 스트리밍 서비스, 향상된 목록 파일 처리 등의 기능을 지원합니다.

deviceIF.basedir

파일을 읽거나 기록할 대상 디렉토리 경로를 지정합니다. 파일 시스템에 마운트되는 SnapLock 논리 볼륨(LV)들의 상위 마운트 포인트를 지정해야 합니다.

예: "/vol" (마운트 포인트가 "/vol/Stg1", "/vol/Stg2" 등 인 경우)

deviceIF.hashdirs

새로운 파일을 작성할 경우 사용할 Hash Directory 수. Hash Directory는 파일을 여러 디렉토리에 분산하여 작성하는 방법이며 가급적 한 디렉토리에 적은 양의 파일 수가 존재하도록 조정하십시오. 0으로 입력하면 Hash Directory를 사용하지 않습니다.

기본 값: "10"

deviceIF.dateformat

중분 백업이 가능하도록 하기 위해 사용되는 파라미터로 지정하게 되면 해시 디렉토리 상위에 지정된 Date 포맷의 폴더를 생성합니다. 지정하지 않으면 해당 디렉토리를 생성하지 않습니다.

예: "yyyyMMdd", "yyyy-MM-dd" (대소문자 구분함)

deviceIF.retention

해당 아카이브로 저장되는 문서의 보존주기를 의미합니다. 아카이브에 최초 등록된 시점으로부터 삭제하지 못하도록 유지하기 위하여 초단위로 지정됩니다. 최대 68년까지 초단위로 지정할 수 있으며, 영구보존의 경우 '-1'로 지정할 수 있습니다. 지정하지 않으면 기본 값으로 설정됩니다.

기본 값: "-2" (보존주기 미지정)

deviceIF.retentiontype

보존주기의 계산 방식을 지정합니다.

예: "0" - 현재 시간(파일 기록 시점) + retention 값,

“1” - 절대 시간(Timestamp),

deviceIF.volume

파일 시스템에 마운트된 SnapLock 의 논리 볼륨(LV)명을 지정합니다.

deviceIF.mountpt

volume 파라미터에 지정된 논리 볼륨의 마운트 포인트를 지정합니다.

deviceIF.filer

SnapLock에 저장된 파일의 보존주기를 지정하기 위해서 장비 고유의 IP를 지정합니다.

deviceIF.filerlist

파일 시스템에 마운트된 SnapLock 의 논리 볼륨명과 마운트 포인트, filer 등이 복수개로 구성될 경우 별도의 구성 리스트로 지정할 수 있습니다. 이 파라미터에 별도로 지정된 구성 리스트 파일의 위치를 지정합니다. 구성 리스트 작성 시에는 각 항목의 순서를 반드시 지켜야 합니다.

deviceIF.filercomparelength

filer가 복수개로 구성될 경우 논리 볼륨이 속한 filer 정보를 추출하기 위하여 filer list 파일로부터 논리 볼륨(volume) 항목의 값을 읽어 비교해야 하는데 시스템 구성에 따라 논리 볼륨의 값이 불필요하게 길어질 수 있으므로 값 비교를 위한 유효 범위를 지정할 수 있도록 지원하는 파라미터입니다.

deviceIF.checkhealth

해당 filer가 정상적으로 구동하고 있는지의 여부를 점검 하도록 지정합니다.

기본 값: “true”

deviceIF.checkport

해당 filer가 정상적으로 구동하고 있는지의 여부를 점검 하기위한 TCP/IP 서비스 포트 번호를 지정합니다.

기본 값: “2049”

deviceIF.checktimeout

해당 filer가 정상적으로 구동하고 있는지의 여부를 판단하기 위한 응답 제한 시간을 지정합니다.
(단위: milliseconds)

기본 값: “1000” (1 초)

deviceIF.user

해당 filer에 접속할 사용자 아이디 정보를 지정합니다.

deviceIF.passwd

해당 filer에 접속할 사용자 패스워드 정보를 지정합니다.

deviceIF.info

해당 filer에서 수행되는 트랜잭션의 디버깅 정보에 대한 로깅 여부를 지정합니다. 활성화 될 경우 표준 출력으로 디버깅 정보가 출력됩니다.

기본 값: "false"

구성 예)

```
<agent name="MAIN" type="ARCHIVE" desc="Local Disk"
class="com.windfire.agents.archive.asysAgentArchive" opclass="/serMet/asysservMets.asysOperArchive"
msgfile="../../msg/archive.mc">
  <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaSnapLock"
basedir="d:/STORAGE/MAIN" hashdirs="500" retentiontype="0" retention="1800"
checkhealth="false" MOUNTPT="" VOLUME="" filerlist="filer1.lst" filercomparelength="11"
FILER="10.10.1.7" USER="root" PASSWD="hanaceda" INFO="TRUE"/>
    <threadpool ops="read,write,delete" count="50"/>
  </scheduler>
</agent>
```

리스트 파일 예)

filer1.lst

논리볼륨(volume),IP(filer),마운트위치(mountpt)

/vol/ssar01,10.10.1.1,/vol/Stg1

/vol/ssar02,10.10.1.2,/vol/Stg2

/vol/ssar03,10.10.1.3,/vol/Stg3

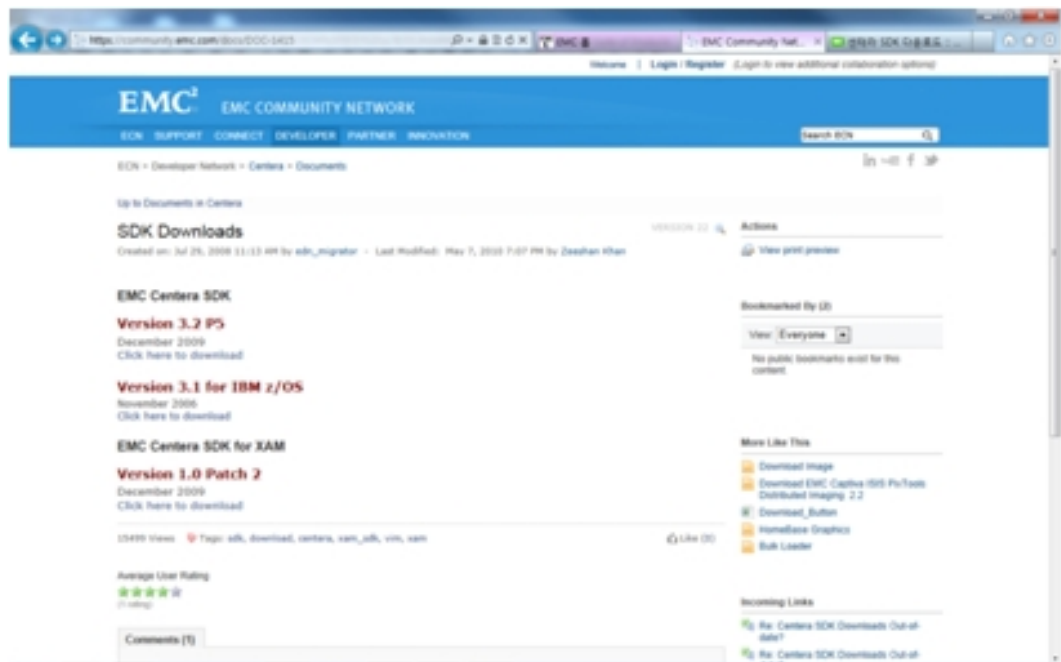
EMC Centera

인터페이스 방식

EMC Centera SDK(JNI 방식)를 통해 TCP/IP 소켓기반으로 연동

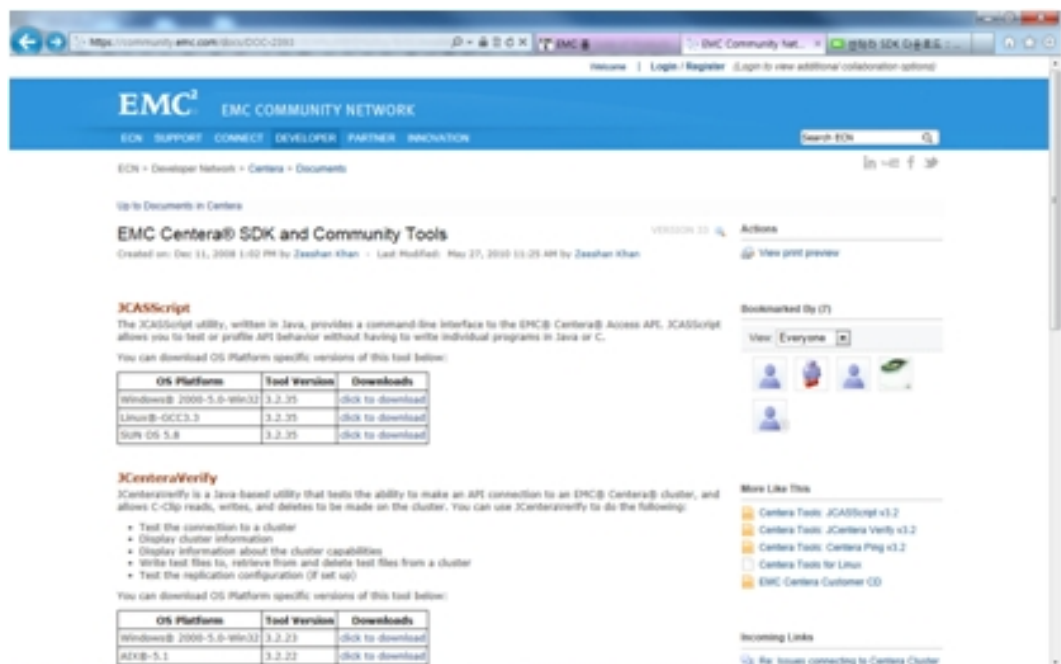
Centera SDK 설치

OS 별 전용 SDK를 다운받아 서버에 설치, 회원가입 필요
URL : <https://community.emc.com/docs/DOC-1415>



Centera Tools

EMC COMMUNITY NETWORK (개발자 커뮤니티 사이트)
 JCASScript, JCenteraVerify, Centera Ping
 URL : <https://community.emc.com/docs/DOC-2393>



EMC Centera 연동

연동 및 기능

C 기반 라이브러리를 JNI 방식으로 호출하여 연동.
노드 간 로드밸런싱(Round-Robin) 및 Fail-Over 기능 및 캐쉬(Cache) 기능 등 지원.

환경 설정

각 OS별 엔진실행할 자바버전에 맞게 LIBRARY_PATH 설정 해준다.

```
솔라리스: LD_LIBRARY_PATH
AIX      : LIBPATH
HP-UX    : SHLIB_PATH
Max os   : DYLD_LIBRARY_PATH
Linux    : LD_LIBRARY_PATH
```

.profile 설정 하기

```
export
LD_LIBRARY_PATH=/usr/lib:/usr/openwin/lib:/usr/local/lib:/work/eims/inzi:/usr/local/Centera_SDK/lib/32:/usr
/local/Centera_SDK/lib/64
자바 버전이 32bit 이면 /32 잡아주고 64비트이면 /64 잡아준다.
```

실행파일 설정 예제

Xtormctl

```
#!/bin/sh
# export LD_LIBRARY_PATH=/Xtorm/Centera_SDK3.1/lib/64
export SHLIB_PATH=/usr/lib:/Xtorm/Centera_SDK3.1/lib/64

Xtorm_HOME=/Xtorm/Xtormbr
SERVICE_HOME=/Xtorm/Xtormbr/servers/Xtorm
JAVA_HOME=/opt/java1.5/bin

# Classpath should be current directory for com.windfire...
CLASSPATH=$Xtorm_HOME/lib/Xtorm.jar
.....
CLASSPATH=/Xtorm/Centera_SDK3.1/lib/FPLibrary.jar:$CLASSPATH

JVM_OPT="-Du=Xtormbr -d64 -server -Xms512m -Xmx2048m -Xss300k -Xrs"
# Set current directory before launching java
cd $SERVICE_HOME

case "$1" in
  'start') # Start Xtorm
    echo "Starting Xtorm BR Service....: "
    sleep 3
    # cp system.out system_out/system.out_`date +%Y%m%d%H%M%S`
    $JAVA_HOME/java -cp $CLASSPATH Changelp 172.31.38.110 2102
    $JAVA_HOME/java $JVM_OPT -cp $CLASSPATH
    com.windfire.agents.asysStartup conf_br.xml AUTOSTART 1> system.out 2>&1 &
    RVAL=$?
    sleep 5
    echo $RVAL
    exit $RVAL
```

```

;;
'stop') # Stop Xtorm
    echo "Stopping Xtorm BR Service...: "
    cp system.out system_out/system.out_`date +%Y%m%d%H%M%S`
    $JAVA_HOME/java -cp $CLASSPATH com.windfire.agents.asysStop
172.31.38.110 2102
    sleep 5
    RVAL=$?
    echo $RVAL
    exit $RVAL
;;
.....
*)
    echo Usage: `basename $0` "{start|stop|restart|status}"
    exit 1
;;
esac

```

HIS HCP

인터페이스 방식

WebDAV 프로토콜 방식과 HTTP 프로토콜 방식 지원

WebDAV 방식

WebDAV Client SDK (Jakarta Slide WebDAV Client)를 통해 WebDAV 프로토콜 기반으로 연동. 보안 취약점 등의 문제로 적용 중인 고객 사이트는 없음.
(현재 Slide 프로젝트는 종료되었으며 JackRabbit 이 활성화 되어 있음)

URL : <http://jackrabbit.apache.org/downloads.html>



HTTP 방식

HTTP Client SDK (Jakarta Commons Http Client)를 통해 HTTP 프로토콜 기반으로 연동

URL : <http://hc.apache.org/downloads.cgi>



HIS HCP 연동

연동 및 기능

WebDAV Client 및 HTTP Client JAVA 라이브러리를 호출하여 연동.
노드 간 로드밸런싱(Round-Robin) 및 Fail-Over 기능, 묶음 처리, 스트리밍 서비스 등 지원.

HCP jar파일 추가

엔진 CLASSPATH 에 commons-logging.jar , commons-httpclient-3.1.jar,jakarta-slide-webdavlib-2.1.jar,commons-codec-1.4.jar
추가 해준다.

실행파일 설정 여제

Xtormctl

```
#!/bin/sh
```

```
Xtorm_HOME=/Xtorm/Xtormbr
```

```
SERVICE_HOME=/Xtorm/Xtormbr/servers/Xtorm
```

```

JAVA_HOME=/opt/java1.5/bin

# Classpath should be current directory for com.windfire...
CLASSPATH=$Xtorm_HOME/lib/Xtorm.jar
.....
# for WebDAV
#CLASSPATH=/Xtorm/lib/ext/ jakarta- slide-webdavlib-2.1.jar :$CLASSPATH
# for HTTP
CLASSPATH=/Xtorm/lib/ext/ commons-httpclient.jar:/Xtorm/lib/ext/ c commons-
logging.jar:$CLASSPATH

JVM_OPT="-Du=Xtormbr -d64 -server -Xms512m -Xmx2048m -Xss300k -Xrs"
# Set current directory before launching java
cd $SERVICE_HOME

case "$1" in
    'start') # Start Xtorm
        echo "Starting Xtorm BR Service...: "
        sleep 3
        # cp system.out system_out/system.out_`date "+%Y%m%d%H%M%S"`
        $JAVA_HOME/java -cp $CLASSPATH Changelp 172.31.38.110 2102
        $JAVA_HOME/java $JVM_OPT -cp $CLASSPATH
        com.windfire.agents.asysStartup conf_br.xml AUTOSTART 1> system.out 2>&1 &
        RVAL=$?
        sleep 5
        echo $RVAL
        exit $RVAL
        ;;
    'stop') # Stop Xtorm
        echo "Stopping Xtorm BR Service...: "
        cp system.out system_out/system.out_`date "+%Y%m%d%H%M%S"`
        $JAVA_HOME/java -cp $CLASSPATH com.windfire.agents.asysStop
        172.31.38.110 2102
        sleep 5
        RVAL=$?
        echo $RVAL
        exit $RVAL
        ;;
    *)
        echo Usage: `basename $0` "{start|stop|restart|status}"
        exit 1
        ;;
    esac

```

필터 사용법

필터 정의

아카이브 에이전트가 파일에 대한 입/출력 처리도중 엔진 기본 기능 이외에 파일에 대한 부수적인 가공(암/복호화 등)을 위해 사용자에게 의하여 구현된 자바 클래스.

필터 종류

입력용 : com.windfire.agents.data.asys.ext.asysExtArchivePut
 조회용 : com.windfire.agents.data.asys.ext.asysExtArchiveGet

필터 구현체 작성

예제1 - 입력용)

```
package filter;
import com.windfire.agents.data.asys.ext.asysExtArchivePut;
public class Encrypt extends asysExtArchivePut{
    // Constructor
    public Encrypt(){

    }

    public boolean putFilter(String srcFile, String parms, StringBuffer err, Object pool){
        System.out.println("filter.Encrypt called --> " + srcFile);
        System.out.println("filter.Encrypt params : " + parms);
        // true or false를 반환, 파일 경로는 불변
        return true;
    }
}
```

예제2 - 조회용)

```
package filter;
import com.windfire.agents.data.asys.ext.asysExtArchiveGet;
public class Decrypt extends asysExtArchiveGet{
    // Constructor
    public Decrypt(){

    }

    public String retrieveFilter(String srcFile, String parms, StringBuffer err){
        System.out.println("filter.Decrypt called --> " + srcFile);
        System.out.println("filter.Decrypt params : " + parms);
        // 파일처리 이후 최종 저장된 파일의 경로를 반환
        return srcFile;
    }
}
```

서버 클래스 설정 및 구동

Xtormctl

```
#!/bin/sh

Xtorm_HOME=/Xtorm/Xtormbr
SERVICE_HOME=/Xtorm/Xtormbr/servers/Xtorm
JAVA_HOME=/opt/java1.5/bin

# Classpath should be current directory for com.windfire...
CLASSPATH=$Xtorm_HOME/lib/Xtorm.jar
.....
# for Filter (/Xtorm/lib 하위에 filter/ Encrypt .class 파일이 존재)
CLASSPATH=/Xtorm/lib:$CLASSPATH

JVM_OPT="-Du=Xtormbr -d64 -server -Xms512m -Xmx2048m -Xss300k -Xrs"
# Set current directory before launching java
cd $SERVICE_HOME
```



```

case "$1" in
    'start') # Start Xtorm
        echo "Starting Xtorm BR Service...: "
        sleep 3
        # cp system.out system_out/system.out_`date "+%Y%m%d%H%M%S"`
        $JAVA_HOME/java -cp $CLASSPATH Changelp 172.31.38.110 2102
        $JAVA_HOME/java $JVM_OPT -cp $CLASSPATH
        com.windfire.agents.asysStartup conf_br.xml AUTOSTART 1> system.out 2>&1 &
        RVAL=$?
        sleep 5
        echo $RVAL
        exit $RVAL
        ;;
    'stop') # Stop Xtorm
        echo "Stopping Xtorm BR Service...: "
        cp system.out system_out/system.out_`date "+%Y%m%d%H%M%S"`
        $JAVA_HOME/java -cp $CLASSPATH com.windfire.agents.asysStop
        172.31.38.110 2102
        sleep 5
        RVAL=$?
        echo $RVAL
        exit $RVAL
        ;;
    .....
    *)
        echo Usage: `basename $0` "{start|stop|restart|status}"
        exit 1
        ;;
esac

```

API를 통한 필터 사용

입력 여제)

```

asysUsrElement uePage1 = new asysUsrElement(con);
uePage1.m_localFile="C:/Temp/trComm.txt";
uePage1.m_descr = "ScanedImage";
uePage1.m_cClassId = "BASIC";
uePage1.m_userSClass = "NONE";
uePage1.m_eClassId = "IMAGE";
uePage1.addIndexValue("DOC", "f1", "1");
uePage1.addIndexValue("DOC", "f2", "1");
int ret = uePage1.create("Xtorm_MAIN", "Encrypt", "myPutFilterTest");
// Check for errors - last message always in API object
if (ret != 0)
    System.out.println("Error, createFilter, " + uePage1.getLastError());
else
    System.out.println("Success, createFilter, " + uePage1.m_elementId);

```

조화 여제)

```

asysUsrElement uePage1 = new asysUsrElement(con);
uePage1.m_elementId="Xtorm_MAIN::2011081613280302::IMAGE";

```

```
int ret = uePage1.getContent("C:/Temp/test.tif", "Decrypt", "1234");
if (ret != 0)
    System.out.println("Error, download normal, " + uePage1.m_lastError);
else
    System.out.println("Success, download normal, " + uePage1.m_elementId);
```

서버

기본 메시지 파일 및 로그 작성기와 같은 서버에 대한 일반적인 정보를 정의합니다.

속성

server

현재 서버(컴퓨터)에 설치되어 있는 Xtorm에 대한 일반적인 정보를 정의합니다.

server.basemsg

Xtorm 기본 메시지 즉, 각종 정보, 경고, 에러 메시지를 표시하기 위한 기본 메시지 파일 정의

기본 값: "msg/base.mc"

server.deleteconf

환경 파일의 삭제 여부를 설정합니다.

기본 값: "false"

logwriter

로그 작성기를 정의. Xtorm은 동시에 하나 이상의 로그 작성기를 지원합니다. 각각의 로그 작성기는 각각 다른 목적지에 로그 메시지를 보냅니다. 대상 목적지로는 콘솔, 로그 파일 또는 윈도우 이벤트 뷰어의 어플리케이션 로그 등이 될 수 있습니다.

logwriter.class

로그를 기록하는 데에 사용되는 자바 클래스 이름으로 다음의 4 가자가 있습니다.

파일: "com.windfire.agents.asysLogWriterFile", "com.windfire.agents.asysLogWriterLog4j"

NT 이벤트 뷰어: "com.windfire.agents.asysLogWriterNTEvent"

콘솔: "com.windfire.agents.asysLogWriterConsole"

참고: 이 모두는 설치 시 기본으로 구성됩니다.

file 파일 및 NT 이벤트 로그 작성기에만 해당되며, 로그 파일에 대한 특정한 정보를 정의합니다.

logwriter.level

로그 레벨을 구분할 수 있도록 지원하며 전체 레벨 수준은 다음과 같이 6 가지로 구분됩니다.

로그레벨 1 : 환경설정 부분만 로그에 남김

로그레벨 2 : GET / PUT / DELETE / REPLACE 부분 로그 남김

로그레벨 3 : 볼륨정보 체크 및 스케줄 쓰레드 동작 로그 남김

로그레벨 5 : 기존과 동일 하며 추가로 (SIMPLESQL 및 TRANSACTSQL 쿼리 로그 남김)

기본값: "0"

file.name

서버 로그 메시지를 저장할 파일 이름. 윈도우의 경우 이벤트 로그 및 파일 로그는 동일한 내용이 기록됩니다.

기본 값: "log/server.log"

다음의 속성은 "asysLogWriterFile"에만 유효합니다. 서버 파일 로그는 기본이 유니코드가 아니지만, 선택적으로 유니코드로 세팅할 수 있습니다.

file.encoding

로그 내용에 유니코드 문자를 가능하게 합니다. 속성을 특별히 지정하지 않으면 로그 파일은 single byte 로 기록되며, XML 로그 파일의 경우는 ISO-8859-1로 인코딩 됩니다. encoding="Unicode"을 지정하면 로그 파일은 Unicode로 기록되며, XML 로그 파일은 UTF-16 (Unicode)으로 인코딩 됩니다.

기본 값: ""

또한, 다음의 파라미터를 이용하여 서버에서 날짜/시간 표시 형식을 지정할 수 있습니다.

file.dateformat

로그 파일에서 사용할 날짜 형식. 자바의 "SimpleDateFormat" 클래스에서 지원하는 모든 날짜 형식 가능.

기본 값: "dateformat="yyyy-MM-dd 'at' HH:mm:ss""

한국 또는 일본 지역에서 설정 시 "yyyy-MM-dd a hh:mm:ss"로 설정하면, a 부분에 오전 또는 오후가 표시됩니다.

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다.

만약 위의 파라미터를 정의한 경우 서버를 시작하기 전에 기존의 로그 파일을 지우십시오.

file.tmode

서버로그 파일을 분리하는 기준을 지정 합니다.

(1: Size, 2: Date)

file.tparam

tmode가 '1'인 경우 -

서버로그를 구분할 파일의 크기를 지정해야함.

지정되지 않을 경우 기본값은 10000Kb.

(단위: Kb)

tmode가 '2'인 경우 -

서버로그를 구분할 일자의 유형을 지정해야함.

월단위 - 'monthly'
 주단위 - 'weekly'
 일단위 - 'daily'

debugtracer

에이전트 단위로 디버깅 용도의 로그를 저장하는 기능을 제공합니다.

debugtracer.class

디버그 로그를 기록하는 자바 클래스 이름으로 현재 아래의 버전이 제공됩니다.

파일: "com.windfire.agents.asysDebugTracerFile"

debugger

에이전트 단위로 디버깅 용도의 로그를 저장하는 기능을 제공합니다.

debugger.id

에이전트 단위로 사용될 디버거의 이름을 정의 합니다.

debugger.file

디버거가 사용할 파일의 경로를 지정 합니다.

debugger.size

디버거가 사용할 파일의 최대 크기를 지정 합니다.

(단위: Kb)

debugger.encoding

디버깅 로그 내용에 유니코드 문자를 가능하게 합니다. 속성을 특별히 지정하지 않으면 로그 파일은 single byte로 기록되며, XML 로그 파일의 경우는 ISO-8859-1로 인코딩 됩니다.
 encoding="Unicode"을 지정하면 로그 파일은 Unicode로 기록되며, XML 로그 파일은 UTF-16 (Unicode)으로 인코딩 됩니다.

기본 값: ""

또한, 다음의 파라미터를 이용하여 서버에서 날짜/시간 표시 형식을 지정할 수 있습니다.

debugger.dateformat

로그 파일에서 사용할 날짜 형식. 자바의 "SimpleDateFormat" 클래스에서 지원하는 모든 날짜 형식 가능.

기본 값: "dateformat="yyyy-MM-dd 'at' HH:mm:ss""

한국 또는 일본 지역에서 설정 시 "yyyy-MM-dd a hh:mm:ss"로 설정하면, a 부분에 오전 또는 오후가 표시됩니다.

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다.

만약, 위의 파라미터를 정의한 경우 서버를 시작하기 전에 기존의 로그 파일을 지우십시오.

예:

```
<server basemsg="msg/base.mc" deleteconf="false">
  <logwriter class="com.windfire.agents.asysLogWriterFile" level="5">
    <file name="log/server.log" truncate="false" encoding="Unicode" dateformat="yyyy-MM-dd
'at' HH:mm:ss" tmode="1" tparam="10000"/>
  </logwriter>

  <!--
  <logwriter class="com.windfire.agents.asysLogWriterNTEvent">
    <file name="log/server.log" truncate="false"/>
  </logwriter>
  -->

  <logwriter class="com.windfire.agents.asysLogWriterConsole"/>
    <debugtracer class="com.windfire.agents.asysDebugTracerFile">
      <debugger id="debug1" file="log/debug.log" size="10240" encoding="Unicode"
dateformat="yyyy-MM-dd 'at' HH:mm:ss"/>
      <debugger id="debug2" file="log/migDebug.log" size="10240"/>
    </debugtracer>
  </logwriter>
</server>
```

Log4j 설정

❖ asysLogWriterLog4j (Log4j 지원)

설정방법

- 1) conf.xml 구성
- 2) JAVA CLASSPATH 상에 log.properties 파일 저장
- 3) log.properties 파일 구성
 - <권장 – Daily Rolling>

```
log4j.logger.main=INFO, R
```

```
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%5p %d{[yyyy.MM.dd,HH:mm:ss]} - %m
%n
```

```
log4j.appender.R=org.apache.log4j.DailyRollingFileAppender
log4j.appender.R.File=./log/server_2102.log
log4j.appender.R.ImmediateFlush=true
log4j.appender.R.Append=true
log4j.appender.R.DatePattern='.[yyyy-MM-dd
```

```
log4j.appender.R.layout=org.apache.log4j.PatternLayout
log4j.appender.R.layout.ConversionPattern=%d{[yyyy.MM.dd,HH:mm:ss]} - %m%n
```

- 4) windows 환경에서는 asysLogWriterConsole 은 특별한 경우가 아닌 이상 주석 처리 해야 함.
성능에 영향 있음.

```
<logwriter class="com.windfire.agents.asysLogWriterConsole" systemout="System.out" />
```

예:

```
<server basemsg="../../msg/base.mc">
  <logwriter class="com.windfire.agents.asysLogWriterFile">
    <file name="log/server.log" truncate="false"/>
  </logwriter>

  <!--
    <logwriter class="com.windfire.agents.asysLogWriterLog4j" level="5">
      <property name="log.properties" section="main"/>
    </logwriter>
  -->
  <logwriter class="com.windfire.agents.asysLogWriterConsole" systemout="System.out" />
</server>

<localagents machineid="0">
...
</localagents>
```

Audit 에이전트(유형: AUDIT)

속성

AUDIT 유형의 에이전트는 AUDIT 유형의 포스트 프로세서에 의해 캡처된 Xform 트랜잭션을 감시하고 하나 또는 하나 이상의 미디어에 기록을 작성합니다. 특정 미디어에 필터하고 기록하는 모듈인 "Audit 작성기(audit writers)"를 통하여 해당 트랜잭션들은 기록됩니다. 현재 AUDIT 에이전트는 파일, 버퍼, 테이블과 같은 유형을 지원합니다.

auditwriter

Audit 작성기와 관련된 구성 정보를 정의합니다. 하나의 Audit 에이전트 내에서 동시에 동일한 유형의 또는 다른 유형의 작성기를 여러 개 운영할 수 있습니다.

name

Audit 작성기의 고유 이름. 필요에 따라 컴포넌트를 설명하는 어떠한 이름이라도 지정 가능합니다. 동일한 유형을 여러 개 지정하거나, 서로 다른 유형을 정의하는 모든 경우에 이름은 고유해야 합니다. 이 속성은 Xform 운영 어플리케이션에서 표시 목적으로만 사용됩니다.

desc

Audit 작성기에 대한 간략한 설명을 입력합니다.

class

Audit 작성기에 대한 로직을 포함하고 있는 자바 클래스 또는 모듈 이름

파일: "com.windfire.agents.audit.asysAgentAuditWriterFile"

버퍼: "com.windfire.agents.audit.asysAgentAuditWriterBuffer"

테이블: "com.windfire.agents.audit.asysAgentAuditWriterTable"

encoding

Audit 로그 내용에 유니코드 문자를 가능하게 합니다. 속성을 특별히 지정하지 않으면 로그 파일은 single byte로 기록되며, XML 로그 파일의 경우는 ISO-8859-1로 인코딩 됩니다. encoding="Unicode"을 지정하면 로그 파일은 Unicode로 기록되며, XML 로그 파일은 UTF-16 (Unicode)으로 인코딩 됩니다.

기본 값: "ISO-8859-1"

opclass

웹 운영 어플리케이션에서 Audit 작성기와 인터페이스할 서블릿 이름

파일: "/servlet/asysservlets.asysOperAuditFile"

버퍼: "/servlet/asysservlets.asysOperAuditBuffer"

테이블: "/servlet/asysservlets.asysOperAuditTable"

id

Audit 작성기를 실제로 고유하게 구분하는 고유 ID. 하나의 Xtorm 서비스 내에서 동시에 동일한 유형의 또는 다른 유형의 작성기를 여러 개 운영할 수 있습니다. Audit 포스트 프로세서가 트랜잭션을 캡처할 때 하나 또는 하나 이상의 Audit 작성기와 이 ID를 이용하여 통신합니다.

다음의 속성들은 "파일" 및 "버퍼" 유형의 Audit 작성기에만 유효합니다.

dateformat

"파일" 및 "버퍼" 유형의 Audit 작성기에서 사용할 날짜 및 시간 형식. 자바의 "SimpleDateFormat" 클래스에서 지원하는 모든 날짜 형식 가능.

기본 값: "dateformat="yyyy-MM-dd 'at' HH:mm:ss""

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다. "hh"를 사용할 경우, 오전(a.m.) 인지 오후(p.m.)인지를 명시해야 합니다. (예: dateformat="yyyy-MM-dd 'at' hh:mm:ss a")

참고: "테이블" 유형의 Audit 작성기는 "dateformat" 속성을 필요로 하지 않습니다. 데이터베이스에 설정되어 있는 형식을 사용해야 하기 때문입니다. 만약, 테이블에 기록되는 날짜 형식을 변경하고자 하는 경우는 데이터베이스 변수를 변경하십시오.

다음의 속성들은 "테이블" 유형의 Audit 작성기에만 유효합니다.

agentData

테이블에 Audit 정보를 기록하는 일에 사용할 데이터 에이전트의 이름

기본 값: "DATA"

agentDataServer

"sqlgateway" 속성에서 정의한 게이트웨이가 실행중인 Xtorm 서버의 이름을 정의. MACHINE:PORT 형식으로 정의합니다. MACHINE은 호스트 이름 또는 IP 주소이며, PORT는 Xtorm 서버의 리스닝 포트 번호입니다. 공란으로 지정하면 기본 값을 사용합니다.

기본 값: "localhost:2102"

sqlgateway

Audit 작성기가 테이블에 정보를 기록하는 일에 사용하는 SIMPLESQL 유형의 게이트웨이 인스턴스를 지정합니다.

기본 값: "SIMPLESQL_MAIN"

dbtype

데이터베이스 유형을 지칭하는 플래그. 아무 값도 지정하지 않으면 Microsoft SQL Server가 기본으로 사용됩니다.

1. Microsoft SQL Server
2. Oracle
3. DB2
4. Sybase
5. MySQL
6. Altibase
7. Informix
8. Tiberio
9. PostgreSQL

다음의 속성들은 "파일" 유형의 Audit 작성기에만 유효합니다.

file

파일 유형의 Audit 작성기에 대한 특정 구성 정보를 정의

filename

Audit 정보를 저장할 파일의 이름

basedir

Audit 정보 저장 파일을 위치시킬 경로. 시스템 상에서 접근할 수 있는 올바른 경로를 지정합니다.

권장 값: "C:/SpenoCom/Xtorm/log"

maxsize

Audit 파일의 최대 크기(Kb). 최대 크기에 도달하면 "deleteonfull"에서 정의한 내용에 따라 파일을 삭제

하거나, 파일 이름을 변경할 것입니다.

deleteonfull

Audit 파일이 최대 크기에 도달하면 파일을 삭제할 것인지를 정의합니다.

Yes 파일을 삭제한 후 동일한 이름의 새로운 파일을 생성합니다.

No 파일을 새로운 이름으로 변경한 후, 원래의 이름으로 새로운 파일이 생성됩니다. 변경되는 파일의 이름은 "filename_datetime.ext" 형식입니다.

fielddelimiter

파일에 정보를 기록할 때 Audit 필드 간에 문자열을 구분하는 구분자를 정의합니다.

field

개별 트랜잭션을 캡처할 필드를 지정합니다. 지정할 필드마다 한 라인으로 정의합니다.

detail

모든 트랜잭션의 공통 필드들 중 명시적으로 지정된 필드를 제외한 나머지 필드 정보를 의미합니다.

예: DESC="HMFOLDER" elementid="Xtorm_MAIN::200910191929291::FOLDER"
FILTER="" id="Xtorm_MAIN" cmd="200" ECLA="FOLDER" SCLA="ACL"
FILTERPARMS=""

설정 예: detail="**"

다음의 속성들은 "버퍼" 유형의 Audit 작성기에만 유효합니다.

buffer

버퍼 유형의 Audit 작성기에 대한 특정 구성 정보를 정의

maxrecord

버퍼 상에 유지할 레코드의 최대 수를 정의. 최대 수를 초과하여 트랜잭션이 캡처되는 경우 오래된 순서대로 레코드가 삭제됩니다.

field

개별 트랜잭션을 캡처할 필드를 지정합니다. 지정할 필드마다 한 라인으로 정의합니다.

detail

모든 트랜잭션의 공통 필드들 중 명시적으로 지정된 필드를 제외한 나머지 필드 정보를 의미합니다.

예: DESC="HMFOLDER" elementid="Xtorm_MAIN::200910191929291::FOLDER"
FILTER="" id="Xtorm_MAIN" cmd="200" ECLA="FOLDER" SCLA="ACL"

`FILTERPARMS=""`

설정 예: `detail="**"`

다음의 속성들은 “테이블” 유형의 Audit 작성기에만 유효합니다.

table

테이블 유형의 Audit 작성기에 대한 특정 구성 정보를 정의

tablename

Audit 정보를 저장할 데이터베이스 테이블 이름을 정의. Audit 에이전트는 Xtorm SIMPLESQL 게이트웨이를 사용하여 테이블에 데이터를 기록합니다. 지정된 테이블이 존재하지 않으면 Xtorm이 자동으로 테이블을 생성합니다.

column

트랜잭션을 캡처하여 저장할 필드들에 대한 테이블의 컬럼을 정의합니다. 각각의 컬럼에 대하여 한 라인에서 정의합니다.

length

테이블 컬럼의 길이를 정의합니다.

type

테이블 컬럼의 유형을 정의합니다.

etc

모든 트랜잭션의 공통 필드들 중 명시적으로 지정된 필드를 제외한 나머지 필드 정보를 의미합니다.

예: `DESC="HMFOLDER" elementid="Xtorm_MAIN::200910191929291::FOLDER"`
`FILTER="" id="Xtorm_MAIN" cmd="200" ECLA="FOLDER" SCLA="ACL"`
`FILTERPARMS=""`

설정 예: `etc="**"`

“field” 또는 “column” 속성 이후에 존재하는 다음의 속성들은 파일, 버퍼 또는 테이블에 필드 또는 컬럼을 기록할 때 사용하는 타이틀 정보입니다. 이들 타이틀은 의미있는 어떠한 이름도 지정 가능합니다. 필드 또는 컬럼에 따라 의미를 표시할 수 있는 적절한 내용이어야 하며, 공란이나 특수 문자를 포함할 수는 없습니다. 이곳에서 정의하는 필드 또는 컬럼에 지정되는 값들은 반드시 Xtorm 트랜잭션으로부터 캡처할 수 있는 필드이어야 합니다.

다음은 모든 트랜잭션에서 사용할 수 있는 공통 필드들입니다:

subCmd

Xtorm 트랜잭션에 의해 사용된 서브 명령. 이 값은 숫자이며 Xtorm 내부적으로 사용되는 값입니다. 예: 1303, 1140, 1001, ...

설정 예: subCmd="subCmd"

Gateid

트랜잭션이 발생한 게이트웨이 ID. 예: Xtorm_MAIN, DASL_MAIN, ...

설정 예: Gateid="engine"

srcAgent

트랜잭션을 처리한 소스 에이전트. 주로 트랜잭션을 처리한 데이터 에이전트를 의미합니다. 예: DATA.

설정 예: srcAgent="srcAgent"

srcMach

트랜잭션을 처리한 소스 서버 정보. 데이터 에이전트가 위치한 Xtorm 서버 정보를 의미합니다. 예: NorthServer:2102.

설정 예: srcMach="srcMach"

userid

트랜잭션을 처리한 사용자 ID. 예: SUPER, Mike, Lisa, ...

설정 예: userid="userid"

requestMach

트랜잭션이 보내진 목적지 서버 정보. 트랜잭션이 전달된 Xtorm 서버 또는 클라이언트 어플리케이션을 의미합니다. 예: HDUAL:2102

설정 예: requestMach="targetMach"

requestDetail

트랜잭션이 보내진 목적지 서버의 세부 정보. 트랜잭션이 전달된 Xtorm 서버 또는 클라이언트 어플리케이션의 세부 정보를 의미합니다.

예: ARCHIVE, Win32 Api, ...

설정 예: requestDetail="targetAgent"

downIpAddr

파일에 대한 다운로드 트랜잭션에 대하여 다운로드를 요청한 클라이언트의 IP 정보를 의미합니다. 예: 10.1.1.2

설정 예: downIpAddr="ipaddr"

downHostName

파일에 대한 다운로드 트랜잭션에 대하여 다운로드를 요청한 클라이언트의 호스트 정보를 의미합니다. 예: HDUAL,Lisa, ...

설정 예: downHostName="hostname"

elementId

트랜잭션의 대상이 되는 엘리먼트의 내부 ID 정보를 의미합니다. 배치 트랜잭션의 경우 엘리먼트가 ‘, 로 구분되어 복수개로 구성될 수 있습니다.

예: 201002231123140,201002231123150,201002231123160

설정 예: elementId="dstEid"

서로 다른 트랜잭션에 특정 필드들을 정의할 수 있습니다. 하나 또는 하나 이상의 이러한 필드들이 트랜잭션 내부에 존재할 수 있습니다.

참고: 이 부분에서는 적용 가능한 모든 필드 값들을 설명하진 않습니다. 모든 가능한 필드들을 알고자 하는 경우는 최소 하나 이상의 트랜잭션을 캡처하여 버퍼 유형의 Audit 작성기에 보내십시오. 결과를 보고 캡처할 필드들을 결정하십시오. 캡처할 필요성에 따라 하나 또는 하나 이상의 필드/컬럼 속성을 생성함으로써 이러한 필드들을 캡처할 수 있습니다.

* 공통 필드 이외에, 트랜잭션의 부분을 이루고 있는 모든 기타 필드

rcode

트랜잭션의 반환 코드

cmd

트랜잭션의 명령 ID

transid

트랜잭션의 트랜잭션 ID

targetAgent

트랜잭션이 전달된 목적지 에이전트

reason

트랜잭션에 에러가 발생한 경우, 표시된 에러 메시지 번호

errmsg

실제 에러 메시지

sql

검색 트랜잭션에 사용된 SQL 문장

예:

```
<agent name="AUDIT" type="AUDIT" desc="Audit agent"
class="com.windfire.agents.audit.asysAgentAudit" opclass="/serMet/asysservlets.asysOperAudit"
msgfile="msg/audit.mc">

    <auditwriter name="Audit file writer" desc="Audit file writer"
class="com.windfire.agents.audit.asysAgentAuditWriterFile" encoding="ISO-8859-1" dateformat="yyyy-MM-
dd HH:mm:ss" opclass="/serMet/asysservlets.asysOperAuditFile" id="file">
    <file basedir=" C:/SpenoCom/Xtorm/log" maxsize="1000" deleteonfull="YES"
fielddelimiter="|" filename="auditfile.log">
        <field subCmd="subCmd" />
        <field Gateid="id" />
        <field srcAgent="srcAgent" />
        <field srcMach="srcMach" />
        <field userid="userid" />
        <field detail="*" />
    </file>
</auditwriter>

    <auditwriter name="Audit buffer writer" desc="Audit buffer writer"
class="com.windfire.agents.audit.asysAgentAuditWriterBuffer" encoding="ISO-8859-1" dateformat="yyyy-
MM-dd HH:mm:ss" opclass="/serMet/asysservlets.asysOperAuditBuffer" id="buffer">
    <buffer maxrecord="100">
        <field subCmd="subCmd" />
        <field Gateid="id" />
        <field srcAgent="srcAgent" />
        <field srcMach="srcMach" />
        <field targetMach="targetMach" />
        <field userid="userid" />
        <field detail="*" />
    </buffer>
</auditwriter>

    <auditwriter name="Audit table writer" desc="Audit table writer"
class="com.windfire.agents.audit.asysAgentAuditWriterTable" encoding="unicode"
opclass="/serMet/asysservlets.asysOperAuditTable" agentData="DATA"
agentDataServer="LOCALHOST:2102" sqlgateway="SIMPLESQL_MAIN" dbtype="1" id="table">
    <table tablename="UT_AUDIT" >
```

```

        <column subCmd="subCmd" length="24" type="char" />
        <column Gateid="engine" length="24" type="char" />
        <column srcAgent="srcAgent" length="24" type="char"/>
        <column srcMach="srcMach" length="24" type="char"/>
        <column userid="userid" length="24" type="char"/>
        <column requestMach="targetMach" length="24" type="char"/>
        <column requestDetail="targetAgent" length="24" type="char"/>
        <column downIpAddr="ipaddr" length="24" type="char"/>
        <column downHostName="hostname" length="24" type="char"/>
        <column elementId="dstEid" length="256" type="char"/>
        <column etc="*" length="256" type="varchar"/>
    </table>
</auditwriter>

</agent>

```

Cache Cleanup 에이전트(유형: CACHECLEAN)

속성

logging

Cache Cleanup 에이전트에 대한 로깅 파라미터를 정의

basedir

Cache Cleanup 에이전트에 의해 사용될 로그 파일의 경로를 설정. 로그 파일을 XML 형식으로 작성됩니다.

권장 값: "Xtorm/log"

maxentries

로그 파일 안에 기록한 최대 엔트리 수. 하나의 엔트리 수는 에이전트에 의해 접수된 모든 정보를 포함합니다. 최대 엔트리에 이르면 로그 내용은 뒤에 추가될 것입니다.

권장 값: "10"

dateformat

로그 파일에서 사용할 날짜 형식 지정. 자바의 "SimpleDateFormat" 클래스에서 지원하는 모든 날짜 형식 가능

기본 값: "dateformat="yyyy-MM-dd 'at' HH:mm:ss""

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다. "hh"를 사용할 경우, 오전(a.m.) 인지 오후(p.m.)인지를 명시해야 합니다. (예: dateformat="yyyy-MM-dd 'at' hh:mm:ss a")

encoding

로그 파일을 작성할 때 유니코드를 사용할 지 여부를 지정합니다. 지정하지 않으면 로그 XML 파일은

ISO-8859-1 형식으로 작성되며 지정할 경우 UTF-16 (Unicode) 형식으로 작성됩니다.

기본 값: ""

cache

Cache를 비울 대상 Cache 정보를 정의. 복수의 Cache를 지정할 수 있습니다. 또한, 대상 Cache는 로컬 또는 원격 서버의 Cache가 될 수 있습니다.

agentDataServer

사용될 Information 에이전트를 정의. 아무 이름도 지정되지 않으면 Cache Cleanup 에이전트는 Cache Cleanup 에이전트가 설치된 동일 서버의 2102 포트에 있는 Information 에이전트를 사용할 것입니다.

target

해당 Cache Cleanup 에이전트와 연결되어 있는 게이트웨이 이름 정의

cache

지울 대상 Cache ID 이름 지정 또는 target에 정의한 게이트웨이에 의해 관리되는 모든 Cache를 지울 경우엔 "*"를 입력합니다.

schedule

Cache Cleanup 에이전트를 운영할 스케줄 정의

run

Cache Cleanup 에이전트가 스케줄 기능에 의해 또는 자동으로 운영될 지 여부를 지시하는 플래그 지정

Auto

자동 시작. 이 값을 세팅한 경우에는 Cache Cleanup 에이전트는 언제 시작하고 얼마나 자주 실행하기 위한 정보를 알기 위하여 startafter 및 runevery 속성을 사용할 것입니다. 이 두 속성은 분 단위로 설정합니다.

Schedule

스케줄 기능에 의해 실행함. 이 값을 세팅한 경우에는 Cache Cleanup 에이전트는 스케줄러에 의해 시작됩니다.

startafter

에이전트가 시작되고 나서 Cache 비우기를 시작할 분 단위를 지정합니다. 이 속성은 run 속성의 값이 "Auto"인 경우에만 유효합니다.

runevery

에이전트에 의해 Cache 비우기를 실행할 시간 간격을 분 단위로 지정합니다. 이 속성은 run 속성의 값이 "Auto"인 경우에만 유효합니다.

clean

Cache Cleanup 에이전트에 의해 지워질 필요가 있는 Cache의 양을 지정합니다.

pctfree

지워질 Cache에 의해 사용되고 있는 전체 스토리지의 퍼센티지를 지정합니다. "50"으로 지정하면 Cache를 비운 후에 Cache는 최대 크기의 50%를 여유 크기로 가져야 합니다.

예:

```
<agent name="CACHEMNGR" type="CACHECLEAN" desc="Cache Cleaner"
class="com.windfire.agents.cache.asysAgentCache" opclass="/servlet/asyservlets.asysOperCache"
msgfile="msg/cache.mc" minhealth="1">
  <logging basedir="c:/Xtorm/log" maxentries="10" dateformat="yyyy-MM-dd 'at' HH:mm:ss"
encoding="Unicode" />
  <cache agentDataServer="" target="Xtorm_MAIN" cache="*">
    <schedule run="auto" startafter="10" runevery="30" />
    <clean pctfree="50" />
  </cache>
</agent>
```

IMPORT 에이전트(유형: IMPORT)

속성

startup

IMPORT 에이전트에 대한 사용자 정의 파라미터 정의

bundlesize

한번에 처리할 번들 또는 그룹 내의 엘리먼트 수

권장 값: "50" 또는 "100"

threads

IMPORT 진행 중에 쓰레드 풀 안에서 생성될 쓰레드 수 정의. 이 쓰레드는 번들을 처리하는 일에 사용됩니다.

권장 값: “20”

delete

배치 처리를 성공적으로 진행한 경우 파일을 지울 지 여부를 정의. 배치 안에서 단 하나의 번들에서 에러가 발생하더라도 배치는 삭제되지 않습니다.

Yes 성공적으로 수행된 경우 파일을 삭제함
No 성공적으로 수행된 경우 파일을 삭제하지 않음

권장 값: “no”

postprocess

배치가 끝난 후에 포스트 프로세서를 실행할 지 여부를 정의.

Yes 포스트 프로세서를 실행함
No 포스트 프로세서를 실행하지 않음

권장 값: “yes”

참고: 이 값은 IMPORT 게이트웨이가 포스트 프로세서를 필요로 하는지 여부에 의해 결정됩니다. 제공된 기본 Import 게이트웨이는 처리된 모든 번들의 상태를 변경하기 위하여 포스트 프로세서 실행을 필요로 합니다.

highWaterBundle

한번에 생성될 수 있는 새로운 번들의 최대 수. 이 파라미터는 아카이브 에이전트가 특정 자바 가상 머신에 의해 허용되는 최대 오픈 파일 수를 초과하지 않게 하기 위하여 사용됩니다.

권장 값: “20”

참고: 번들 및 엘리먼트 수위(Watermark)는 자바 가상 머신으로 하여금 너무 많은 파일을 오픈 함으로써 오버로드 되지 않게 하는 IMPORT 프로세스를 조절하는 쓰로틀 밸브의 개념입니다.

lowWaterBundle

IMPORT 에이전트가 새로운 번들을 생성하기를 시작하기 전에 큐에 쌓아 둘 최소 번들 수 정의. 이 숫자에 다다르게 되면 Import 에이전트는 다음 번들을 생성하기 시작합니다.

권장 값: “10”

highWaterElement

항상 프로세스를 진행하는 동안 허용하는 최대 엘리먼트 수. 이 숫자에 다다르게 되면 IMPORT 게이트웨이는 엘리먼트 IMPORT를 중단합니다.

권장 값: “1000”

참고: 번들 및 엘리먼트 수위(Watermark)는 자바 가상 머신으로 하여금 너무 많은 파일을 오픈 함으로써 오버로드 되지 않게 하는 IMPORT 프로세스를 조절하는 쓰로틀 밸브의 개념입니다.

lowWaterElement

IMPORT 게이트웨이 가 새로운 엘리먼트를 IMPORT 하기 전에 큐에 쌓아 둘 최소 엘리먼트 수 정의. 이 숫자에 다다르게 되면 IMPORT 게이트웨이는 엘리먼트 IMPORT 를 재 시작할 것입니다.

권장 값: "800"

agentData

IMPORT를 진행할 데이터 에이전트의 이름

권장 값: "DATA"

target

IMPORT를 진행할 게이트웨이의 이름

권장 값: "Xtorm_MAIN"

agentDataServer

데이터 에이전트가 운영 중인 서버의 이름 및 포트 번호. 이는 IMPORT 에이전트가 데이터 에이전트로부터 분리되어 다른 서버에서 운영될 수 있게 합니다.

권장 값: "machine:port"

encoding

로그 파일을 작성할 때 유니코드를 사용할 지 여부를 지정합니다. 지정하지 않으면 로그 XML 파일은 ISO-8859-1 형식으로 작성되며 지정할 경우 UTF-16 (Unicode) 형식으로 작성됩니다.

기본 값: ""

retryInMins

IMPORT 에이전트가 재시도 상태에 처하였을 때 얼마나 자주 재시도 처리를 해야 하는지를 의미하는 파 리미터입니다. 기본 값은 5 분이며 이 보다 작게 설정할 수 없습니다. 또한, IMPORT 에이전트로 하여금 재시도 모드 및 자동 복원 모드로 전환하도록 하는 여러 코드들을 지정하는 retry 섹션이 있습니다. 이 리 스트에 추가적인 여러 코드를 기입할 수 있습니다.

기본 값: "5"

recovery

IMPORT 프로세스와 관련된 복원 정보 정의

filename

복원 정보가 저장될 파일 이름

권장 값: "recovery.xml"

basedir

복원 정보 파일의 경로

권장 값: "Xtorm/log"

statistic

IMPORT 프로세스와 관련된 통계 정보 정의

filename

통계 정보가 저장될 파일 이름

권장 값: "statistic.xml"

basedir

통계 정보 파일의 경로

권장 값: "Xtorm/log"

ageInDays

통계 정보가 제거되기 전에 통계 파일 안에 저장할 날짜 수

권장 값: "5"

current

현재의 통계 파일 안의 배치 수

권장 값: "15"

error

IMPORT 프로세스와 관련된 에러 정보 정의

filename

에러 정보가 저장될 파일

권장 값: "error.xml"

basedir

에러 정보 파일의 경로

권장 값: "Xtorm/log"

ageInDays

에러 정보가 제거되기 전에 에러 파일 안에 저장할 날짜 수

권장 값: “5”

current

현재의 에러 파일 안의 번들 수

권장 값: “25”

Native IMPORT 게이트웨이(“asysNativeGateway” 및 “asysWindfireGateway”)

importgateway

IMPORT 게이트웨이와 관련된 정보 정의

name

IMPORT 게이트웨이 이름

권장 값: “IMPORTGATEWAY”

class

IMPORT 게이트웨이에 의해 수행될 클래스 이름

권장 값: “com.windfire.agents.importagent.gateway.asysNativeGateway” 또는 “com.windfire.agents.importagent.gateway.asysWindfireGateway”

basedir

IMPORT할 데이터가 있는 기본 디렉토리 경로

참고: 이 파라미터는 실제 운영 환경마다 위치가 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정 하십시오.

eClassId

IMPORT할 엘리먼트의 유형 ID. Xtorm 데이터베이스의 asysEClass 테이블을 참조하십시오.

권장 값: “IMAGE”

userSecureClassId

엘리먼트에 할당된 사용자 보안 클래스 ID. Xtorm 데이터베이스의 asysSecureClass 테이블을 참조하십시오. 보안 설정을 하지 않을 경우 "NONE"으로 세팅 하십시오. 이 항목은 선택적입니다.

참고: 이 항목은 Xtorm의 보안 정책에 의해 사용될 수도 있고 사용되지 않을 수도 있습니다. 보안 관련 문서를 참조 하십시오.

archiveId

IMPORT를 진행하여 파일을 저장할 아카이브 에이전트 ID. 이는 엘리먼트를 어느 스토리지에 생성할 지를 결정합니다. Xtorm 데이터베이스의 asysArchive 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 아카이브가 달라지므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 속성을 세팅하는 경우 contentClassId 속성은 선택적으로 사용될 수 있습니다. 만약, 두 가지 속성을 모두 지정하는 경우 Xtorm은 contentClassId를 이용할 것입니다. 만약 둘 모두 지정하지 않으면 생성되는 엘리먼트는 컨테이너가 될 것입니다.

contentClassId

컨텐츠 클래스의 ID. 이는 오브젝트를 어디에 IMPORT하는 지와 이후에 이 오브젝트를 어떻게 관리하는 지를 나타냅니다. Xtorm 데이터베이스의 asysContentClass 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 컨텐츠 클래스 ID가 달라지므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 속성이 사용될 경우 archiveId 속성은 선택적입니다. 만약 두 가지 속성이 모두 세팅되는 경우 Xtorm은 contentClassId를 사용할 것입니다.

interval

IMPORT할 신규 배치를 체크하기 위하여 IMPORT 게이트웨이를 호출하기 전에, 더 이상의 유효한 데이터가 없는 경우 대기하는 시간(초)

권장 값: "20"

start

Xtorm 서버가 시작될 때 IMPORT 에이전트를 자동으로 시작할 지 여부를 지정.

| | |
|-----|-------------------------|
| Yes | Xtorm 서버 시작과 함께 자동으로 시작 |
| No | 자동으로 시작하지 않음 |

권장 값: "yes"

importdir

이 파라미터는 asysWindfireIGateway에만 적용합니다. 계속적으로 반복 배치 입력할 기본 경로를 지정합니다.

참고: 이 파라미터는 실제 운영 환경마다 경로가 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정 하십시오.

encoding

“index.dat” 파일을 읽을 때 사용할 인코딩 표준을 지정합니다. 특별히 이 속성을 지정하지 않거나, 공란으로 지정하면 IMPORT 게이트웨이는 일단 처음에는 유니코드로 처리를 시도하고, 유니코드가 아닌 것으로 판단되면 ANSI 표준으로 처리합니다. encoding="Unicode"로 지정하면 IMPORT 게이트웨이는 “index.dat” 파일을 유니코드 인코딩으로 읽으며, 기타 다른 값으로 지정하면 지정된 표준에 따라 파일을 읽어들이니다.

기본 값: “”

index

IMPORT를 진행하는 동안 데이터베이스 테이블에 색인 처리할 색인 정보 정의

indexid

색인 테이블의 인덱스 ID. Xtorm 데이터베이스의 asysOpenIndex 테이블을 참조하십시오.

delimiter

색인 데이터와 필드를 구분하는 문자

참고: 기본 값은 없습니다. 단, IMPORT 에이전트가 데이터 에이전트와 분리된 서비스에서 실행되는 경우, hyphen (-), underscore (_), colon (:), 또는 period (.) 중에 하나로 반드시 지정해야 합니다.

storproc

IMPORT 에이전트가 사용자 테이블에 엘리먼트를 배치로 색인할 때 저장 프로시저(Stored Procedure)를 사용할 수 있도록 하는 선택적인 파라미터입니다. 사용하려는 저장 프로시저 이름 및 프로시저에서 사용할 모든 파라미터를 함께 정의해야 합니다. 이 저장 프로시저는 각각의 고객 사이트의 특성에 맞게 수동으로 작성해야 하며 또한 데이터베이스에 직접 생성해야 합니다. Xtorm이 저장 프로시저를 자동으로 생성하지 않습니다. 이 속성에 아무 값도 지정되지 않은 경우, Xtorm은 다이나믹 SQL을 이용하여 사용자 테이블에 색인을 처리할 것입니다.

참고: 저장 프로시저를 사용하여 파일을 배치 처리하고자 하는 경우에만 이 옵션을 사용합니다.

field

색인 데이터 파일(index.dat) 안의 각각의 필드에 대한 정보. 여러 라인에 걸쳐 복수의 엔트리를 정의할 수 있습니다.

name

색인 테이블의 컬럼 이름

position

색인 데이터 파일 안에서 이 컬럼에 해당하는 데이터가 시작하는 위치

참고: 첫 컬럼의 위치는 0이 아닌 1 입니다.

length

데이터 필드의 길이 또는 최대 위치 값

default

색인 데이터 파일(index.dat)의 지정된 위치에 해당 필드 값이 존재하지 않거나, 날짜 필드와 같은 특별한 값들을 기본적으로 입력하고자 하는 경우에 사용합니다. 달리 말하면, 정해진 필드에 값이 없으면 이곳에서 지정한 값으로 입력됩니다. 또한, 색인 내용중에 없는 내용도 색인으로 처리할 수 있습니다.

“”

이 값을 지정하면 이 필드에 색인 값이 없는 경우 테이블에 빈 문자열을 입력할 것입니다. “char”, “nchar”, 또는 다른 유형의 모든 스트링 데이터 유형의 경우에 사용할 수 있습니다.

\$NULL\$

이 값을 지정하면 이 필드에 색인 값이 없는 경우 테이블에 NULL을 입력할 것입니다. numeric 및 char 데이터 유형에 적용할 수 있습니다. “\$NULL\$”은 반드시 대문자로 지정해야 합니다.

Any Value

필드에 어떠한 값도 지정할 수 있습니다. 단, 필드의 데이터 유형에 맞는 값으로 지정해야 합니다. 예를 들면, “Numeric” 필드에 색인 데이터가 없는 경우 0으로 입력하고 싶으면 0을 지정하십시오.

\$DATE\$yyyyMMdd

이 값을 지정하면 이 필드에 색인 값이 없는 경우 테이블에 현재의 날짜를 입력할 것입니다. “\$DATE\$”은 반드시 대문자로 지정해야 합니다.

참고: 이 속성을 지정하지 않으면 빈 문자열(empty string (“”))이 기본으로 사용됩니다.

기본 값: “”

document

색인 데이터 파일(index.dat) 안에서 실제적인 파일 이름을 나타내는 시작 위치

position

색인 데이터 파일 안에서 실제적인 파일 이름을 나타내는 시작 위치

참고: 첫 컬럼의 위치는 0이 아닌 1 입니다.

description

색인 데이터 파일 안에서 엘리먼트의 설명으로 사용할 필드 정보 정의

참고: description 속성은 여러 필드를 결합하여 사용할 수 있습니다.

field

필드 정보. 여러 라인에 걸쳐 복수의 엔트리를 정의할 수 있습니다.

position

색인 데이터 파일 안에서 엘리먼트의 설명으로 사용할 필드의 시작 위치

참고: 첫 컬럼의 위치는 0이 아닌 1 입니다.

length

데이터 필드의 길이 또는 최대 위치 값

Native Container 게이트웨이(“asysNativeGContainer”)

importgateway

IMPORT 게이트웨이와 관련된 정보 정의

name

IMPORT 게이트웨이 이름

권장 값: “IMPORTGATEWAY”

class

IMPORT 게이트웨이에 의해 수행될 클래스 이름

권장 값: “com.windfire.agents.importagent.gateway.asysNativeGContainer”

basedir

IMPORT할 데이터가 있는 기본 디렉토리 경로

참고: 이 파라미터는 실제 운영 환경마다 위치가 달라지므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

indexid

IMPORT를 진행하는 동안 데이터베이스 테이블에 색인 처리할 인덱스의 ID를 정의. Xtorm 데이터베이스의 asysOpenIndex 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 인덱스가 달라지므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 파라미터는 “index.dat” 파일의 엘리먼트 정의 부분에 INDEXID가 지정되지 않은 경우에 사용

됩니다.

eClassId

IMPORT할 엘리먼트의 유형 ID. Xtorm 데이터베이스의 asysEClass 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 파라미터는 "index.dat" 파일의 엘리먼트 정의 부분에 ECLASSID가 지정되지 않은 경우에 사용됩니다.

userSecureClassId

엘리먼트에 지정할 사용자 보안 클래스 ID. Xtorm 데이터베이스의 asysSecureClass 테이블을 참조하십시오. 보안 설정을 하지 않을 경우 "NONE"으로 세팅 하십시오. 이 항목은 선택적입니다.

참고: 이 파라미터는 실제 운영 환경마다 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 파라미터는 "index.dat" 파일의 엘리먼트 정의 부분에 SCLASSID가 지정되지 않은 경우에 사용됩니다.

contentClassId

컨텐츠 클래스의 ID. 이는 오브젝트를 어디에 IMPORT하는지와 이후에 이 오브젝트를 어떻게 관리하는지를 나타냅니다. Xtorm 데이터베이스의 asysContentClass 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 파라미터는 "index.dat" 파일의 엘리먼트 정의 부분에 CCLASSID가 지정되지 않은 경우에 사용됩니다.

참고: 이 속성이 사용될 경우 archiveId 속성은 선택적입니다. 만약 두 가지 속성이 모두 세팅되는 경우 Xtorm은 contentClassId를 사용할 것입니다.

archiveId

IMPORT를 진행할 아카이브 에이전트의 ID. 이는 엘리먼트를 어느 스토리지에 생성할 지를 결정합니다. Xtorm 데이터베이스의 asysArchive 테이블을 참조하십시오.

참고: 이 파라미터는 실제 운영 환경마다 다르므로 권장 값이 따로 없습니다. 실 환경에 맞게 설정하십시오.

참고: 이 파라미터는 "index.dat" 파일의 엘리먼트 정의 부분에 ARCHIVEID가 지정되지 않은 경우에 사용됩니다.

참고: 이 속성을 세팅하는 경우 contentClassId 속성은 선택적으로 사용될 수 있습니다. 만약, 두 가지 속성을 모두 지정하는 경우 Xtorm은 contentClassId를 이용할 것입니다. 만약 둘 모두 지정하지 않으면 생성되는 엘리먼트는 컨테이너가 될 것입니다. 실제 물리적인 파일을 입력 처리하는 경우 "archiveId", "contentClassId" 중 하나는 반드시 지정되어야 합니다.

interval

IMPORT할 신규 배치를 체크하기 위하여 IMPORT 게이트웨이를 호출하기 전에, 더 이상의 유효한 데이터가 없는 경우 대기하는 시간(초)

권장 값: "20"

start

Xtorm 서버가 시작될 때 IMPORT 에이전트를 자동으로 시작할 지 여부를 지정.

Yes Xtorm 서버 시작과 함께 자동으로 시작
No 자동으로 시작하지 않음

권장 값: "yes"

pvd delimiter

색인 데이터 파일(index.dat) 안에서 필드 이름과 필드 값을 구분하는 일에 사용하는 구분자를 정의

참고: 권장 값은 없습니다.

fdelimiter

색인 데이터 파일(index.dat) 안에서 엘리먼트의 필드들을 구분하는 일에 사용하는 구분자를 정의

참고: 권장 값은 없습니다.

tfdelimiter

데이터 에이전트 레벨에서 색인 데이터와 필드를 구분하는 문자

참고: 기본 값은 없습니다. 단, IMPORT 에이전트가 데이터 에이전트와 분리된 서비스에서 실행되는 경우, hyphen (-), underscore (_), colon (:), 또는 period (.) 중에 하나로 반드시 지정해야 합니다.

encoding

"index.dat" 파일을 읽을 때 사용할 인코딩 표준을 지정합니다. 특별히 이 속성을 지정하지 않거나, 공란으로 지정하면 IMPORT 게이트웨이는 일단 처음에는 유니코드로 처리를 시도하고, 유니코드가 아닌 것으로 판단되면 ANSI 표준으로 처리합니다. encoding="Unicode"로 지정하면 IMPORT 게이트웨이는 "index.dat" 파일을 유니코드 인코딩으로 읽으며, 기타 다른 값으로 지정하면 지정된 표준에 따라 파일을 읽어들이는 것입니다.

기본 값: ""

storproc

IMPORT 에이전트가 사용자 테이블에 엘리먼트를 배치로 색인할 때 저장 프로시저(Stored Procedure)를 사용할 수 있도록 하는 선택적인 파라미터입니다. 사용하려는 저장 프로시저 이름 및 프로시저에서 사용할 모든 파라미터를 함께 정의해야 합니다. 이 저장 프로시저는 각각의 고객 사이트의 특성에 맞게 수동으로 작성해야 하며 또한 데이터베이스에 직접 생성해야 합니다. Xtorm이 저장 프로시저를 자동으로 생성하지 않습니다. 이 속성에 아무 값도 지정되지 않은 경우, Xtorm은 다이나믹 SQL을 이용하여 사용자 테이블에 색인을 처리할 것입니다.

참고: 저장 프로시저를 사용하여 파일을 배치 처리하고자 하는 경우에만 이 옵션을 사용합니다.

다음의 속성들은 모든 구성에 공통적인 내용입니다.

retry

메인 데이터 서버(데이터 에이전트)와의 통신 오류 등으로 인하여 배치 처리가 실패하는 경우의 에러 메시지를 정의합니다. 이들 메시지는 단지 로그 분석을 위한 목적입니다. 달리 말하면, 이들 메시지들은 Xtorm 데이터 서버에 에러가 발생한 경우 로그 파일에 에러 로그를 남길 수 있게 하기 위해서입니다.

code

에러 메시지의 번호와 설명을 정의합니다. 각각의 에러에 대하여 한 라인씩 정의합니다.

reason

에러 메시지 번호

description

해당 에러에 대한 간단한 텍스트 메시지

예:

Native IMPORT 게이트웨이 (“asysNativeGateway”)

```
<agent name="IMPORT" type="IMPORT" desc="Import Data"
class="com.windfire.agents.importagent.asysAgentImport" opclass="/serMet/asysservlets.asysOperImport"
msgfile="msg/import.mc" minhealth="1">
```

```
  <startup bundleSize="100" threads="20" highWaterBundle="20" lowWaterBundle="10"
highWaterElement="1000" lowWaterElement="800" delete="yes" postProcess="yes" agentData="DATA"
target="Xtorm_MAIN" agentDataServer="HDUAL:2102" encoding="Unicode" retryInMins="5" />
```

```
  <recovery filename="recovery.xml" basedir="c:/Xtorm/log"/>
  <statistic filename="statistic.xml" basedir="c:/Xtorm/log" ageInDays="5"/>
  <error filename="error.xml" basedir="c:/Xtorm/log" ageInDays="5"/>
```

```
  <importgateway name="IMPORTGATEWAY"
class="com.windfire.agents.importagent.gateway.asysNativeGateway" basedir="c:/ImportData"
```

```

eClassId="IMAGE" userSecureClassId="PUBLIC" archiveId="MAIN" interval="20" start="yes" encoding="">

    <index indexid="SUPYO" delimiter="_" storproc="sp_usrAddChecks(EID,
    KYOHWAN_DATE, INDEX_NO, SUPYO_NO, JI_BANKCODE, JI_JIJUMCODE, KWONCODE,
    TOT_AMOUNT" >
        <field name="KYOHWAN_DATE" position="1" length="8" default=""/>
        <field name="INDEX_NO" position="10" length="12" default=""/>
        <field name="SUPYO_NO" position="23" length="8" default=""/>
        <field name="JI_BANKCODE" position="32" length="2" default=""/>
        <field name="JI_JIJUMCODE" position="35" length="4" default=""/>
        <field name="KWONCODE" position="40" length="2" default=""/>
        <field name="TOT_AMOUNT" position="43" length="10" default=""/>
    </index>

    <document position="54"/>

    <description>
        <field position="10" length="12"/>
        <field position="23" length="8"/>
    </description>
</importgateway>

<retry>
    <code reason="403" description="No database connection"/>
    <code reason="303" description="Service down"/>
</retry>

</agent>

```

Native IMPORT 게이트웨이 (“asysWindfirelGateway”)

```

<agent name="IMPORT" type="IMPORT" desc="Import Data"
class="com.windfire.agents.importagent.asysAgentImport" opclass="/serMet/asysservlets.asysOperlImport"
msgfile="msg/import.mc" minhealth="1">

    <startup bundleSize="100" threads="20" highWaterBundle="20" lowWaterBundle="10"
    highWaterElement="1000" lowWaterElement="800" delete="yes" postProcess="yes" agentData="DATA"
    target="Xtorm_MAIN" agentDataServer="HDUAL:2102" encoding="Unicode" retryInMins="5" />

    <recovery filename="recovery.xml" basedir="c:/Xtorm/log"/>
    <statistic filename="statistic.xml" basedir="c:/Xtorm/log" ageInDays="5"/>
    <error filename="error.xml" basedir="c:/Xtorm/log" ageInDays="5"/>

    <importgateway name="IMPORTGATEWAY"
class="com.windfire.agents.importagent.gateway.asysWindfirelGateway" basedir="c:/ImportData"
eClassId="IMAGE" userSecureClassId="PUBLIC" archiveId="MAIN" interval="20" start="yes"
importdir="hpi_1" encoding="">

        <index indexid="SUPYO" delimiter="_" storproc="sp_usrAddChecks(EID,
        KYOHWAN_DATE, INDEX_NO, SUPYO_NO, JI_BANKCODE, JI_JIJUMCODE,
        KWONCODE, TOT_AMOUNT" >
            <field name="KYOHWAN_DATE" position="1" length="8" default=""/>
            <field name="INDEX_NO" position="10" length="12" default=""/>

```

```

        <field name="SUPYO_NO" position="23" length="8" default=""/>
        <field name="JI_BANKCODE" position="32" length="2" default=""/>
        <field name="JI_JIJUMCODE" position="35" length="4" default=""/>
        <field name="KWONCODE" position="40" length="2" default=""/>
        <field name="TOT_AMOUNT" position="43" length="10" default=""/>
    </index>

    <document position="54"/>

    <description>
        <field position="10" length="12"/>
        <field position="23" length="8"/>
    </description>

</importgateway>

<retry>
    <code reason="403" description="No database connection"/>
    <code reason="303" description="Service down"/>
</retry>

</agent>

```

Native Container 게이트웨이 (“asysNativeIGContainer”)

```

<agent name="IMPORT" type="IMPORT" desc="Import Data"
class="com.windfire.agents.importagent.asysAgentImport" opclass="/servlet/asyservlets.asysOperImport"
msgfile="msg/import.mc" minhealth="1">

    <startup bundleSize="100" threads="20" highWaterBundle="20" lowWaterBundle="10"
highWaterElement="1000" lowWaterElement="800" delete="yes" postProcess="yes" agentData="DATA"
target="Xtorm_MAIN" agentDataServer="HDUAL:2102" encoding="Unicode" retryInMins="5" />
    <recovery filename="recovery.xml" basedir="c:/Xtorm/log"/>

    <statistic filename="statistic.xml" basedir="c:/Xtorm/log" ageInDays="5"/>

    <error filename="error.xml" basedir="c:/Xtorm/log" ageInDays="5"/>

    <importgateway name="IMPORTGATEWAY"
class="com.windfire.agents.importagent.gateway.asysNativeIGContainer" basedir="c:/ImportData"
indexid="Checks" eClassId="IMAGE" userSecureClassId="PUBLIC" archiveId="MAIN" interval="20"
start="no" pvdDelimiter=":" fdelimeter="," tfdelimeter="_" encoding="" storproc="sp_usrAddChecks(EID,
KYOHWAN_DATE, INDEX_NO, SUPYO_NO, JI_BANKCODE, JI_JIJUMCODE, KWONCODE,
TOT_AMOUNT" >
    </importgateway>

    <retry>
        <code reason="403" description="No database connection"/>
        <code reason="303" description="Service down"/>
    </retry>

</agent>

```

정보/데이터에이전트(유형: DATA)

속성

dbpools

데이터베이스 연결 풀과 관련된 정보 정의. 단일 데이터 에이전트에 의해 무한대의 데이터베이스 풀이 사용될 수 있습니다. Xtorm은 마스터 데이터베이스에 연결하기 위하여 적어도 하나의 풀을 필요로 합니다.

pool

데이터베이스 풀 및 연결 정보 정의

참고: 이 부분에서 보여지고 있는 기본값은 Xtorm 기본 풀에 대해서만 유효합니다. 또한, 이 기본값은 Xtorm 설치 시 설정되며 Xtorm 기본 구성 요소입니다. 여기에선 "Xtorm Pool 기본 값"으로 표현됩니다.

pool.name

풀의 논리적인 이름. 이 이름은 특정 데이터베이스 카백션과 아답터, 보안 관리자 또는 이벤트 관리자를 연결하기 위해 사용됩니다.

기본 값: "master"

pool.driver

JDBC 드라이버 클래스 이름

1. SQL Server(NetDirect Driver): "com.jnetdirect.jsql.JSQLDriver"
2. Oracle: "oracle.jdbc.driver.OracleDriver"
3. DB2: "com.ibm.db2.jcc.DB2Driver"
4. Sybase: "com.sybase.jdbc2.jdbc.SybDriver"
5. MySQL: "com.mysql.jdbc.Driver"
6. Altibase: "Altibase.jdbc.driver.AltibaseDriver"
7. Informix: "com.informix.jdbc.IfxDriver"
8. Tiberio: "com.tmax.tibero.jdbc.TbDriver"
9. Postgresql: "org.postgresql.Driver"

pool.count

동시에 사용될 최대 연결 수. 이는 특정 목적에 의해 데이터베이스 동시 접속 수를 제어할 필요가 있는 경우에 사용될 수 있습니다. 이 값은 데이터베이스가 어떻게 구성되어 있느냐에 따라 달라질 수 있음.

기본 값: "1"

pool.tableprefix

Xtorm이 사용하는 DB 인스턴스가 다 계층적으로 구성되어 여러 계층 하위에 존재할 경우 Xtorm 스키마에 접근하기 위한 계층 경로를 지정 합니다. 기본값은 "" 이며 계층 구분자는 DBMS 종류별로 달라 질 수 있음.

예: "edms.Xtorm." (Oracle)

pool.connect

데이터베이스 연결 문자열. 이 문자열은 번다별 JDBC 드라이버에 따라 달라집니다.
 SQL Server(NetDirect Driver): "jdbc:JSQLConnect://localhost/database=xxx"
 xxx는 Xtorm 설치 시에 입력한 데이터베이스 이름입니다.

Oracle: "jdbc:oracle:thin:@machine:port:instance"
 machine은 Oracle이 설치되어 있는 컴퓨터 이름
 port는 Oracle이 사용하는 포트 번호. 기본 1521
 instance는 접속할 데이터베이스 인스턴스 이름 입니다.

DB2: "jdbc:db2:// machine: port/xxx"
 machine은 DB2가 설치되어 있는 컴퓨터 이름 또는 IP
 port는 DB2가 사용하는 포트 번호. 기본 50000
 xxx는 접속할 데이터베이스 인스턴스 이름 입니다.

Sybase: "jdbc:sybase:Tds:machine:port/xxx"
 machine은 Sybase가 설치되어 있는 컴퓨터 이름 또는 IP
 port는 Sybase가 사용하는 포트 번호. 기본 4100
 xxx는 접속할 데이터베이스 인스턴스 이름 입니다.

Mysql : "jdbc:mysql://server:port/instance"
 Server는 Mysql 이 설치되어 있는 컴퓨터 이름 또는 IP 주소
 port는 Mysql 이 사용하는 포트 번호. 기본 3306
 instance는 접속할 데이터베이스 인스턴스 이름.

Altibase : "jdbc:Altibase://server:port/instance"
 Server는 Altibase가 설치되어 있는 컴퓨터 이름 또는 IP 주소
 port는 Altibase가 사용하는 포트 번호. 기본 20300
 instance는 접속할 데이터베이스 인스턴스 이름.

Informix: "jdbc:informix-sqli://server:port/instance:INFORMIXSERVER=IMX;user=id;password=passwd"
 Server는 informix가 설치되어 있는 컴퓨터 이름 또는 IP 주소
 Port는 informix가 사용하는 포트번호. 기본 1526
 Instance는 접속할 데이터베이스 인스턴스 이름.

Tibero: " jdbc:tibero:thin:@machine:port:instance"
 machine은 Tibero가 설치되어 있는 컴퓨터 이름
 port는 Tibero가 사용하는 포트 번호. 기본 8629
 instance는 접속할 데이터베이스 인스턴스 이름 입니다.

postgresql : "jdbc:postgresql://localhost:port/instance"
 localhost은 postgresql가 설치되어 있는 컴퓨터 hostname 또는 아이피
 port는 postgresql가 사용하는 포트 번호. 기본 5432
 instance는 접속할 데이터베이스 인스턴스 이름 입니다.
 예) "jdbc:postgresql://localhost:5432/XEDM"

pool.user

데이터베이스에 연결하기 위한 데이터베이스 사용자 ID

기본 값: Xtorm 설치 시에 입력한 특정 데이터베이스 사용자 이름입니다.

pool.pswd

데이터베이스에 연결하기 위한 데이터베이스 사용자 비밀번호

기본 값: Xtorm 설치 시에 입력한 특정 데이터베이스 사용자 비밀번호입니다.

pool.preconnect

Xtorm이 시작될 때 최대 수 만큼 미리 확보할 것인지를 결정합니다.

True Xtorm이 시작될 때 최대 수 만큼 미리 확보합니다.

false 필요한 경우에 따라 최대 연결을 할 것입니다. 대부분의 경우 이 값을 사용합니다.

기본 값: "false"

pool.unicode

특정 게이트웨이를 위하여 만약 몇몇 데이터베이스 테이블들이 유니코드 컬럼을 포함하고 있는 경우에 사용될 수 있습니다.

true 유니코드 테이블 방식으로 표준 테이블에 접근합니다.

false 유니코드 테이블 방식을 사용하지 않습니다.

Xtorm Pool 기본 값: Xtorm 설치 시 지정할 수 있습니다.

참고: 이 부분에서 보여지고 있는 기본값은 Xtorm 기본 풀에 대해서만 유효합니다. 또한, 이 기본값은 Xtorm 설치 시 세팅되며 Xtorm 기본 구성 요소입니다. 여기에선 "Xtorm Pool 기본 값"으로 표현됩니다. Oracle의 경우, 데이터베이스가 UTF8 또는 유니코드 데이터가 저장되어 있다 하더라도 Unicode 파라미터 속성은 항상 FALSE여야 합니다. 이 속성은 Microsoft SQL Server를 위한 Unicode 용 SQL 문장을 만드는 일에 사용되며 Oracle은 Character Set에 의해 유니코드가 지원됩니다.

pool.dbtype

데이터베이스 종류를 정의

- 1 Microsoft SQL Server
- 2 Oracle
- 3 DB2
- 4 Sybase
- 5 Mysql
- 6 Altibase
- 7 Informix
- 8 Tiberio
- 9 postgresql

pool.maxtrans

커넥션이 열려있는 동안 실행될 수 있는 트랜잭션 개수의 최대값

Xtorm의 데이터베이스 커넥션은 재요청될 경우 자동으로 재개 됩니다.

이 속성의 목적은 ORACLE의 최대 오픈 커서 개수 제한에 따라 야기될 수 있는 문제에 대해 대비하기 위함입니다

파라미터 설정은 선택사항이며 반드시 지정될 필요는 없습니다.

지정되지 않을 경우, DB 연결 커넥션은 무제한으로 허용됩니다.

주의: 이 파라미터는 커넥션 오버헤드로 인해 시스템 성능에 심각한 영향을 끼칠 수 있습니다.

pool.tableprefix

Xtorm이 사용하는 DB 인스턴스가 다 계층적으로 구성되어 여러 계층 하위에 존재할 경우 Xtorm 스키마에 접근하기 위한 계층 경로를 지정 합니다. 기본값은 "" 이며 계층 구분자는 DBMS 종류별로 달라 질 수 있습니다.

예: "edms.Xtorm." (Oracle)

pool.connectionTimeout

데이터베이스 접속에 대한 타임아웃시간 설정(초)

pool.connectionProperties

데이터베이스 접속에 대한 접속 프로퍼티 속성

connectionProperties="oracle.net.CONNECT_TIMEOUT=10000;oracle.jdbc.ReadTimeout=10000"

pool.encryptclass

DB 패스워드 암호화 사용 가능 암호화된 패스워드를 복호화할 클래스명 지정

1. 암호화는 별도로 프로그램 통해서 암호화 후 pwsd 값 항목에 입력 함.
2. 복호화하는 모듈로 복호화 하는 클래스 직접 작성 하여 엔진 시작 스크립트 CLASS파일 경로를 CLASSPATH 잡아줘야함.

상세 내용은 DB 패스워드 암호화 가이드 참고

listspooler.timeout

검색한 결과 리스트를 얼마 동안 Xtorm 서버에 남겨둘지를 결정하는 값, 특별히 값을 지정하지 않으면 기본 값 10 분이 적용됨. timeout 값은 초 단위로 설정하며, Xtorm 서버가 주기적으로 결과 리스트를 닫음으로써, 어플리케이션에서 검색을 수행한 후 명시적으로 결과 리스트를 닫는 것을 실패했다 하더라도 메모리를 계속 사용하는 것을 예방함.

기본 값: "300"

physicalvolumemanager

물리적 볼륨 정보를 자동으로 관리하기 위해 전용 쓰레드를 구동 시키도록 설정, 기존 볼륨 처리 방식은 볼륨 정보의 장합성을 유지하기 위해 불가피하게 DB 처리의 병목 현상이 발생할 수 밖에 없었지만 자동 관리 방식은 물리적인 볼륨크기를 주기적으로 엔진에 반영하는 해당 쓰레드를 활성화시켜 매 트랜잭션마다 볼륨 정보를 갱신함에 따른 DB 병목 현상을 해결할 수 있음.

startup

Xtorm 서버 시작 시 마스터 데이터베이스에 접속할 데이터베이스 풀 정의. 이는 데이터 에이전트로 하여금 마스터 데이터베이스에 저장되어 있는 시스템 정보에 어떻게 접근할 수 있는지를 정의하며 따라서 해당 게이트웨이, 권한 관리자 및 포스트 프로세서 등이 제대로 로드되고 사용되게 합니다. 하나의 논리적인 서버 내에 있는 모든 데이터 에이전트는 동일 마스터 데이터베이스에 접근하도록 정의되어야 합니다.

startup.mainpool

Xtorm 서버 시작 시 마스터 데이터베이스에 접속할 풀 이름 정의

기본 값: "master"

startup.localtime

Xtorm 서버와 DBMS 서버가 물리적으로 분리되어 설치될 경우 타임 스탬프(Timestamp) 값을 DBMS 서버와 동기화 하기 위해 현재 시간에 대한 지역 편차값을 설정합니다. Oracle, Tiberio 등에만 유효합니다.

예: "GMT+09:00"

기본 값: "GMT"

startup.checksum

파일 중복 저장 방지 기능의 활성화 여부를 지정합니다. 해당 기능은 운영 도중 변경할 경우 파일이 유실될 수 있으므로 초기 활성화 이후, 비활성화로 설정을 변경해서는 안됩니다.

기본 값: "NO" (비활성화)

startup.VOLUMEEXPANSION

VOLUMEID 컬럼사이즈 DB에서 자동으로 가져와서 세팅 하도록 되어 있음.

볼륨컬럼사이즈 변경에 따른 옵션 추가에 따른 명시가능 volumeid 컬럼사이즈가 VARCHAR2(64)일 경우 VOLUMEEXPANSION="TRUE"

startup.TransactionRetries

SimpleSQL 또는 내부 Query가 Timeout등의 오류가 발생하면, 무한 재시도를 하면서 A connection to the database has been lost. Closing connection and retrying transaction...

로그가 지속적으로 발생하는 문제 수정.(재시도 회수 지정)

startup.xedrmuserclass

XEDRM 유저 패스워드 XTORM과 호환

엔진 유저패스워드 방식을 XtormDB 타입을 사용 해야 가능함.

- conf.xml 설정

<startup mainpool="master" checksum="no" xedrmuserclass="xedrmCryptClass"/>

- conf.xml 설정 후 클래스패스에 해당 클래스(xedrmCryptClass) 추가 하여 사용

xedrmsample폴더 클래스(xedrmCryptClass) 배포함.

listspooler

스풀 리스트와 관련된 정보 정의

listspooler.timeout

검색한 결과 리스트를 얼마 동안 Xtorm 서버에 남겨둘지를 결정하는 값 입니다. 특별히 값을 지정하지 않으면 기본 값 10 분이 적용됩니다. timeout 값은 초 단위로 설정하며, Xtorm 서버가 주기적으로 결과 리스트를 닫음으로써, 어플리케이션에서 검색을 수행한 후 명시적으로 결과 리스트를 닫는 것을 실패했다 하더라도 메모리를 계속 사용하는 것을 막아 줄 것입니다.

기본 값: "300"

listspooler.listdir

대용량 결과 리스트를 캐싱하고 스푼링하는 데에 사용될 기본 디렉토리 정의

기본 값: "spool"

listspooler.memcache

스푼링하는 데에 메모리에서 처리 될 건수 지정 가능

기본 값: "250"

physicalvolumemanager

물리적 볼륨 정보를 자동으로 관리하기 위해 전용 쓰레드를 구동 시키도록 설정 합니다. 기존 볼륨 처리 방식은 볼륨 정보의 정합성을 유지하기 위해 불가피하게 DB 처리의 병목 현상이 발생할 수 밖에 없었지 만 자동 관리 방식은 물리적인 볼륨크기를 주기적으로 엔진에 반영하는 해당 쓰레드를 활성화시켜 매 트랜잭션 마다 볼륨 정보를 갱신함에 따른 DB 병목 현상을 해결할 수 있습니다.

physicalvolumemanager.iteration

볼륨 정보 관리 쓰레드의 구동 주기를 지정

단위: 초

physicalvolumemanager.OnErrorReadOnly

물리볼륨 정보 취득 실패 시 해당 볼륨을 '읽기전용'으로 자동 변경할지의 여부 설정. 읽기전용으로 변경된 이후 물리볼륨관리 모듈이 주기적으로 볼륨정보 추출을 시도하여 정상적으로 정보가 재취득될 경우 아카이브 에이전트의 'Autowritable' 옵션이 'True'이면 다시 'writable'로 변경되어 해당 볼륨을 사용하게 됨.

'YES': 정보 취득 실패 시 '읽기전용'으로 해당볼륨을 설정하고 다음 볼륨 사용,
'NO': 정보 취득 실패에도 불구하고 에러로그를 반환한 뒤, 해당 볼륨 재사용 시도

physicalvolumemanager.USABLESPACECHECK

물리볼륨 설정시 UsableSpace 값 초과 설정시 UsableSpace값을 Maximum Space값으로 변경 하는 부분 옵션값으로 선택할수 있음. (default는 변경안함)

USABLESPACECHECK 값 TRUE 설정하면 UsableSpace값이 Maximum Space값보다 크게 설정되어 있는 경우 UsableSpace 값을 Maximum Space 값으로 변경함.

기본값: "FALSE"

volumeinspector

메모리릭 문제로 사용 하면 안되는 옵션 임.

대량의 트랜잭션 처리 시 볼륨 정보 업데이트와 관련하여 DB 트랜잭션의 오버헤드를 줄이기 위해 일괄적으로 볼륨 정보를 업데이트 할 수 있는 기능을 활성화 시킵니다. 구동 방식은 업데이트가 발생한 볼륨리스트를 유지하다가 주기적으로 볼륨에 대한 일괄 업데이트를 수행합니다. 또한, Xtorm 서버의 정상 종료 시에도 볼륨 변경이 있을 경우 일괄 업데이트를 수행합니다.

volumeinspector.iteration

해당 모듈의 구동 주기를 지정 합니다. 기본적으로 활성화되지 않습니다.

(단위: 초)

volumeinspector.rstime

볼륨 크기에 가감이 발생하면 각각의 볼륨별로 구분되는 변경 리스트를 유지하는데, volume inspector 모듈이 동작하는 시점에 이 리스트에 저장되어 있는 볼륨 크기의 변경 시점을 기준으로 해당 볼륨 정보를 DB에 갱신할지의 여부를 결정짓게 됩니다.

이때, 기준이 되는 시점을 '볼륨 크기 변경 발생 시점'에 rstime ('리스트 잔존 시점')을 더하여 volume inspector 모듈의 동작 시점 보다 크면 DB에 갱신할 대상으로 판단합니다. 도식화 하여 정리하면 다음과 같습니다.

DB 업데이트 대상 선정 방법:

'volume inspector 동작 시점' > '볼륨별 크기 변경 시점' + 'rstime

(단위: 초)

volumeinspector.batchupdate

매 트랜잭션 단위로 DB 업데이트를 수행하지 않고 일단 리스트에 크기 변경이 발생한 블록 정보를 기록해 두었다가 일괄적으로 DB 업데이트를 수행하도록 지정 합니다.

기본값: "NO"

volumeinspector.debug

해당 모듈의 디버깅 정보를 디버거 ID를 지정하여 별도 파일로 저장하도록 지정합니다.

activemanager

A-A 구성시 블록정보 및 엔진 설정 정보 자동 동기화 기능

activemanager.iteration

iteration 주기로 동기화 함 단위(초). 권장값 600(10분)
값: 600(초)

checkquery

DB 카백션 폴 체크기능

checkquery.CHECKQUERYENABLE

DB 카백션 폴 체크기능 활성화 여부

값: "true"

checkquery.CHECKQUERYTIMEOUT

DB 카백션 폴 체크쿼리 타임아웃 시간 설정(초)
값: "10"

checkquery.VALIDATIONINTERVAL

DB 카백션 폴 카백션폴 체크주기(초)
최소 300 부터 가능함.
값: "300"

farm

High-Availability 기능을 지원하기 위한 서버 파밍(farming) 정보 정의

farm.enabled

High-Availability 기능의 활성화 여부를 지정 합니다.

기본 값: "false"

farm.ucode

서버 파밍(farming)을 구성하는 해당 노드의 유일키(Unique Key) 값을 지정 합니다. 이 값은 각 노드에서 생성되는 파일의 ElementId 값에 반영되며 기본적으로 Xtorm은 파일명으로 ElementId 값을 사용하기 때문에 저장공간을 공유하고 있는 각 노드별로 파일명이 중복되지 않도록 하기 위해 유일키 값을 지정 합니다.

예: "1"

farm.farmeid

A-A 구성시 farm enable false 일 경우에도 elementid 중복발생 없이 2로 시작하는게 기능 추가

- 1) farm 부분에 farmeid="true" 추가.
 - 2) ucode에 따라서 elementid 채번 규칙 정해짐. 각 서버에서 ucode는 다르게 설정되어야 함.
- 예: "true"
기본값 "false"

farm.dserver

서버 파밍(farming)을 구성하는 다른 노드의 구성 정보를 정의 합니다.

farm.dserver.address

서버 파밍(farming)을 구성하는 다른 노드의 IP 정보를 지정 합니다.

예: "10.1.1.2"

farm.dserver.port

서버 파밍(farming)을 구성하는 다른 노드의 서비스 포트 정보를 지정 합니다.

예: "2102"

다음의 속성들은 Audit 에이전트를 위한 속성들입니다.

Audit 에이전트에 의해 데이터 에이전트의 트랜잭션을 캡처하려면 데이터 에이전트 구성 부분에 이를 정의해야 합니다.

defaultaudit

Audit 에이전트에 의해 데이터 에이전트의 트랜잭션을 캡처하도록 이 부분에서 파라미터를 정의합니다. 특별히 파라미터를 정의하지 않거나, agentname 및 agentserver 파라미터가 공란으로 지정되면 데이터 에이전트는 Audit 에이전트에 트랜잭션을 보내지 않습니다.

defaultaudit.agentname

사용할 AUDIT 유형의 에이전트 이름을 지정합니다. 캡처되는 트랜잭션을 기록할 에이전트를 의미합니다.

기본 값: "AUDIT"

defaultaudit.agentserver

Audit 에이전트가 설치되어 있는 Xtorm 서버의 이름을 MACHINE:PORT 형식으로 지정합니다. MACHINE은 Audit 에이전트가 설치되어 있는 컴퓨터의 IP 주소 또는 호스트 이름입니다. PORT는 해당 컴퓨터에서 Xtorm에 의해 사용되는 리스닝 포트 번호입니다. 포트 파라미터에 아무 값도 지정되지 않으면 기본 값인 2102가 사용됩니다. agentname 파라미터에는 값이 지정되어 있고 agentserver 파라미터에는 아무 값도 지정되어 있지 않으면 Audit 에이전트의 위치는 localhost:2102 이라고 간주됩니다.

예: Microsoft SQL Server

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
```

```

msgfile="msg/data.mc" minhealth="1">
  <dbpools>
    <pool name="master" driver="com.jnetdirect.jsql.JSQLDriver"
    connect="jdbc:JSQLConnect://localhost/database=Xtorm" user="sa" pswd="" count="10"
    preconnect="false" unicode="true" dbtype="1" tableprefix="ecm.Xtorm."/>
  </dbpools>

  <startup mainpool="master" checksum="NO"/>
  <listspooler listdir="spool" timeout="600"/>
  <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />
  <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

  <farm enabled="false" ucode="1" >
    <dserver address="192.168.110.2" port="2102" />
  </farm>
</agent>

```

예: Oracle

```

<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile="../
../msg/data.mc" minhealth="1">
  <dbpools>
    <pool name="dbpool_oracle" driver="oracle.jdbc.driver.OracleDriver"
    connect="jdbc:oracle:thin:@10.90.1.35:1521:ORCLDABB" user="asysarm" pswd="asysarm" count="20"
    preconnect="FALSE" unicode="FALSE" dbtype="2" tableprefix="edms." />
    <pool name="dbpool_mssql" driver="com.jnetdirect.jsql.JSQLDriver"
    connect="jdbc:JSQLConnect://10.90.1.32/database=Xtorm" user="sa" pswd="sa" count="10"
    preconnect="FALSE" unicode="FALSE" dbtype="1"/>
  </dbpools>
  <startup mainpool="master" checksum="NO" localtime="GMT+09:00" TransactionRetries="100" />
  <checkquery CHECKQUERYENABLE="true" CHECKQUERYTIMEOUT="10"
  VALIDATIONINTERVAL="300" />
  <listspooler listdir="spool" timeout="600" memcahe="250"/>
  <physicalvolumemanager iteration="60" OnErrorReadOnly="no" USABLESPACECHECK="false"/>
  <activemanager iteration="600"/>
  <defaultaudit agentserver="dabyp:2102" agentname="AUDIT"/>
</agent>

```

예: DB2

```

<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
msgfile="msg/data.mc" minhealth="1">
  <dbpools>
    <pool name="master" driver="com.ibm.db2.jcc.DB2Driver"
    connect="jdbc:db2://scooby:50000/Xtorm:driverType=4;" user="Administrator" pswd="adminsRus2"
    count="10" dbtype="3" maxtrans="400" tableprefix="ecm.XT."/>
  </dbpools>

  <startup mainpool="master" checksum="NO"/>
  <listspooler listdir="spool" timeout="600"/>

```

```

<defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />

<physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

<farm enabled="false" ucode="1" >
    <dserver address="192.168.110.2" port="2102" />
</farm>
</agent>

```

예: Sybase

```

<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
msgfile="msg/data.mc" minhealth="1">
    <dbpools>
        <pool name="master" driver="com.sybase.jdbc2.jdbc.SybDriver"
connect="jdbc:sybase:Tds:daphne:2048/Xtorm" user="sa" pswd="" count="10" preconnect="false"
unicode="false" dbtype="4" maxtrans="400" tableprefix="ecm.XT." />
    </dbpools>

    <startup mainpool="master" checksum="NO"/>
    <listspooler listdir="spool" timeout="600"/>
    <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />

    <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

    <farm enabled="false" ucode="1" >
        <dserver address="192.168.110.2" port="2102" />
    </farm>
</agent>

```

예: MySQL

```

<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile="../
../msg/data.mc">
    <dbpools>
        <pool name="master" driver="com.mysql.jdbc.Driver"
connect="jdbc:mysql://localhost:3306/Xtorm" user="root" pswd="root" count="20"
preconnect="false" unicode="false" dbtype="5" maxtrans="400" tableprefix="ecm.XT." />
    </dbpools>

    <startup mainpool="master" checksum="NO"/>
    <listspooler listdir="spool" timeout="600"/>
    <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />

    <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

    <farm enabled="false" ucode="1" >
        <dserver address="192.168.110.2" port="2102" />
    </farm>
</agent>

```

예: Altibase

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile="../
../msg/data.mc">
  <dbpools>
    <pool name="master" driver="Altibase.jdbc.driver.AltibaseDriver"
    connect="jdbc:Altibase://localhost:20300/mydb" user="Xtorm" pswd="Xtorm" count="20"
    preconnect="false" unicode="false" dbtype="6" maxtrans="400" tableprefix="ecm.XT."/>
  </dbpools>

  <startup mainpool="master" checksum="NO"/>

  <listspooler listdir="spool" timeout="600"/>

  <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />
  <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

  <farm enabled="false" ucode="1" >
    <dserver address="192.168.110.2" port="2102" />
  </farm>
</agent>
```

예: Informix

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile="../
../msg/data.mc">

  <dbpools>
    <pool name="master" driver=" com.informix.jdbc.IfxDriver" connect=" jdbc:informix-
    sqli://HDUAL:1526/Xtorm:INFORMIXSERVER=IMX; user=informix;password=manager" user=""
    pswd="" count="20" unicode="false" dbtype="7" maxtrans="400" tableprefix="ecm.XT."/>
  </dbpools>

  <startup mainpool="master" checksum="NO"/>

  <listspooler listdir="spool" timeout="600"/>

  <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />
  <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>

  <farm enabled="false" ucode="1" >
    <dserver address="192.168.110.2" port="2102" />
  </farm>

</agent>
```


예: Tibero

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
msgfile="msg/data.mc" minhealth="1">
  <dbpools>
    <pool name="master" driver=" com.tmax.tibero.jdbc.TbDriver"
connect="jdbc:tibero:thin:@10.90.1.41:8629:tibero" user="Xtorm" pswd="Xtorm" count="10"
preconnect="false" unicode="false" dbtype="8" maxtrans="400" tableprefix="ecm.XT."/>
  </dbpools>

  <startup mainpool="master" checksum="NO" localtime="GMT+09:00"/>
  <listspooler listdir="spool" timeout="600"/>
  <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />
  <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>
  <farm enabled="false" ucode="1" >
    <dserver address="192.168.110.2" port="2102" />
  </farm>
</agent>
```

예: PostgreSQL

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
msgfile="msg/data.mc" minhealth="1">
  <pool name="master" driver="org.postgresql.Driver"
connect="jdbc:postgresql://localhost:5432/XEDM"
user="Xtorm" pswd="Xtorm" count="10" preconnect="FALSE"
unicode="FALSE" dbtype="9" />
  <startup mainpool="master" checksum="NO" localtime="GMT+09:00"/>
  <listspooler listdir="spool" timeout="600"/>
  <defaultaudit agentname="" agentserver="" />
  <physicalvolumemanager iteration="60" OnErrorReadOnly="yes"/>
  <farm enabled="false" ucode="0" >
    <dserver address="192.168.110.2" port="2102" />
  </farm>
</agent>
```

DB패스워드 암호화 가이드

속성

pool.encryptclass

암호화된 패스워드를 복호화할 클래스명 지정

1. 암호화는 별도로 프로그램 통해서 암호화 후 pswd 값 항목에 입력 함.
2. 복호화하는 모듈로 복호화 하는 클래스 직접 작성 하여 엔진 시작 스크립트 CLASS파일 경로를

CLASSPATH 잡아줘야함.

예:

```
<agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData" msgfile="../
../msg/data.mc" >
  <dbpools>
    <pool name="master" driver="oracle.jdbc.driver.OracleDriver"
    connect="jdbc:oracle:thin:@192.168.0.33:1521:ORCL" encryptclass="passwdDecrypt"
    user="Xtorm" pswd="13AA9692AE30D042535504630E5BEC71" count="20" preconnect="FALSE"
    unicode="FALSE" dbtype="2" />
  </dbpools>
  <startup mainpool="master" checksum="NO" locasysime="GMT+09:00" retentionignore="NO"/>
  <listspooler listdir="spool" memcache="250"/>
  <!--<physicalvolumemanager iteration="60" OnErrorReadOnly="no"/>-->
  <defaultaudit agentserver="" agentname="" />
  <farm enabled="false" ucode="1">
    <dserver address="192.168.1.3" port="3102" />
  </farm>
</agent>
```

DB패스워드 암호화 정의

DATA 에이전트가 DB패스워드에 대한 암호화 기능 구현

암호화: com.windfire.agents.data.asys.ext.asysExtEncryptPw

DB패스워드 복호화 구현체 작성

예제1 – DB 패스워드 복호화 수행

```
// Xtorm
import com.windfire.agents.data.asys.ext.asysExtEncryptPw;
import com.windfire.ws.util.wfCrypto;

// This search field validation module will check for wildcards
public class passwdDecrypt extends asysExtEncryptPw{
  // Constructor
  public passwdDecrypt(){
  }
  public String getDecryptPasswdKey(String decryptKey){
    try {

      // decryptKey = conf.xml 에서 입력한 암호화된 passwd 값 넘어옴.
      // 암호화 키= "spenocom1234"

      String decStr = "";
      wfCrypto wfC = new wfCrypto();

      //암호화된 passwd값 복호화
      decStr = wfC.decryptToString(decryptKey, "spenocom1234", new StringBuffer());

      decryptKey = decStr;
```

```

    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    // 복호화된 값 리턴
    return decryptKey;
}
}

```

마이그레이션 에이전트(유형: **MIGRATION**)

속성

logging

마이그레이션 에이전트에 대한 로깅 파라미터 정의

logging.basedir

마이그레이션 에이전트에 의해 사용될 로그 파일의 경로를 설정. 로그 파일을 XML 형식으로 작성됩니다.

권장 값: "/Xtorm/log"

logging.maxentries

로그 파일 안에 기록한 최대 엔트리 수. 하나의 엔트리 수는 에이전트에 의해 접수된 모든 정보를 포함합니다. 최대 엔트리에 이르면 로그 내용은 뒤에 추가될 것입니다.

권장 값: "10"

logging.dateFormat

로그 파일에서 사용할 날짜 형식 지정. 자바의 "SimpleDateFormat" 클래스에서 지원하는 모든 날짜 형식 가능

기본 값: "dateFormat="yyyy-MM-dd 'at' HH:mm:ss""

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다. "hh"를 사용할 경우, 오전(a.m.) 인지 오후(p.m.)인지를 명시해야 합니다. (예: dateFormat="yyyy-MM-dd 'at' hh:mm:ss a")

logging.encoding

로그 파일을 작성할 때 유니코드를 사용할 지 여부를 지정합니다. 지정하지 않으면 로그 XML 파일은 ISO-8859-1 형식으로 작성되며 지정할 경우 UTF-16 (Unicode) 형식으로 작성됩니다.

기본 값: ""

defaults

마이그레이션 에이전트의 쓰레드 수와 한번에 작업하는 배치의 크기를 정의.

이 부분을 이용하여 모든 콘텐츠 클래스에 공통적으로 적용하거나, 또는 콘텐츠 클래스마다 개별적으로

적용할 수도 있는데 콘텐츠 클래스마다 개별적으로 적용하는 경우 defaults 부분에 있는 기본 값들은 무시됨.
아무 파라미터도 적용하지 않으면 기본 값인 maxreqs="4", bundlesize="100"이 적용.

defaults.maxreqs

번들 단위로 아카이브 에이전트나 데이터 에이전트에 동시에 처리를 요청할 수 있는 최대 쓰레드 수를 의미.

defaults.bundlesize

한번에 처리하는 번들의 파일 개수. 해당 파라미터를 크게 지정한다고 해서 성능이 빨라지는 것은 항상 아님. I/O의 상황에 따라 적절하게 튜닝하여 지정해야 함.

defaults.aborttime

Operation Page에서 마이그레이션을 강제 종료할 경우, 진행중인 마이그레이션 작업을 종료할 때까지의 대기시간. 마이그레이션 작업이 지정된 시간동안 종료되지 않을 경우, 강제 종료 시킴. (단위: 초)
기본 값: 300초 (5분)

defaults.timediff

마이그레이션 에이전트가 구동되는 서버의 시간과 DB 서버의 시간차를 조정하기 위한 파라미터. (단위: 분)

기본 값: 0 분

예: 540분 (9시간)

defaults.maxignoreerrors

이판 대상 번들에 포함된 파일들을 저장소에서 읽기 시도 시, 에러가 발생할 경우 에러로 처리되어 구동이 중지되지 않고 다음 번들로 이어서 처리되도록 지정하는 기능입니다. 해당 파라미터에 지정된 값을 참조하여 읽기를 재시도할 다음 번들 개수를 결정하게 됩니다.

기본 값: 100

cclass

처리할 각각의 Content Class에 필요한 정보를 정의

cclass.agentDataServer

데이터 에이전트가 설치되어 있는 컴퓨터 이름. 포트 번호가 지정되어 있지 않으면 기본으로 2102를 사용할 것입니다.

cclass.target

"cclass" 속성에 정의되어 있는 콘텐츠 클래스를 관리하고 있는 Xtorm 게이트웨이 ID

cclass.cclass

마이그레이션을 진행하기 위하여 사용될 콘텐츠 클래스 ID

cclass.maxreqs

컨텐츠 클래스 단위로 지정되는 마이그레이션 에이전트, 아카이브 에이전트, 데이터 에이전트 간에 동시에 처리하는 최대 쓰레드 수를 의미합니다.

cclass.bundlesize

컨텐츠 클래스 단위로 지정되는 한번에 작업하는 파일 수

cclass.putfilters

컨텐츠 클래스 단위로 지정되는 입력용 필터

cclass.filterparms

컨텐츠 클래스 단위로 지정되는 입력용 필터의 파라미터

cclass.manual

대용량의 레코드를 저장하고 있는 상태에서 마이그레이션의 성능을 높이기 위해 관리자가 수동으로 지정한 범위값을 사용할지의 여부를 지정 합니다.

기본 값: "NO"

cclass.debug

컨텐츠 클래스 단위로 지정되는 디버깅 파라미터

cclass.maxlistsize

마이그레이션시 한번에 가져오는 대상건수 지정
기본값: 20000

cclass.migstat

마이그레이션 수행 정보를 DB 이력 남김.
기본값: false
주의 사항: 옵션 활성화시 DB 테이블 및 프로시저 추가해야함
테이블명: ASYSMIGSTAT
프로시저 명: sp_asysAddMigStat
값: true

defaults.maxignoreerrors

이관 대상 번들에 포함된 파일들을 저장소에서 읽기 시도 시, 에러가 발생할 경우 에러로 처리되어 구동이 중지되지 않고 다음 번들로 이어서 처리되도록 지정하는 기능.
해당 파라미터에 지정된 값을 참조하여 읽기를 재시도할 다음 번들 개수를 결정하게 됩니다.

기본 값: 100

예:

```
<agent name="MIGRATION" type="MIGRATION" desc="Migration Manager"
class="com.windfire.agents.migration.asysAgentMig" opclass="/serMet/asysservlets.asysOperMig"
```

```

msgfile="msg/migration.mc" minhealth="1">
  <logging basedir="log/mig" maxentries="10" dateformat="yyyy-MM-dd 'at' HH:mm:ss"
encoding="Unicode"/>
  <defaults maxreqs="30" bundlesize="50" aborttime="300" timediff="540"
maxignoreerrors="100"/>
  <cclass agentDataServer="HDUAL" target="Xtorm_MAIN" cclass="TEST" maxreqs="5"
bundlesize="50" putfilters="Encrypt1" filterparms="" manual="YES" MAXLISTSIZE="20000"
migstat="TRUE" />
  <cclass agentDataServer="HDUAL" target="Xtorm_MAIN" cclass="BASIC" maxreqs="5"
bundlesize="50" putfilters="Encrypt2" filterparms="" manual="YES" />
  ...
</agent>

```

마이그레이션 범위 수동 지정 및 자동 갱신

cclass.manual

“YES”로 설정. 수동으로 지정한 범위 값을 사용하고 마이그레이션 후 자동 갱신 여부를 지정

Range Update로 위치 지정 가능.

Active-Active 일 경우, Migration agent 에 설정된 data server에 해당하는 machineID 로 마이그레이션이 진행됨.

에러가 발생한 elementId 는 건너뛰고 Begin 이 설정되므로, 로그 확인이 필요함.

The screenshot displays the 'All Content Classes' window with a list of classes: BASIC --> BASIC and EXPIRECC --> EXPIRECC. Below the list are 'New', 'Edit', and 'Delete' buttons. To the right, 'Selected Class Details' shows attributes for the 'BASIC' class: ID (BASIC), Description (BASIC), User Security Class (NONE), and Admin Security Class (NONE).

Below this, the 'States' section features an 'Update' button and a table with columns: ID, Description, When, Condition, and Locations.

| ID | Description | When | Condition | Locations |
|-------------|-------------|------------------|--|-------------------------------|
| BASIC_INI → | BASIC_INI | Initial Required | | MAIN (Archive) |
| go1 → | go1 | Ongoing | ageover(30,days) | CAS (Archive) |
| go2 → | 111 | Ongoing | ageover(0, days) and ageunver(30,days) | CAS (Archive), MAIN (Archive) |

At the bottom, the 'Range' section has an 'Update' button and a table with columns: Begin, End.

| Begin | End |
|---------|---------|
| minimum | maximum |

스케줄러에이전트 (유형: SCHEDULER)

속성

info

Xtorm 운영자 어플리케이션의 스케줄러 페이지 상에 나타나게 할 스케줄 태스크에 대한 일반 사항을 정의합니다.

info.view

Xtorm 운영자 어플리케이션의 스케줄러 페이지 상의 스케줄 태스크 리스트 및 트리거 태스크 리스트에 나타나게 할 최대 엔트리 수를 정의합니다.

기본 값: "20"

info.datetimeformat

Xtorm 운영자 어플리케이션의 스케줄러 페이지 상에 표시할 날짜 및 시간 표현 형식 지정. 자바의 "SimpleDateFormat" 클래스에서 지원 하는 모든 날짜 형식 가능

기본 값: "dateformat="yyyy-MM-dd 'at' HH:mm:ss""

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다. "hh"를 사용할 경우, 오전(a.m.) 인지 오후(p.m.)인지를 명시해야 합니다.

(예: dateformat="yyyy-MM-dd 'at' hh:mm:ss a")

info.dateformat

Xtorm 운영자 어플리케이션의 스케줄러 페이지 상에 표시할 날짜 표현 형식 지정. 자바의 "SimpleDateFormat" 클래스에서 지원 하는 모든 날짜 형식 가능

기본 값: "yyyy-MM-dd"

info.timeformat

Xtorm 운영자 어플리케이션의 스케줄러 페이지 상에 표시할 시간 표현 형식 지정. 자바의 "SimpleDateFormat" 클래스에서 지원 하는 모든 날짜 형식 가능

기본 값: "HH:mm:ss"

참고: 시간을 지정하는 경우(HH:mm:ss), "HH"는 24 시간 단위로 표현되며, "hh"는 12 시간 단위로 표현됩니다. "hh"를 사용할 경우, 오전(a.m.) 인지 오후(p.m.)인지를 명시해야 합니다.

(예: dateformat="yyyy-MM-dd 'at' hh:mm:ss a")

info.encoding

로그 파일을 작성할 때 유니코드를 사용할 지 여부를 지정합니다. 지정하지 않으면 로그 XML 파일은 ISO-8859-1 형식으로 작성되며 지정할 경우 UTF-16 (Unicode) 형식으로 작성됩니다.

기본 값: ""

tasks

저장될 태스크에 대한 정보를 정의

tasks.filename

모든 태스크 정보를 저장할 파일 이름 지정

기본 값: "tasks.xml"

tasks.basedir

태스크 정보 저장 파일이 위치할 경로 지정

history

저장될 모든 태스크 히스토리에 대한 정보를 정의

history.filename

모든 태스크의 진행 내역 히스토리 정보를 저장할 파일 이름 지정. XML 형식으로 저장 됩니다.

기본 값: "history.xml"

history.basedir

히스토리 정보 저장 파일이 위치할 경로 지정

history.ageInDays

태스크의 히스토리 정보가 히스토리 저장 파일 안에 보관될 일 수를 지정

기본 값: "5"

histories

각각 태스크 별로 저장될 히스토리에 대한 정보를 정의

histories.basedir

태스크 별로 개별 히스토리 정보 저장 파일을 저장할 경로 지정

histories.max

개별 태스크의 히스토리 정보 저장 파일 안에 보관할 최대 엔트리 수를 지정.
이 수에 도달하면 이전 엔트리부터 차례대로 삭제 됩니다.

기본 값: "100"

예:

```
<agent name="SCHEDULER" type="SCHEDULER" desc="Task Scheduler"
class="com.windfire.agents.scheduler.asysAgentSched" opclass="/servlet/asysServerMets.asysOperSched"
msgfile="msg/scheduler.mc" minhealth="1">
  <info view="20" datetimeformat="yyyy-MM-dd 'at' HH:mm:ss a" dateformat="yyyy-MM-dd"
timeformat="HH:mm:ss a" encoding="Unicode"/>
  <tasks filename="tasks.xml" basedir="C:/SpenoCom/Xtorm/scheduler"/>
  <history filename="history.xml" basedir="C:/SpenoCom/Xtorm/scheduler" ageInDays="5"/>
  <histories basedir="C:/SpenoCom/Xtorm/scheduler/history" max="100"/>
</agent>
```

중첩 엘리먼트 카테고리 - 서버리스트

"ServerList" 항목은 이 구성의 일부분인 서버에 대한 일반정보를 정의합니다. 여기에서 정의하는 서버는 웹 운영 어플리케이션 상에 나타나며 서버간에 쉽게 해당 페이지로 점프할 수 있습니다.

속성

serverlist

웹 운영 어플리케이션 상에서 서버 간 점프 기능을 사용하기 위하여 서버 리스트를 지정합니다.

externserver

각각의 서버에 대하여 서버 이름 및 포트번호를 정의합니다. 동일 서버 또는 이종 컴퓨터에 여러 서버를 정의할 수 있습니다.

address

각각의 서버에 대하여 서버 이름 및 포트번호를 정의 합니다.

name

웹 운영 어플리케이션 상에서 서버를 표시할 서버 이름 또는 캡션 이름. 대규모 분산 환경에서, 이 영역에 표시할 분산 서버 리스트가 많은 경우 단지 IP 주소로만으로는 각각의 서버를 구분하기 어려운 경우, 서버의 특성에 맞게 서버의 설명을 입력합니다. 이 속성은 선택적 항목입니다. 이 속성을 지정하지 않으면 address 속성에서 정의한 값으로 외부 서버가 표시됩니다. 단, 이 속성에 표시 이름을 지정할 때, 운영 어플리케이션에서 문자열이 잘리지 않도록 10 ~ 15자 내외로 입력하는 것이 적당합니다.

예:

```
<serverlist>
  <externserver address="hplaptop:2102" name="hplaptop"/>
  <externserver address="hdual:2102"/>
</serverlist>
```

Configuration File 여제

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<Xtorm>
```

```
  <server basemsg="msg/base.mc" deleteconf="false">
    <logwriter class="com.windfire.agents.asysLogWriterFile">
      <file name="log/server.log" truncate="false" encoding="Unicode"
dateformat="yyyy-MM-dd 'at' HH:mm:ss" tmode="1" tparam="10000"/>
    </logwriter>

    <!--
    <logwriter class="com.windfire.agents.asysLogWriterNTEvent">
      <file name="log/server.log" truncate="false"/>
    </logwriter>
  -->
```

```

<logwriter class="com.windfire.agents.asysLogWriterConsole"/>

<debugtracer class="com.windfire.agents.asysDebugTracerFile">
    <debugger id="debug1" file="log/debug.log" size="10240" encoding="Unicode"
dateformat="yyyy-MM-dd 'at' HH:mm:ss"/>
    <debugger id="debug2" file="log/migDebug.log" size="10240"/>
</debugtracer>

</server>

<localagents timediff="540" machineid="0" >

    <!-- LAUNCHER -->
    <agent name="LAUNCHER" type="LAUNCHER"
class="com.windfire.agents.launcher.asysAgentLauncher"
opclass="/servlet/asysservlets.asysOperLauncher" msgfile="../../msg/launcher.mc">
        <sockets port="2100" dport="2102" dhost="10.1.1.1,10.1.1.2" />
        <sockets port="2101" dport="2103" dhost="10.1.1.1,10.1.1.2" />
        <checker iteration="1" idletime="1" />
    </agent>

    <!-- COMM -->
    <agent name="COMM" type="COMM"
class="com.windfire.agents.comm.asysAgentComm"
opclass="/servlet/asysservlets.asysOperComm" msgfile="../../msg/comm.mc">
        <sockets hostname="10.1.1.1" port="2102" debug="NO" virtualhost="HDUAL"
virtualip="10.1.1.1" virtualport="2100" />
    </agent>

    <!-- Data -->
    <agent name="DATA" type="DATA" class="com.windfire.agents.data.asysAgentData"
msgfile="msg/data.mc" minhealth="1">

        <dbpools>

            <pool name="master" driver="oracle.jdbc.driver.OracleDriver"
connect="jdbc:oracle:thin:@hdual:1521:XT" user="Xtorm" pswd="Xtorm" count="10"
preconnect="false" unicode="false" dbtype="2" maxtrans="400" tableprefix="ecm.XT."/>
            <pool name="instance1" driver="com.jnetdirect.jsql.JSQLDriver"
connect="jdbc:JSQlConnect://localhost/database=Xtorm" user="sa" pswd="" count="10"
preconnect="false" unicode="TRUE" dbtype="1" maxtrans="400"
tableprefix="ecm.Xtorm."/>
            <pool name="instance2" driver="com.ibm.db2.jcc.DB2Driver"
connect="jdbc:db2://scooby:50000/Xtorm:driverType=4;" user="Administrator"
pswd="adminsRus2" count="10" dbtype="3" maxtrans="400" tableprefix="ecm.XT."/>
            <pool name="instance3" driver="com.sybase.jdbc2.jdbc.SybDriver"
connect="jdbc:sybase:Tds:daphne:2048/Xtorm" user="sa" pswd="" count="10"
preconnect="false" unicode="false" dbtype="4" maxtrans="400" tableprefix="ecm.XT."/>
            <pool name="instance4" driver="com.mysql.jdbc.Driver"
connect="jdbc:mysql://localhost:3306/Xtorm" user="root" pswd="root" count="20"
preconnect="false" unicode="false" dbtype="5" maxtrans="400" tableprefix="ecm.XT."/>
            <pool name="instance5" driver="Altibase.jdbc.driver.AltibaseDriver"
connect="jdbc:Altibase://localhost:20300/mydb" user="Xtorm" pswd="Xtorm" count="20"
preconnect="false" unicode="false" dbtype="6" maxtrans="400" tableprefix="ecm.XT."/>
        </dbpools>
    </agent>

```

```

        <pool name="instance6" driver=" com.informix.jdbc.IfxDriver" connect="
jdbc:informix-sqli://HDUAL:1526/Xtorm:INFORMIXSERVER=IMX;
user=informix;password=manager" user="" pswd="" count="20" unicode="false"
dbtype="7" maxtrans="400" tableprefix="ecm.XT."/>
        <pool name="instance7" driver=" com.tmax.tibero.jdbc.TbDriver"
connect="jdbc:tibero:thin:@10.90.1.41:8629:tibero" user="Xtorm" pswd="Xtorm"
count="10" preconnect="false" unicode="false" dbtype="8" maxtrans="400"
tableprefix="ecm.XT."/>
        <pool name="master" driver="org.postgresql.Driver"
connect="jdbc:postgresql://localhost:5432/XEDM" user="Xtorm" pswd="Xtorm" count="10"
preconnect="FALSE" unicode="FALSE" dbtype="9" />
    </dbpools>

    <startup mainpool="master" localtime="GMT+09:00"/>
    <listspooler listdir="spool" memcache="250" timeout="600"/>
    <defaultaudit agentname="AUDIT" agentserver="HDUAL:2102" />

    <farm enabled="false" ucode="0" farmeid="true" >
        <dserver address="10.1.1.2" port="2102" />
    </farm>
</agent>

<!-- Archive -->

<!-- File System -->

<agent name="MAIN" type="ARCHIVE" desc="Local Disk"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asyservlets.asysOperArchive" msgfile="msg/archive.mc"
minhealth="1">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
        <deviceIF class="com.windfire.agents.archive.asysAgentArMediaFilesys"
basedir="c:/Xtorm/storage" hashdirs="10" retention="240" dateformat="yyyyMMdd"
batch="true" />
        <threadpool ops="read,write,delete" count="20"/>
    </scheduler>
</agent>

<!-- EMC -->

<agent name="BCH_CACHE_DT" type="ARCHIVE" desc="EMC BATCH CACHE DIRECT"
class="com.alerotech.agents.archive.asysAgentArchive"
opclass="/servlet/scservlets.asysOperArchive" msgfile="../../msg/archive.mc"
minhealth="1">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
        <deviceIF
class="com.windfire.agents.archive.asysAgentArMediaEMCCenteraBatchCacheDirect"
basedir="storage/EMCBATCH" tempdel="true" pooladdrlist="10.1.1.1,10.1.1.2?
D:/us2_profile1_rqecw.pea" poolbuffersize="16" timeout="120" retention="300"
c_avoidance="true" blobpartialread="true"/>
        <threadpool ops="read,write,delete" count="20"/>
    </scheduler>
</agent>

<!-- HCAP (HTTP) -->

```

```

<agent name="HCAP_ARC" type="ARCHIVE" desc="HCAP Storage"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asysservlets.asysOperArchive" msgfile="../../msg/archive.mc"
minhealth="1">

    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
        <deviceIF class="com.windfire.agents.archive.asysAgentArMediaHTTP"
baseurl="http://10.13.10.60;http://10.13.10.61;http://10.13.10.62"
basedir="fcfs_data/Xtorm/" hashdirs="500" curetype="1" secureuser="" securepswd=""
securedomain="" secureserver="" tempdir="/sw/xtorm/temp" retries="3" retrywait="5000"
failnodeafter="3" failnodewait="5000" retention="300"/>
        <threadpool ops="read,write,delete" count="20"/>
    </scheduler>
</agent>

<agent name="HCP_ARC" type="ARCHIVE" desc="HCPStorage"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asysservlets.asysOperArchive" msgfile="../../msg/archive.mc"
minhealth="1">
<scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaHCP"
baseurl="http://eXtormtest.XtormTest.hisbmt.hcp.com"
    basedir="rest" hashdirs="1000" securetype="1" secureuser="service"
securepswd="passwd9" securedomain="eXtormtest.XtormTest.hisbmt.hcp.com" secureserver=""
tempdir="c:/temp" TIMEOUT="5000" retrywait="1000" failnodeafter="0" failnodewait="0"
retention="" MAXTOTALCONNECTIONS="100" MAXHOSTCONNECTIONS="100" FILESYSTEMKMB
="true" />
    <threadpool ops="read,write,delete" count="20"/>
</scheduler>
</agent>

<!-- SnapLock -->

<agent name="SNARC" type="ARCHIVE" desc="SnapLock"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asysservlets.asysOperArchive" msgfile="../../msg/archive.mc">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
        <deviceIF
class="com.windfire.agents.archive.asysAgentArMediaSnapLock"
basedir="d:/STORAGE/MAIN" hashdirs="500" retentiontype="0" retention="1800"
checkhealth="false" MOUNTPT="" VOLUME="" filerlist="filer1.lst" filercomparelength="11"
FILER="10.10.1.7" USER="root" PASSWD="hanaceda" INFO="TRUE"/>
        <threadpool ops="read,write,delete" count="20"/>
    </scheduler>
</agent>

<!-- Avamar -->

<agent name="AVMARC" type="ARCHIVE" desc="AVAMAR backup"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asysservlets.asysOperArchive" msgfile="../../msg/archive.mc">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo"
ITERATIONOP="300" RSTIME="60">
        <deviceIF class="com.windfire.agents.archive.asysAgentArMediaAvamar"
basedir="/storage/AVAMAR" pooladdrlist="sut1,sut2,sut3"
hostloclist="/opt/AVMCInt1,/opt/AVMCInt2,/opt/AVMCInt3" tempdel="true"
cmdshell="../../bin/avamarctl" errlog="log/AVAMAR" retention="604800" />
        <threadpool ops="read,write,delete" count="20"/>
    </scheduler>

```

```

</agent>

<!-- KT block chain -->
<agent name="KTDATA_ARC" type="ARCHIVE" desc="HCAP Storage"
class="com.windfire.agents.archive.asysAgentArchive"
opclass="/servlet/asyservlets.asysOperArchive" msgfile="../../msg/archive.mc" minhealth="1">
    <scheduler class="com.windfire.agents.archive.asysAgentArSchedFifo">
    <deviceIF class="com.windfire.agents.archive.asysAgentArMediaDataChain"
baseurl="14.63.172.220:18080/dcedm"
    basedir="/file" SECRETCODE="" tempdir="c:/temp" TIMEOUT="0"
DELETEDAGE="300" retrywait="3000" failnodeafter="3" failnodewait="3000" retention=""
    MAXTOTALCONNECTIONS="100" MAXHOSTCONNECTIONS="20"/>
    <threadpool ops="read,write,delete" count="5"/>
    </scheduler>
</agent>

<!-- CACHECLEAN -->

<agent name="CACHEMNGR" type="CACHECLEAN" desc="Cache Cleaner"
class="com.windfire.agents.cache.asysAgentCache"
opclass="/servlet/asyservlets.asysOperCache" msgfile="msg/cache.mc" minhealth="1">
    <logging basedir="c:/Xtorm/log" maxentries="10" dateformat="yyyy-MM-dd 'at'
HH:mm:ss" encoding="Unicode" />
    <cache agentDataServer="" target="Xtorm_MAIN" cache="*">
        <schedule run="auto" startafter="10" runevery="30" />
        <clean pctfree="50" />
    </cache>
</agent>

<!-- HPI -->

<agent name="IMPORT" type="IMPORT" desc="Import Data"
class="com.windfire.agents.importagent.asysAgentImport"
opclass="/servlet/asyservlets.asysOperImport" msgfile="msg/import.mc" minhealth="1">
    <startup bundlesize="100" threads="20" highWaterBundle="20"
lowWaterBundle="10" highWaterElement="1000" lowWaterElement="800" delete="yes"
postProcess="yes" agentData="DATA" target="Xtorm_MAIN"
agentDataServer="HDUAL:2102" encoding="Unicode" retryInMins="5" />
    <recovery filename="recovery.xml" basedir="c:/Xtorm/log"/>
    <statistic filename="statistic.xml" basedir="c:/Xtorm/log" ageInDays="5"/>
    <error filename="error.xml" basedir="c:/Xtorm/log" ageInDays="5"/>
    <importgateway name="IMPORTGATEWAY"
class="com.windfire.agents.importagent.gateway.asysNativeIGContainer"
basedir="c:/ImportData" indexid="Checks" eClassId="IMAGE"
userSecureClassId="PUBLIC" archiveld="MAIN" interval="20" start="no" pvdelimeter=":"
fdelimiter="," tfdelimeter="_" encoding="" storproc="sp_usrAddChecks(EID,
KYOHWAN_DATE,INDEX_NO, SUPYO_NO,JI_BANKCODE, JI_JIJUMCODE,
KWONCODE, TOT_AMOUNT">
    </importgateway>
    <retry>
        <code reason="403" description="No database connection"/>
        <code reason="303" description="Service down"/>
    </retry>
</agent>

```

```
<!-- AUDIT -->
```

```
<agent name="AUDIT" type="AUDIT" desc="Audit agent"
class="com.windfire.agents.audit.asysAgentAudit"
opclass="/servlet/asysservlets.asysOperAudit" msgfile="msg/audit.mc">
  <auditwriter name="Audit file writer" desc="Audit file writer"
class="com.windfire.agents.audit.asysAgentAuditWriterFile" encoding="ISO-8859-1"
dateformat="yyyy-MM-dd HH:mm:ss" opclass="/servlet/asysservlets.asysOperAuditFile"
id="file">
    <file basedir=" C:/SpenoCom/Xtorm/log" maxsize="1000"
deleteonfull="YES" fielddelimiter="|" filename="auditfile.log">
        <field subCmd="subCmd" />
        <field Gateid="id" />
        <field srcAgent="srcAgent" />
        <field srcMach="srcMach" />
        <field userid="userid" />
        <field detail="*" />
    </file>
</auditwriter>

  <auditwriter name="Audit buffer writer" desc="Audit buffer writer"
class="com.windfire.agents.audit.asysAgentAuditWriterBuffer" encoding="ISO-8859-1"
dateformat="yyyy-MM-dd HH:mm:ss" opclass="/servlet/asysservlets.asysOperAuditBuffer"
id="buffer">
    <buffer maxrecord="100">
        <field subCmd="subCmd" />
        <field Gateid="id" />
        <field srcAgent="srcAgent" />
        <field srcMach="srcMach" />
        <field targetMach="targetMach" />
        <field userid="userid" />
        <field detail="*" />
    </buffer>
</auditwriter>

  <auditwriter name="Audit table writer" desc="Audit table writer"
class="com.windfire.agents.audit.asysAgentAuditWriterTable" encoding="unicode"
opclass="/servlet/asysservlets.asysOperAuditTable" agentData="DATA"
agentDataServer="LOCALHOST:2102" sqlgateway="SIMPLESQL_MAIN" dbtype="1"
id="table">
    <table tablename="UT_AUDIT" >
        <column subCmd="subCmd" length="24" type="char" />
        <column Gateid="engine" length="24" type="char" />
        <column srcAgent="srcAgent" length="24" type="char"/>
        <column srcMach="srcMach" length="24" type="char"/>
        <column userid="userid" length="24" type="char"/>
        <column requestMach="targetMach" length="24" type="char"/>
        <column requestDetail="targetAgent" length="24" type="char"/>
        <column downIpAddr="ipaddr" length="24" type="char"/>
        <column downHostName="hostname" length="24" type="char"/>
        <column elementId="dstEid" length="256" type="char"/>
        <column etc="*" length="256" type="varchar"/>
    </table>
</auditwriter>
</agent>
```

```
<!-- MIGRATION -->
```

```

<agent name="MIGRATION" type="MIGRATION" desc="Migration Manager"
class="com.windfire.agents.migration.asysAgentMig"
opclass="/servlet/asysservlets.asysOperMig" msgfile="msg/migration.mc" minhealth="1">
    <logging basedir="c:/Xtorm/log" maxentries="10" dateformat="yyyy-MM-dd 'at'
HH:mm:ss" encoding="Unicode"/>
    <defaults maxreqs="30" bundlesize="50" aborttime="300" timediff="540"
maxignoreerrors="100"/>
    <cclass agentDataServer="HDUAL" target="Xtorm_MAIN" cclass="TEST"
maxreqs="5" bundlesize="50" putfilters="Encrypt1" filterparms="" manual="YES"
MAXLISTSIZE="20000" migstat="TRUE" />
    <cclass agentDataServer="HDUAL" target="Xtorm_MAIN" cclass="BASIC"
maxreqs="5" bundlesize="50" putfilters="Encrypt2" filterparms="" manual="YES" />
</agent>

```

```

<!-- SCHEDULER -->

```

```

<agent name="SCHEDULER" type="SCHEDULER" desc="Task Scheduler"
class="com.windfire.agents.scheduler.asysAgentSched"
opclass="/servlet/asysservlets.asysOperSched" msgfile="msg/scheduler.mc"
minhealth="1">
    <info view="20" datetimeformat="yyyy-MM-dd 'at' hh:mm:ss a" dateformat="yyyy-
MM-dd" timeformat="HH:mm:ss" encoding="Unicode"/>
    <tasks filename="tasks.xml" basedir="c:/scheduler"/>
    <history filename="history.xml" basedir="c:/scheduler" ageInDays="5"/>
    <histories basedir="c:/scheduler/history" max="100"/>
</agent>

```

```

<!-- PROCESS -->

```

```

<agent name="PROCESS" type="PROCESS" desc="Process Manager"
class="com.windfire.agents.process.asysAgentProcess"
opclass="/servlet/asysservlets.process.asysOperProcess" msgfile="msg/process.mc">
    <processgateway target="XPROC_MAIN" agentDataServer="HDUAL:2102" />
    <Xtormgateway target="Xtorm_MAIN" agentDataServer="HDUAL:2102"
defeclass="WFOLDER" defincattr="1" defdescr="Default" defpri="2"/>
    <local editdir="process" dateformat="yyyy-MM-dd 'at' HH:mm:ss" />
</agent>

```

```

<!-- PROCREMOTE -->

```

```

<agent name="PROCESS_INV" type="PROCREMOTE" desc="Process Remote"
class="com.windfire.agents.process.invoke.asysAgentProclInvoke" opclass="/servlet/
yservlets.process.asysOperProcess" msgfile="msg/process.mc"
/>
</agent>

```

```

</localagents>

```

```

<serverlist>

```

```

    <externserver address="HDUAL1:2102" name="HDUAL1"/>
    <externserver address="HDUAL2:2102"/>

```

```

</serverlist>

```

</Xtorm>

프롤로그

프롤로그는 XML 데이터가 시작하기 전에 정의하는 XML 문서의 한 부분입니다. 프롤로그는 특정 문서를 XML 문서로 구분해주는 선언 부분을 포함하고 있습니다.

version

데이터 부분에서 사용한 XML 마크업 언어의 버전을 정의 합니다. 이 값은 선택적이지 않습니다.

encoding

데이터를 인코딩하는 문자 셋을 정의. "ISO-8859-1"는 "Latin-1" 즉, Western European 및 English 언어 문자 셋 입니다. (기본은 유니코드 (UTF-8) 방식으로 압축됩니다.)

예:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```