

SAP Business Objects
Document Version: 4.0 Support Package 6 - 2013-09-02

데이터 연합 관리 도구 가이드

목차

1	데이터 연합 서비스의 관리 및 조정 소개.....	5
1.1	데이터 연합 서비스 관리 및 조정 소개.....	5
2	데이터 연합 관리 도구 사용.....	6
2.1	데이터 연합 관리 도구의 용도.....	6
2.2	데이터 연합 관리 도구 시작.....	6
2.3	데이터 연합 관리 도구에 대한 관리 권한이 있는 사용자 추가.....	6
2.4	Microsoft Active Directory 인증에 대해 데이터 연합 관리 도구 구성.....	7
2.5	데이터 연합 관리 도구 세션 로그아웃.....	7
2.6	데이터 연합 쿼리 엔진에서 실행 중인 쿼리 검사.....	7
2.7	데이터 연합 쿼리 서버에서 SQL 쿼리 테스트.....	7
2.8	데이터 연합 쿼리 엔진의 쿼리 계획 방법 확인.....	8
2.9	데이터 연합 쿼리 서버에서 실행된 쿼리 기록 검색.....	8
2.10	메타데이터 쿼리.....	8
2.11	쿼리 취소.....	9
2.11.1	쿼리 취소.....	9
2.12	데이터 연합 관리 도구의 쿼리 패널 탭.....	9
2.13	데이터 연합 관리 도구의 쿼리 모니터링 탭.....	11
2.14	데이터 연합 관리 도구의 시스템 매개 변수 탭.....	13
2.15	데이터 연합 관리 도구의 커넥터 구성 탭.....	14
2.16	데이터 연합 관리 도구의 통계 탭.....	16
2.17	데이터 연합 관리 도구에서 속성 뷰 표시.....	18
2.18	데이터 연합 관리 도구에서 SSL 에 대해 구성된 서버에 연결.....	18
3	쿼리 최적화.....	19
3.1	데이터 연합 쿼리의 성능 조정.....	19
3.2	시스템 매개 변수를 사용하여 메모리 사용 최적화.....	19
3.2.1	메모리를 사용하는 연산자.....	20
3.3	통계를 사용하여 응용 프로그램에서 소스 쿼리에 최적인 알고리즘 선택.....	21
3.3.1	열 카디널리티 정보.....	21
3.3.2	열 간의 관계를 나타내는 팬아웃 값 정보.....	21
3.3.3	보고서 최적화에 필요한 통계만 계산하도록 기록된 통계 필터링.....	22
3.4	쿼리 계획 최적화.....	22
3.4.1	데이터 연합 관리 도구의 쿼리 계획 뷰.....	22
3.4.2	통계 설명 명령.....	23
3.4.3	쿼리 설명 기능을 사용해 피드백을 얻어 쿼리 조정.....	24
3.4.4	데이터 연합 관리 도구를 사용하여 연산자가 무시되고 있는지 여부 확인.....	25
3.4.5	시스템 매개 변수를 사용하여 큰 테이블에 조인된 작은 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침.....	25

3.4.6	시스템 매개 변수를 사용하여 정렬 가능한 데이터가 포함된 큰 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침.	27
3.4.7	시스템 매개 변수를 사용하여 순서 기준 연산자의 활성화 제어.	28
3.4.8	강제로 데이터 소스 하위 쿼리 병렬 실행.	29
3.4.9	semi-join 실행 전략.	29
3.5	특정 커넥터 최적화.	29
3.5.1	SAP NetWeaver BW 에 대한 병렬 쿼리의 콜백 동시성 증가.	29
3.5.2	SAP NetWeaver BW 에 대한 쿼리에서 응답 패키지 크기 변경.	30
3.6	데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기.	30
4	데이터 소스로 커넥터 구성.	31
4.1	데이터 연합 관리 도구에서 커넥터 정보 보기.	31
4.2	데이터 연합 관리 도구에서 커넥터 속성 변경.	31
4.3	관계형 데이터 소스에 대한 커넥터 구성.	31
4.3.1	관계형 데이터 소스의 공통 커넥터 속성 목록.	31
4.3.2	MySQL 데이터 소스의 특정 커넥터 속성 목록.	34
4.3.3	Teradata 데이터 소스의 특정 커넥터 속성 목록.	34
4.3.4	Sybase ASE 데이터 소스의 특정 커넥터 속성 목록.	34
4.3.5	SQL Server 데이터 소스의 특정 커넥터 속성 목록.	35
4.3.6	Generic ODBC 또는 JDBC 데이터 소스의 특정 커넥터 속성 목록.	35
4.3.7	Oracle 데이터 소스의 특정 커넥터 속성 목록.	37
4.3.8	SAP Hana 데이터 소스의 특정 커넥터 속성 목록.	37
4.3.9	MaxDB 데이터 소스의 특정 커넥터 속성 목록.	37
4.4	SAS 용 커넥터 구성.	38
4.4.1	SAS 데이터 소스의 커넥터 속성 목록.	38
4.4.2	카디널리티를 기준으로 from 절에서 테이블을 정렬하여 SAS 쿼리 최적화.	43
4.5	SAP NetWeaver BW 에 대한 커넥터 구성.	44
4.5.1	SAP NetWeaver BW 데이터 소스의 커넥터 속성 목록.	44
4.5.2	SAP NetWeaver BW 가 데이터 연합 서비스에 연결하는 데 사용하는 콜백 ID 를 수동으로 설정	47
4.5.3	SAP NetWeaver BW 연결에 대한 콜백 ID 정리.	48
4.5.4	SAP 분석 권한을 사용하여 자동으로 데이터 필터링.	48
4.5.5	다중 소스 유니버스의 SAP NetWeaver BW 연결 아키텍처.	51
4.5.6	다중 소스 유니버스의 SAP NetWeaver BW 연결 콜백 시퀀스.	51
4.6	데이터 연합 관리 도구를 사용하여 관계형 커넥터 및 SAS 커넥터 기능 설정.	51
4.7	관계형 데이터 소스의 전체 커넥터 기능 목록.	52
5	시스템 및 세션 매개 변수 관리.	54
5.1	시스템 및 세션 매개 변수 정보.	54
5.2	데이터 연합 관리 도구를 사용하여 시스템 매개 변수 변경.	54
5.3	데이터 연합 관리 도구를 사용하여 세션 매개 변수 변경.	54
5.4	데이터 연합 관리 도구를 사용하여 관계형 커넥터 및 SAS 커넥터 기능 설정.	54

5.5	시스템 매개 변수 목록.....	55
5.6	세션 매개 변수 목록.....	64
5.7	데이터 연합 응용 프로그램의 데이터 정렬.....	64
5.7.1	데이터 연합 응용 프로그램에서 지원되는 데이터 정렬.....	65
5.7.2	데이터 연합 SQL 쿼리에 대해 문자열 정렬 및 문자열 비교 동작 설정.....	66
5.7.3	이진 데이터 정렬 사용 시 데이터 연합 응용 프로그램에서 소스에 쿼리를 푸시하는 방법을 결정하는 방식.....	68
6	SQL 구문 참조.....	69
6.1	데이터 연합 쿼리 엔진용 쿼리 언어.....	69
6.1.1	식별자 및 명명 규칙.....	69
6.1.2	데이터 연합 쿼리 엔진에 사용된 데이터 형식.....	71
6.1.3	문.....	74
6.1.4	식.....	75
6.1.5	설명.....	78
6.2	SELECT 절의 문법.....	78
7	용어집.....	85
7.1	용어 및 설명.....	85
8	문제 해결.....	86
8.1	데이터 연합 서비스 로그 정보.....	86
8.2	SAP NetWeaver BW 데이터 소스에서 장기 실행 쿼리로 인한 연결 종료.....	86
8.3	SAP NetWeaver BW 커넥터의 NoClassDefFoundError: CpicDriver 오류.....	86

1 데이터 연합 서비스의 관리 및 조정 소개

1.1 데이터 연합 서비스 관리 및 조정 소개

데이터 연합 서비스를 관리하거나 조정하려면 데이터 연합 관리 도구를 사용합니다.

관리

서비스의 데이터 처리 방식과 관련된 데이터 연합 서비스 요소를 관리해야 할 경우 데이터 연합 관리 도구를 사용합니다. 이러한 요소로는 데이터 소스와 관련된 커넥터 속성 관리, 메모리 구성, 데이터 연합 쿼리 엔진의 쿼리에 적용되는 매개 변수 설정 등이 있습니다.

데이터 연합 관리 도구를 사용하면 커넥터를 검색 및 관리하고, 데이터 소스를 찾아보고, 데이터 소스에 대해 쿼리를 실행하고, 통계를 관리하고, 과거 쿼리 및 실행 중인 쿼리의 목록을 볼 수 있습니다. 과거 쿼리 또는 실행 중인 쿼리의 목록을 보고자 할 경우 이러한 목록은 실제 시스템에 있으므로 보고 응용 프로그램에서는 사용자 개입 없이 쿼리를 생성하여 쿼리 서버로 보냅니다. 생성된 쿼리를 통해 사용자는 필요한 작업이 시스템에서 수행되고 있는지 확인할 수 있습니다.

일반적인 관리(예: 사용자 계정 또는 로깅 관리)에는 데이터 연합 서비스가 설치된 플랫폼의 도구를 사용합니다.

조정

데이터 소스의 데이터에 따라 커넥터 또는 쿼리를 적용하려는 경우 조정을 위해 데이터 연합 관리 도구를 사용합니다.

조정 작업에는 각 데이터 소스로 최대한 많은 작업을 전달할 수 있도록 각 커넥터의 기능을 설정하고, 각 데이터 소스에 대해 적절한 통계를 설정하고, 서버로 보낼 각 쿼리를 최적화하도록 매개 변수를 구성하는 작업이 포함됩니다. 일반적으로 최적화된 데이터 소스가 최대한 많은 작업을 처리하도록 하고 네트워크를 통해 최소한의 데이터를 전송하는 것입니다. 데이터 연합 서비스는 소스에 작업을 푸시하고 데이터 전송을 줄일 수 있는 여러 가지 옵션과 함께 시스템의 쿼리 처리 방식을 파악하는 데 유용한 도구를 제공합니다.

2 데이터 연합 관리 도구 사용

2.1 데이터 연합 관리 도구의 용도

데이터 연합 관리 도구는 사용이 간편한 데이터 연합 서비스 관리 기능을 제공하는 Rich Client 응용 프로그램입니다.

SAP BusinessObjects Enterprise 플랫폼에 완벽히 통합된 데이터 연합 서비스를 통해 서로 다른 데이터 소스에 쿼리를 배포하여 다중 소스 유니버스를 사용하고 단일 데이터 연합을 통해 데이터를 통합할 수 있습니다.

데이터 연합 관리 도구를 통해 데이터 연합 쿼리를 최적화하고 데이터 연합 쿼리 엔진을 세밀하게 조정하여 성능을 최적화할 수 있습니다.

데이터 연합 관리 도구는 다음 작업을 수행하는 데 사용됩니다.

- SQL 쿼리를 테스트합니다.
- 연합 쿼리를 각 소스에 배포하는 방법을 자세히 설명하는 최적화 계획을 시각화합니다.
- 통계를 계산하고 시스템 매개 변수를 설정하여 데이터 연합 서비스를 세밀하게 조정하고 성능을 최적화합니다.
- 커넥터 수준에서의 데이터 소스별 쿼리 실행 방식을 제어하는 속성을 관리합니다.
- 실행 중인 SQL 쿼리를 모니터링합니다.
- 실행된 쿼리의 기록을 찾아봅니다.

관련 링크

[데이터 연합 쿼리 서버에서 SQL 쿼리 테스트](#) [페이지 7]

[데이터 연합 쿼리 엔진의 쿼리 계획 방법 확인](#) [페이지 8]

[데이터 연합 쿼리 서버에서 실행된 쿼리 기록 검색](#) [페이지 8]

[데이터 연합 관리 도구에서 커넥터 정보 보기](#) [페이지 31]

[통계를 사용하여 응용 프로그램에서 소스 쿼리에 최적인 알고리즘 선택](#) [페이지 21]

2.2 데이터 연합 관리 도구 시작

1. **시작** > **프로그램** > *BusinessObjects Data Federator XI Release 4* > **데이터 연합 관리 도구** 를 클릭합니다.
2. 시스템 이름, 사용자 이름 및 암호를 입력한 다음 **확인**을 클릭합니다.

2.3 데이터 연합 관리 도구에 대한 관리 권한이 있는 사용자 추가

SAP BusinessObjects Enterprise 서버에서 데이터 연합 관리자라는 사용자 그룹에는 데이터 연합 서비스를 관리할 수 있는 권한이 부여됩니다.

그룹에 사용자를 추가하는 방법은 *SAP BusinessObjects Enterprise* 관리자 가이드를 참조하십시오.

2.4 Microsoft Active Directory 인증에 대해 데이터 연합 관리 도구 구성

Active Directory 인증에 대해 데이터 연합 관리 도구를 구성하려면 데이터 연합 관리 도구의 초기화 파일을 편집해야 합니다. 이 파일에서 두 개의 구성 파일(로그인 구성 파일과 Kerberos 구성 파일)을 가리켜야 합니다.

1. <설치 디렉터리>\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\win32_x86\DFAdministrationTool.ini 파일을 편집합니다.

파일 끝에 다음 줄을 추가합니다.

```
-Djava.security.auth.login.config=<path-to-bsclogin>\bscLogin.conf  
-Djava.security.krb5.conf=<path-to-kerberos>\krb5.ini
```

예를 들면 다음과 같습니다.

```
-Djava.security.auth.login.config=C:\WINNT\bscLogin.conf  
-Djava.security.krb5.conf=C:\WINNT\krb5.ini
```

2. 두 개의 파일(bscLogin.conf 와 krb5.ini)이 Kerberos 를 사용하는 Active Directory 인증에 대해 구성되었는지 확인합니다.

자세한 내용은 SAP BusinessObjects Enterprise 관리자 가이드의 Windows AD 용 Kerberos 인증 사용 단원을 참조하십시오.

2.5 데이터 연합 관리 도구 세션 로그아웃

도구 모음 왼쪽 맨 위에 있는 로그아웃 단추를 클릭합니다.

2.6 데이터 연합 쿼리 엔진에서 실행 중인 쿼리 검사

1. 데이터 연합 관리 도구를 시작합니다.
2. 쿼리 모니터링 탭을 클릭합니다.
3. 새로 고침을 클릭합니다.
실행 중인 쿼리 창에 실행 중인 쿼리가 표시됩니다.

2.7 데이터 연합 쿼리 서버에서 SQL 쿼리 테스트

1. 데이터 연합 관리 도구를 시작합니다.
2. 쿼리 패널 탭을 클릭합니다.

3. 쿼리를 입력합니다.
4. **실행**을 클릭하여 쿼리를 실행합니다.
쿼리가 실행되고 결과가 **쿼리 결과** 패널에 표시됩니다.

2.8 데이터 연합 쿼리 엔진의 쿼리 계획 방법 확인

데이터 연합 쿼리 엔진은 SQL 쿼리를 분석하여 사용자가 여러 소스에서 정확한 데이터를 최대한 빨리 가져올 수 있도록 SQL 쿼리를 변환하는 방법을 결정합니다. 이 분석을 수행하기 위해 쿼리 엔진은 데이터 소스 간에 최대한 많은 작업을 배포하고, 하위 쿼리를 작성하여 최종 결과를 산출하는 데 필요한 데이터만 네트워크를 통해 반입합니다.

설명 도구를 사용하여 소스 간에 쿼리가 배포된 방식을 확인할 수 있습니다.

1. 데이터 연합 관리 도구를 시작합니다.
2. **쿼리 패널** 탭을 클릭합니다.
3. 보려는 쿼리를 입력합니다.
4. **실행** 옆에 있는 화살표를 클릭한 다음 **쿼리 설명**을 클릭합니다.

쿼리 엔진이 생성한 계획으로 쿼리가 나타납니다.

관련 링크

[데이터 연합 관리 도구의 쿼리 계획 뷰](#) [페이지 22]

2.9 데이터 연합 쿼리 서버에서 실행된 쿼리 기록 검색

사용자 또는 응용 프로그램이 쿼리를 데이터 연합 쿼리 서버로 이미 보낸 경우 데이터 연합 관리 도구를 사용하여 해당 쿼리 목록을 볼 수 있습니다.

1. 데이터 연합 관리 도구를 시작합니다.
2. **쿼리 모니터링** 탭을 클릭합니다.
실행된 쿼리 창에 실행된 쿼리가 표시됩니다.

2.10 메타데이터 쿼리

테이블의 특정 집합과 작동하도록 하드 코딩되지 않은 동적 응용 프로그램에는 연결 대상 데이터베이스에서 개체의 구조 및 특성을 확인하는 메커니즘이 있어야 합니다. 이러한 응용 프로그램에는 다음과 같은 정보가 필요할 수 있습니다.

- 대상과 데이터 소스에 있는 테이블의 수 및 이름
- 테이블의 열 수를 비롯하여 각 열의 이름, 데이터 형식, 소수 자릿수 및 전체 자릿수
- 테이블에 대해 정의된 키

데이터 연합 쿼리 엔진을 기반으로 하는 응용 프로그램은 다음과 같은 저장 프로시저를 사용하여 시스템 카탈로그의 정보에 액세스할 수 있습니다.

```
CALL getTables '<name-of-catalog>', '%', '%'
CALL getColumns '<name-of-catalog>', '<name-of-schema>', '<name-of-table>', '%'
CALL getKeys '<name-of-catalog>', '<name-of-schema>', '<name-of-table>'
```

2.11 쿼리 취소

데이터 연합을 사용하는 경우 명령을 통해 실행 중인 모든 쿼리 또는 실행 중인 특정 쿼리를 취소할 수 있습니다.

취소 명령은 비동기적입니다. 따라서 어떤 경우에는 쿼리를 취소한 후 클라이언트 응용 프로그램에서는 해당 쿼리가 취소된 것으로 나타나지만 데이터 연합 쿼리 엔진에서는 취소가 완료되지 않은 상태일 수 있습니다.

2.11.1 쿼리 취소

1. [쿼리 모니터링](#) 탭을 클릭합니다.
2. 취소할 쿼리를 마우스 오른쪽 단추로 클릭합니다.
3. [취소](#)를 클릭합니다.

2.12 데이터 연합 관리 도구의 쿼리 패널 탭

패널

패널	설명
SQL 텍스트	<p>이 패널에서 SQL 쿼리를 입력할 수 있습니다.</p> <p>요소를 두 번 클릭하거나 카탈로그, 연산자 및 함수 패널에서 끌어 놓는 방식으로 쿼리에 요소를 삽입할 수 있습니다.</p> <p>컨트롤</p> <ul style="list-style-type: none"> • 최대 행: 검색할 최대 행 수입니다. • 전체 행 수 표시: 행을 모두 검색하지 않는 경우에도 결과에 전체 행 수를 표시할지 여부를 지정합니다.
카탈로그	데이터 연합 서비스의 기존 카탈로그를 모두 보여 줍니다.
연산자	가능한 연산자 목록을 보여 줍니다.
함수	범주별로 그룹화된 사용 가능한 함수 목록을 보여 줍니다.

패널	설명
쿼리 결과	쿼리 결과에 대한 컨테이너로, 실행 을 클릭하여 쿼리를 실행할 때 표시됩니다.
원시 데이터	마지막 쿼리 실행의 원시 데이터 결과를 보여 주며, Run 또는 Execute 를 클릭할 때 표시됩니다.
자동 차트	쿼리 결과를 간단한 원형 차트로 보여 주며, 실행(Run) 또는 실행(Execute) 을 클릭할 때 표시됩니다.
쿼리 계획	<p>쿼리를 실행하지 않은 상태의 쿼리 계획을 보여 주며, 쿼리 설명을 클릭할 때 표시됩니다.</p> <p>두 개의 내부 패널이 있습니다.</p> <ul style="list-style-type: none"> • 계획: 계획 구조를 트리 뷰로 보여 줍니다. • 세부 정보: 계획 패널에서 선택한 노드에 대한 세부 정보를 보여 줍니다. <p>속성 뷰에서 선택한 노드에 대한 자세한 내용을 확인할 수 있습니다.</p>
쿼리 통계	통계와 함께 현재 쿼리로 인해 영향을 받는 요소를 보여 주며, 통계 설명 을 클릭할 때 표시됩니다.

단추

단추 레이블	설명
실행	<p>메뉴 항목이 있는 풀다운 단추입니다.</p> <ul style="list-style-type: none"> • 기본 작업: 현재 SQL 텍스트 패널에 있는 쿼리를 실행합니다. • 쿼리 실행 작업: 기본 작업과 동일합니다. • 쿼리 설명 작업: 쿼리 계획을 설명합니다. • 통계 설명 작업: 통계와 함께 영향을 받는 요소를 보여 주며, 카디널리티를 업데이트할 수 있도록 합니다. <p>쿼리 결과 패널에 결과가 표시됩니다.</p>
마지막 변경 실행 취소	SQL 텍스트 패널에서 마지막으로 변경한 내용을 되돌립니다.
마지막 변경 다시 실행	SQL 텍스트 패널에서 마지막으로 변경한 내용을 반복합니다.
카탈로그 새로 고침	카탈로그 패널을 새로 고칩니다.
카탈로그 표시/숨기기	카탈로그 패널을 표시하거나 숨깁니다.
연산자 표시/숨기기	연산자 패널을 표시하거나 숨깁니다.
함수 표시/숨기기	함수 패널을 표시하거나 숨깁니다.
소스 쿼리만 표시	중간 노드를 필터링하여 소스 쿼리 노드만 표시합니다.

상황에 맞는 메뉴

메뉴 항목	설명
계산	<p>하위 메뉴 항목이 있는 풀다운 메뉴입니다.</p> <ul style="list-style-type: none"> 선택 영역만 선택한 노드만 계산합니다. 선택 영역과 하위 선택한 노드와 이 쿼리 컨텍스트에 포함되는 관련 하위를 계산합니다. 계산되지 않은 값만(하위 포함) 현재 카디널리티 열이 확인되지 않은 경우 관련 하위와 함께 선택 영역만 계산합니다.
현재 카디널리티	<p>하위 메뉴 항목이 있는 풀다운 메뉴입니다.</p> <ul style="list-style-type: none"> 사용자 카디널리티 사용: 데이터 연합 서비스에서 선택한 개체에 대한 쿼리 최적화를 위해 사용자가 설정한 카디널리티를 강제로 사용하도록 합니다. 테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다. 이 작업이 완료되면 현재 카디널리티가 사용자 카디널리티와 같아집니다. 소스 카디널리티 사용: 데이터 연합 서비스에서 선택한 개체에 대한 쿼리 최적화를 위해 데이터 소스에서 검색된 카디널리티를 강제로 사용하도록 합니다. 테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다. 이 작업이 완료되면 현재 카디널리티가 소스의 카디널리티와 같아집니다.

관련 링크

[데이터 연합 관리 도구의 쿼리 계획 뷰](#) [페이지 22]

[통계 설명 명령](#) [페이지 23]

2.13 데이터 연합 관리 도구의 쿼리 모니터링 탭

표

열 이름	설명
쿼리	<p>쿼리 또는 하위 쿼리의 ID입니다.</p> <p>다양한 아이콘</p> <ul style="list-style-type: none"> 실행 중 아이콘: 쿼리가 실행 중임을 나타냅니다. 달렸으며 성공함 아이콘: 쿼리가 달리고 성공했음을 나타냅니다. 달렸지만 실패함 아이콘: 쿼리가 달렸지만 실패했음을 나타냅니다. 속성 뷰에서 예외를 확인할 수 있습니다.

열 이름	설명
시작 시간	실행 시작 시간입니다.
종료 시간	실행 종료 시간입니다.
실행 시간	실행 시작 시간에서 종료 시간까지 경과된 시간입니다.
행	쿼리로 추출된 행 수입니다.
상태	쿼리 상태입니다. <ul style="list-style-type: none"> • 분석하는 중: 연합 엔진이 쿼리를 분석하는 중임을 나타냅니다. • 실행하는 중: 연합 엔진이 쿼리를 실행하는 중임을 나타냅니다. • 닫힘: 예외 발생 여부에 관계없이 쿼리가 닫혔음을 나타냅니다.
서버 이름	쿼리를 처리하는 서버의 이름입니다.
사용자 이름	쿼리를 시작한 사용자의 이름입니다.
SQL 텍스트	쿼리의 SQL 텍스트입니다.

필터

필터 레이블	설명
필터	사용 가능한 열의 텍스트에 대한 필터입니다.
상태	쿼리 상태에 대한 필터입니다. <ul style="list-style-type: none"> • 모든 쿼리 • 실행 중인 쿼리 • 실행된 쿼리
유형	쿼리 유형에 대한 필터입니다. <ul style="list-style-type: none"> • 모든 쿼리 • SQL • 명령 • 프로시저
연결	연결에 대한 필터입니다. <ul style="list-style-type: none"> • 모든 연결 • 현재 연결: 현재 데이터 연합 관리 도구 연결의 쿼리만 보여 줍니다.

단추

단추 도구 설명	설명
XML 로 모니터링 정보 저장	XML 로 모니터링 정보를 저장합니다.
새로 고침	서버에서 최신 모니터링 정보를 가져옵니다.

2.14 데이터 연합 관리 도구의 시스템 매개 변수 탭

탭

시스템 매개 변수 탭을 사용하여 시스템 및 세션 매개 변수/속성을 관리할 수 있습니다.

탭 레이블	설명
시스템 매개 변수	<p>시스템 매개 변수를 관리합니다.</p> <p>열</p> <ul style="list-style-type: none"> 매개 변수: 매개 변수의 이름입니다. 현재 값: 현재 매개 변수의 값입니다. 여기서 새 값을 입력할 수 있습니다. 읽기 전용 매개 변수의 경우 배경이 회색으로 표시됩니다. 기본값: 시스템 시작 시 매개 변수의 값입니다. 현재 값을 변경한 경우 변경을 취소하고자 할 때 이 값을 참조로 사용할 수 있습니다. 범주: 매개 변수의 범주입니다. 설명: 매개 변수에 대한 설명입니다. 데이터 연합 관리 도구 가이드에서 설명이 포함된 전체 매개 변수 목록을 확인할 수도 있습니다.
세션 매개 변수	<p>세션 매개 변수를 관리합니다.</p> <p>열</p> <ul style="list-style-type: none"> 매개 변수: 매개 변수의 이름입니다. 현재 값: 현재 매개 변수의 값입니다. 여기서 새 값을 입력할 수 있습니다. 설명: 매개 변수에 대한 설명입니다. 데이터 연합 관리 도구 가이드에서 설명이 포함된 전체 매개 변수 목록을 확인할 수도 있습니다.
시스템 속성	<p>시스템 속성을 보여 줍니다.</p> <p>열</p> <ul style="list-style-type: none"> 매개 변수: 매개 변수의 이름입니다. 현재 값: 현재 매개 변수의 값입니다.
시작 매개 변수	<p>시작 매개 변수를 보여 줍니다.</p> <p>열</p>

탭 레이블	설명
	<ul style="list-style-type: none"> • 매개 변수: 매개 변수의 이름입니다. • 현재 값: 현재 매개 변수의 값입니다.
설치 매개 변수	<p>설치 매개 변수를 보여 줍니다.</p> <p>열</p> <ul style="list-style-type: none"> • 구성 요소: 매개 변수의 구성 요소 이름입니다. • 매개 변수: 매개 변수의 이름입니다. • 현재 값: 현재 매개 변수의 값입니다. • 기본값: 시스템 시작 시 매개 변수의 값입니다. • 원본: 매개 변수 값의 원본입니다. 다음 항목 중 하나입니다. ORIGIN_DEFAULT, ORIGIN_SERVER_PROPERTIES, ORIGIN_SYSTEM_PROPERTIES

표시와 관련된 상황에 맞는 메뉴

메뉴 항목	설명
시스템 및 세션 매개 변수	시스템 및 세션 매개 변수만 보여 줍니다.
모든 매개 변수	모든 탭을 보여 줍니다.

관련 링크

[시스템 매개 변수 목록](#) [페이지 55]

2.15 데이터 연합 관리 도구의 커넥터 구성 탭

패널

패널	설명
커넥터 트리	<p>커넥터 목록을 표시합니다.</p> <p>커넥터 구성을 확인하려면 커넥터 트리에서 해당 커넥터를 두 번 클릭합니다.</p>
일반 정보 탭	<p>현재 커넥터의 일반 정보를 표시합니다.</p> <p>커넥터의 일반 정보를 확인하려면 커넥터 트리에서 해당 커넥터를 두 번 클릭합니다.</p>
기능 탭	<p>현재 커넥터의 기능을 표시합니다.</p> <p>커넥터의 기능을 확인하려면 해당 커넥터를 두 번 클릭하고 기능 탭을 선택합니다.</p>

패널	설명
구성 속성 탭	<p>현재 커넥터의 구성 속성을 표시합니다.</p> <p>커넥터의 구성 속성을 확인하려면 해당 커넥터를 확장하고 구성 노드를 두 번 클릭합니다.</p>

단추

단추 레이블	설명
모두 축소	커넥터 목록을 접습니다.
모두 확장	커넥터 목록을 펼칩니다.
검색 창 표시/숨기기	<p>검색 창을 표시하거나 숨깁니다.</p> <p>이름을 기준으로 커넥터를 검색하려는 경우 이 단추를 사용할 수 있습니다.</p>
새로 고침	커넥터 목록을 새로 고칩니다.
저장	구성 속성을 저장합니다.

상황에 맞는 메뉴

메뉴 항목	설명
구성 만들기	<p>커넥터에 대한 새 구성을 만듭니다.</p> <p>이 메뉴를 통해 구성 속성에 대한 새 값을 설정할 수 있습니다.</p>
구성 편집	<p>커넥터의 구성을 편집할 수 있습니다.</p> <p>아니면 커넥터를 단순히 두 번 클릭해도 됩니다.</p>
구성 삭제	<p>커넥터의 구성을 삭제합니다.</p> <p>구성을 삭제하면 기본값이 사용됩니다.</p>

2.16 데이터 연합 관리 도구의 통계 탭

표

열 이름	설명
카탈로그	<p>다음을 포함할 수 있습니다.</p> <ul style="list-style-type: none"> • 카탈로그 이름 • 스키마 이름 • 테이블 이름 • 열 이름 • 오류 설명 • 대기 메시지 <p>현재 열은 개체가 테이블 또는 열인 경우에만 채워집니다.</p>
마지막 계산 날짜	개체에서 마지막으로 계산 작업이 이루어진 시간 또는 계산 작업이 없는 경우에는 계산 안 함
요청 수	개체에 대해 데이터 연합 서비스에서 실행되는 쿼리 수 또는 쿼리가 없는 경우에는 캐시된 레코드 없음
현재 카디널리티	쿼리 계획을 최적화하기 위해 데이터 연합 서비스에서 현재 사용하는 카디널리티 또는 없는 경우에는 알 수 없음
소스의 카디널리티	개체에 대한 계산 작업이 실행된 후 데이터 소스에서 반환되는 카디널리티 또는 없는 경우에는 알 수 없음
사용자 카디널리티	<p>쿼리 계획을 최적화하기 위해 사용자가 데이터 연합 서비스에서 사용하도록 설정한 카디널리티 또는 없는 경우에는 설정 취소.</p> <p>이 열은 편집 가능합니다. 값을 편집하려면 셀을 클릭하고 정수를 입력한 후 Return 키를 누르거나 다른 곳을 클릭하면 됩니다. 편집한 내용을 삭제하려면 이스케이프를 누릅니다.</p>
- 모든 열	어떤 항목(계산, 새로 고침...)에 대해 현재 작업이 실행 중일 때 그 항목은 기울임꼴로 표시됩니다.

단추

단추 레이블	설명
새로 고침	데이터 연합 서비스에서 현재 표시되는 모든 데이터를 업데이트합니다.

단추 레이블	설명
	많은 개체를 표시할 때 이 작업을 실행하는 데 시간이 많이 걸릴 수 있습니다. 오랜 시간이 걸리는 새로 고침 작업의 경우, 백그라운드에서 실행되는 작업의 진행 상태를 진행률 보기에서 모니터링할 수 있습니다.
계산	<p>선택한 개체가 속한 데이터 소스에서 이 개체의 카디널리티 검색을 데이터 연합 서비스에 요청합니다. 작업 완료 후, 사용자 카디널리티와 마지막 계산 날짜가 업데이트되고 현재 카디널리티가 사용자 카디널리티로 설정됩니다.</p> <p>테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다.</p>

상황에 맞는 메뉴

메뉴 항목	설명
계산	<p>선택한 개체가 속한 데이터 소스에서 이 개체의 카디널리티 검색을 데이터 연합 서비스에 요청합니다. 작업 완료 후, 사용자 카디널리티와 마지막 계산 날짜가 업데이트되고 현재 카디널리티가 사용자 카디널리티로 설정됩니다.</p> <p>테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다.</p>
사용자 카디널리티 사용	데이터 연합 서비스에서 선택한 개체에 대한 쿼리 최적화를 위해 사용자가 설정한 카디널리티를 강제로 사용하도록 합니다. 테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다. 작업 완료 후, 현재 카디널리티는 사용자 카디널리티 와 같습니다.
소스 카디널리티 사용	데이터 연합 서비스에서 선택한 개체에 대한 쿼리 최적화를 위해 데이터 소스에서 검색된 카디널리티를 강제로 사용하도록 합니다. 테이블 또는 열만 선택한 경우 이 작업이 활성화됩니다. 이 작업이 완료되면 현재 카디널리티가 소스의 카디널리티 와 같아집니다.

필터

업	설명
카탈로그	확인한 모든 카탈로그에서 필터를 만듭니다. 확인 을 클릭하여 선택 항목에 대한 유효성을 검사하거나 다른 곳을 눌러 선택 항목을 삭제합니다.
- 다른 모든 필터	<p>이들 필터를 통해 표시된 테이블과 열을 필터링할 수 있습니다.</p> <p>테이블이 필터 조건을 충족시키지 않더라도 테이블의 열 중 하나가 모든 필터 조건을 충족시킨다면 테이블이 표시될 수 있습니다.</p> <p>이런 필터를 이용해 카탈로그나 스키마를 숨길 수 없습니다. 카탈로그와 스키마는 필터 조건을 만족하는 개체가 없더라도 표시됩니다. 전체 카탈로그를 숨기려면 카탈로그 필터를 사용합니다.</p>

관련 링크

2.17 데이터 연합 관리 도구에서 속성 뷰 표시

데이터 연합 관리 도구에서 속성 뷰는 인터페이스의 다양한 요소에 대한 보충 정보를 표시합니다.

▶ 창 ▶ 기타 ▶ 관리 ▶ 속성 ▶을 클릭합니다.

2.18 데이터 연합 관리 도구에서 SSL에 대해 구성된 서버에 연결

보안 강화를 위해 SSL에 대해 구성된 서버에 데이터 연합 관리 도구를 연결해야 하는 경우가 많습니다.

1. <BOE 설치 디렉터리>/win32_x86 디렉터리에 있는 DFAdministrationTool.ini 파일을 편집합니다.
2. 다음 JVM 인수를 추가합니다.

```
-Dbusinessobjects.orb.oci.protocol=ssl  
-DcertDir=C:\SSLCert  
-DtrustedCert=cacert.der  
-DsslCert=servercert.der  
-DsslKey=server.key  
-Dpassphrase=passphrase.txt
```

3 쿼리 최적화

3.1 데이터 연합 쿼리의 성능 조정

데이터 연합 쿼리의 성능을 조정할 수 있습니다. 성능을 조정하기 위해 사용할 전략은 다음과 같습니다.

1. 시스템 매개 변수를 사용하여 메모리 사용을 최적화합니다.
2. 통계를 사용하여 응용 프로그램이 소스 쿼리를 위한 최상의 알고리즘을 선택하도록 합니다.
3. 응용 프로그램에서 semi-join 연산자가 자동으로 활성화되지 않는 경우 매개 변수를 변경하여 semi-join 을 활성화할 수 있는지 여부를 확인합니다.
4. semi-join 이 적합하지 않을 경우 매개 변수를 변경하여 *merge join* 을 활성화할 수 있는지 여부를 확인합니다.
5. 데이터가 기본적으로 비활성화되는 기능을 지원하는 경우 커넥터에서 기능을 활성화합니다.
예를 들어 DB2 가 Null 값의 예측 가능한 순서를 지원하지 않을 때 데이터에 Null 값이 없음을 알고 있는 경우에도 *merge join* 을 사용할 수 있습니다. 이 경우 강제로 *order by* 를 수행하도록 소스의 기능을 설정합니다.

i 노트

데이터 기반에 대한 쿼리를 최적화하기 위해 시스템 매개 변수를 변경한 상태에서 다른 시스템으로 데이터 기반의 수준을 올리려면 시스템 매개 변수의 수준도 올려야 합니다. SAP BusinessObjects Enterprise 용 주기 관리 콘솔에서 이를 수행할 수 있습니다.

관련 링크

[시스템 매개 변수를 사용하여 메모리 사용 최적화](#) [페이지 19]

[시스템 매개 변수를 사용하여 큰 테이블에 조인된 작은 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침](#) [페이지 25]

[시스템 매개 변수를 사용하여 정렬 가능한 데이터가 포함된 큰 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침](#) [페이지 27]

[데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기](#) [페이지 30]

3.2 시스템 매개 변수를 사용하여 메모리 사용 최적화

다음 전략을 통해 응용 프로그램에서 메모리를 사용하는 방법을 최적화할 수 있습니다.

- 응용 프로그램을 실행 중인 JVM(Java Virtual Machine)에서 사용된 메모리 양을 설정합니다. 자세한 내용은 *SAP BusinessObjects Enterprise* 관리자 가이드에서 서버 속성 변경에 대한 부분을 참조하십시오.
응용 프로그램의 속도와 사용 가능한 메모리 양에 따라 기본값을 조정합니다.
- 서버 매개 변수 EXECUTOR_TOTAL_MEMORY 를 설정합니다.
이 매개 변수를 사용하면 쿼리 실행에 사용된 메모리 양을 구성할 수 있습니다.
JVM 에서 사용된 메모리의 백분율 또는 단위를 나타내는 접미사가 있는 고정 값(예: 512M, 512m, 1024K 또는 1024k)으로 이 매개 변수를 설정합니다. 고정 값을 입력하는 경우 이 값은 JVM 에 지정된 값보다 작아야 합니다.
- 서버 매개 변수 EXECUTOR_STATIC_MEMORY 를 설정합니다.
이 매개 변수를 사용하면 초기화 시 연산자에 할당된 최소 메모리 양을 설정할 수 있습니다. 실행기에서 사용된 메모리의 백분율 또는 고정 값을 설정할 수 있습니다. 고정 값을 입력하는 경우 이 값은 JVM 에 지정된 값보다 작아야 합니다.

- 서버 매개 변수 `MAX_CONCURRENT_MEMORY_CONSUMING_QUERIES` 를 설정합니다.
동시에 실행할 수 있는 메모리를 소모하는 쿼리 수를 정의합니다. 다른 쿼리는 영향을 받지 않습니다.
대형 쿼리가 많이 있는 경우 여기에 작은 값을 입력합니다.
소형 쿼리가 많이 있는 경우 큰 번호를 입력합니다.
- `MAX_CONCURRENT_MEMORY_CONSUMING_OPERATORS`
이 매개 변수는 병렬 실행되는 메모리를 소모하는 연산자 수를 제한합니다.
쿼리에서 연산자에 의해 소모되는 메모리 양이 너무 많으면 이 숫자를 줄입니다.
연결된 상태에서 데이터 소스가 서로 다른 대형 테이블 수를 계산하면 쿼리에서 연산자의 평균 크기 및 개수를 예상할 수 있습니다. 예를 들어, 한 매핑 규칙에 데이터 소스가 서로 다른 네 개의 대형 테이블이 있는 경우 결과적으로 세 개의 조인이 메모리를 소모합니다.

예를 들어, JVM 에 대해 1000MB 의 메모리를 할당하려면 JVM 메모리를 1000M 으로 설정합니다.

그런 다음 쿼리 실행을 위해 800MB 의 메모리를 할당하려면 `EXECUTOR_TOTAL_MEMORY` 를 80%로 설정합니다.

각 연산자에 대해 200MB 의 메모리를 할당하려면 `EXECUTOR_STATIC_MEMORY` 를 25%로 설정하고,

동시 연산자를 두 개로 제한하려면 `MAX_CONCURRENT_MEMORY_CONSUMING_QUERIES` 를 2 로 설정합니다.

위의 설정 예제에서 두 개의 쿼리를 동시에 실행할 수 있는데, 각각 100MB 의 최소 메모리를 보유하고 있으며 600MB 의 동적 메모리 풀에 액세스할 수도 있습니다.

시스템의 메모리 사용을 감사하려면 `info buffermanager` 문을 사용합니다.

i 노트

데이터 기반에 대한 쿼리를 최적화하기 위해 시스템 매개 변수를 변경한 상태에서 다른 시스템으로 데이터 기반의 수준을 올리려면 시스템 매개 변수의 수준도 올려야 합니다. SAP BusinessObjects Enterprise 용 주기 관리 콘솔에서 이를 수행할 수 있습니다.

관련 링크

[데이터 연합 관리 도구를 사용하여 시스템 매개 변수 변경](#) [페이지 54]

[메모리를 사용하는 연산자](#) [페이지 20]

[데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기](#) [페이지 30]

3.2.1 메모리를 사용하는 연산자

다음은 쿼리에서 연산자를 사용할 때 데이터 연합 서비스로 인해 메모리가 사용되는 연산자입니다.

- join
- cartesian product
- orderby
- groupby
- groupby - 그룹에 다른 값이 많이 있는 경우(큰 그룹 집합)

데이터 연합 쿼리 엔진에서는 테이블, 프로젝션, 필터, 함수 평가 검사를 수행하거나 연산을 소스에 푸시할 때 메모리가 많이 사용되지 않습니다.

3.3 통계를 사용하여 응용 프로그램에서 소스 쿼리에 최적인 알고리즘 선택

통계는 데이터 연합 쿼리 엔진에서 쿼리를 최적화하기 위해 내부에서 사용됩니다.

통계는 계속해서 새로 고쳐지는 것이 아닙니다. 시스템이 프로덕션에 배포될 때까지 기다린 다음 특정 샘플 시간에 통계를 실행해야 합니다. 그러면 통계가 수집되고 후속 쿼리 계획이 생성됩니다.

통계 하위 시스템은 실제로 다음과 같은 두 부분으로 구성됩니다.

- 데이터 소스 수준에서 확인된 측정값의 카디널리티를 계산하는 도구
- 쿼리가 실행될 때 테이블 또는 특성이 요청되는 횟수를 계산하는 레코더

카디널리티를 수동 값으로 재정의하여 쿼리 계획을 최적화할 때 카디널리티 사용에 영향을 줄 수 있습니다.

관련 링크

[열 카디널리티 정보](#) [페이지 21]

[보고서 최적화에 필요한 통계만 계산하도록 기록된 통계 필터링](#) [페이지 22]

3.3.1 열 카디널리티 정보

카디널리티는 열의 행 수입니다.

다른 요소에 대해서도 카디널리티를 측정할 수 있습니다. 즉, 테이블, 테이블이 포함된 스키마 또는 전체 카탈로그에 대한 카디널리티를 측정할 수 있습니다. 각각의 경우 개체의 카디널리티를 측정하면 포함된 모든 개체의 카디널리티를 간편하게 파악할 수 있습니다. 예를 들어, 스키마의 카디널리티가 1000 일 경우 스키마의 대부분의 테이블에 포함된 거의 모든 열의 행 수는 1000 개입니다.

데이터 연합을 사용하는 경우 시스템에서는 데이터 소스에 있는 열의 카디널리티를 정확히 인식할수록 쿼리를 보다 효율적으로 최적화할 수 있습니다. 이에 따라 데이터 연합 쿼리 엔진이 데이터 소스의 카디널리티를 예측할 수 있으며, 보다 효율적이라고 판단되는 경우 사용자가 직접 카디널리티를 설정할 수도 있습니다.

카디널리티 예측 및 설정은 통계 설정이라는 최적화 작업의 일부입니다.

관련 링크

[통계를 사용하여 응용 프로그램에서 소스 쿼리에 최적인 알고리즘 선택](#) [페이지 21]

3.3.2 열 간의 관계를 나타내는 팬아웃 값 정보

팬아웃 값 예측 및 설정은 통계 설정이라는 최적화 작업의 일부입니다.

팬아웃은 두 열의 데이터 간 연결을 측정합니다. 두 개의 열이 있고 첫 번째 열의 각각의 고유 값에 대한 열인 경우 팬아웃은 두 번째 열의 평균 열 수입니다. 예를 들어, 국가가 나열되는 열과 도시가 나열되는 열이 있을 경우 팬아웃을 통해 각 국가의 평균 도시 수를 측정할 수 있습니다.

데이터 연합을 사용하는 경우 쿼리 엔진에서는 데이터 소스에 있는 열의 팬아웃을 정확히 인식할수록 쿼리를 보다 효율적으로 최적화할 수 있습니다. 이에 따라 데이터 연합 쿼리 엔진을 통해 소스에 있는 열의 팬아웃을 설정할 수 있습니다.

관련 링크

3.3.3 보고서 최적화에 필요한 통계만 계산하도록 기록된 통계 필터링

모든 데이터 소스에 대한 통계를 한 번에 계산할 수 있지만 이 작업은 시간이 오래 걸릴 수 있습니다. 다음 절차에서는 이 프로세스의 속도를 향상시키기 위해 쿼리에 필요한 통계만 계산하는 방법을 보여 줍니다.

이 절차는 SAP BusinessObjects Interactive Analysis 문서를 새로 고쳐 생성된 통계를 가져오는 예를 기반으로 하지만 다른 상황에도 적용할 수 있습니다.

카디널리티 계산은 언제든지 수행할 수 있으며 활성화 작업도 필요 없습니다.

1. SAP BusinessObjects Interactive Analysis 의 **쿼리 편집** 패널에서 보고서를 엽니다.
2. **쿼리 패널**에서 SQL 텍스트 영역을 열고 쿼리의 SQL 을 복사한 후 텍스트 영역을 닫습니다.
3. 데이터 연합 관리 도구에서 SQL 을 **쿼리 패널** 탭의 텍스트 영역에 붙여 넣습니다.
4. **실행**을 클릭합니다.
5. **통계** 탭에서 **서버에서 통계 새로 고침** 단추를 클릭합니다.
쿼리 최적화에 사용된 테이블과 열이 **요청 수** 열에 기록됩니다.
6. **통계** 탭에서 다음 사항을 확인합니다.
 - **요청 수** 열의 필터 값이 **기록됨**으로 설정되어 있는지 확인합니다.
7. Ctrl 키를 누른 상태로 **요청 수** 열에서 값이 있는 행을 모두 선택한 다음 **계산** 단추를 클릭합니다.
데이터 연합 관리 도구가 쿼리에 유용한 통계만 계산합니다.
8. SAP BusinessObjects Interactive Analysis 보고서에서 쿼리를 새로 고쳐 실제 쿼리를 실행합니다.
그러면 데이터 연합 쿼리 엔진이 수집된 통계를 사용하여 최적의 계획을 생성합니다.

관련 링크

[데이터 연합 관리 도구의 통계 탭](#) [페이지 16]

3.4 쿼리 계획 최적화

3.4.1 데이터 연합 관리 도구의 쿼리 계획 뷰

정의

쿼리 설명을 클릭하면 **쿼리 계획** 뷰에 쿼리 최적화의 결과가 표시됩니다. **쿼리 계획** 뷰에는 다음 세 가지 창이 있습니다.

- **계획 창**: 쿼리 계획을 트리 구조에 표시
- **세부 정보 창**: **계획** 창에 강조 표시된 항목의 세부 정보 표시
- **속성 창**: **계획** 창과 **세부 정보** 창에 강조 표시된 항목의 속성을 표시합니다.

계획 창에는 커넥터로 보낸 커넥터 쿼리를 나타내는 리프가 있는 트리 구조로 된 쿼리 계획이 표시됩니다. 중간 노드는 Projection, Order By, Group By, Aggregation, Union, Full Outer Join, Calculation(필터, 조인) 등입니다.

이 문서에서는 쿼리 및 커넥터 쿼리에 대한 일반적인 정보만 설명합니다. 이것은 기본적으로 사용자에게 표시되는 정보 (중간 노드 없음)입니다.

1. 쿼리에 대한 전체 정보:

a. **속성 창:**

- a. 사용된 메모리: 쿼리에 필요한 추정 메모리입니다.
- b. 연산자를 사용하는 동시 메모리 수: 쿼리 계획에서 동시에 수행되는 메모리를 사용하는 최대 연산자 수

b. **세부 정보 창:**

a. **통계**

- a. **테이블 카디널리티:** 이 쿼리에서 반환되는 예상 행 수

2. 커넥터 쿼리의 정보:

a. **속성 창:**

- a. **id:** 커넥터 쿼리의 ID
- b. 데이터 연합 SQL: 다중 커넥터 쿼리 엔진에서 사용하는 SQL 구문에 나타나는 커넥터 쿼리
- c. 고유 커넥터 쿼리: 커넥터에서 지원하는 고유 구문에 나타나는 커넥터 쿼리
- d. 커넥터 이름: 커넥터 이름

b. **세부 정보 창:**

- a. **스키마:** 커넥터 쿼리의 예상 열 목록
- b. **키:** 파생 키(테이블의 키에서 유추된 키)
- c. **통계:** 옵티마이저와 각각의 예상 값에서 사용되는 통계
 - a. **테이블 카디널리티**
 - b. **열 카디널리티**
- d. **기능:** 커넥터가 수행할 수 있는 작업 목록
- e. **semi-join:** semi-join 목록
 - a. **필터링된 열:** semi-join 에 사용되는 열 목록
 - a. **종속 열:** (필터링된) 이 열을 필터링하는 데 사용되는 열
 - b. **종속 소스 쿼리:** semi-join 에 대한 값을 제공하는 커넥터 쿼리 목록
 - c. **전략:** semi-join 연산자의 실행 전략을 원하는 순서대로 표시한 목록
 - d. **감소 인수:** semi-join 을 포함하지 않고 반환된 행 수와 semi-join 을 포함하여 반환된 행 수 간의 비율
 - f. **Data Federator SQL:** 쿼리 엔진에서 사용하는 SQL 구문에 나타나는 커넥터 쿼리
 - g. **고유 커넥터 쿼리:** 커넥터에서 지원하는 고유 구문에 나타나는 커넥터 쿼리

3.4.2 통계 설명 명령

설명

통계 설명 명령을 실행하면 쿼리 엔진에서 SQL 쿼리를 최적화하는 데 필요한 모든 통계가 나열됩니다. 쿼리에 대한 명령을 실행하면 트리와 같은 구조가 반환됩니다. 이 뷰에서는 각 소스에 대해 쿼리에 사용되는 테이블, 필요한 통계 및 통계의 업데이트 여부를 볼 수 있습니다. 이 뷰에서 할 수 있는 작업은 다음과 같습니다.

1. 한 번의 클릭으로 쿼리에서 필요한 모든 통계를 새로 고칠 수 있습니다.
2. 특정 테이블이나 열의 통계를 새로 고칠 수 있습니다.
3. 특정 테이블이나 열의 통계를 설정할 수 있습니다.
4. 최상의 계획을 생성하기 위해 필요한 모든 통계를 사용할 수 있는지 확인할 수 있습니다.
5. 사용되는 통계 보기: 소스에서 보거나 사용자가 설정한 통계를 볼 수 있습니다.

명령의 결과는 6 개의 열로 표시됩니다.

- **카탈로그:** 사용자가 소스와 테이블/열을 찾아볼 수 있는 트리 뷰입니다.
- **마지막 계산 날짜:** 소스에서 통계를 마지막으로 계산한 시간입니다.
- **요청 수:** 시스템에서 열의 고유 값(테이블의 카디널리티)을 요청한 횟수(이 쿼리에 대한 것만은 아님)입니다.
- **현재 카디널리티:** 소스의 카디널리티와 사용자(관리자)의 카디널리티라는 두 가지 유형의 카디널리티가 가능합니다. 어떤 정책이 사용되느냐에 따라, 적절한 카디널리티가 현재 카디널리티로 표시됩니다.
- **소스의 카디널리티:** 데이터 소스의 카디널리티입니다.
- **사용자 카디널리티:** 사용자가 특정 테이블 또는 열에 대해 다른 통계를 설정하는 경우 여기에 표시됩니다.

3.4.3 쿼리 설명 기능을 사용해 피드백을 얻어 쿼리 조정

쿼리 설명 기능을 피드백으로 사용하여 쿼리를 조정할 수 있습니다. 다음 쿼리는 두 개의 다른 데이터 소스에서 두 테이블 간에 조인을 수행합니다. <T1>은 데이터 소스 <S1>에서 가져온 작은 테이블이고, <T2>는 데이터 소스 <S2>에서 가져온 큰 테이블입니다.

1. 쿼리 패널에서 **Select * From <T1>, <T2> where <T1>**을 입력합니다. <C1> = <T2>.<C2>
2. 쿼리 설명을 클릭합니다.
3. 계획 패널에서 소스 쿼리 <S1> [<T1>], <S2> [<T2>]를 클릭합니다.

세부 정보 패널에 자세한 정보가 표시됩니다. 세부 정보를 보면 <S1> 및 <S2>에 대한 소스 쿼리 모두가 전체 테이블 스캔임을 알 수 있습니다. 그러나 <T1>이 작은 테이블임을 알기 때문에 <S2>에서 semi-join 이 생성될 것으로 예상합니다. semi-join 이 생성되지 않는 이유를 조사하려면 두 소스 쿼리의 통계를 살펴보면 됩니다. 옵티마이저가 다음을 사용하려고 함을 알 수 있습니다.

- <T1>의 카디널리티
- <T1>의 카디널리티<C1>
- <T2>의 카디널리티
- <T2>의 카디널리티<C2>

하지만 이러한 통계가 모두 알 수 없음으로 표시됩니다.

4. 통계 설명을 클릭합니다.
쿼리 통계 탭이 표시됩니다.

쿼리 통계 탭에서 통계를 다음 값으로 설정할 수 있습니다.

- 카디널리티(<T1>)=25
- 카디널리티(<T1>.<C1>)=25
- 카디널리티(<T2>)=100000
- 카디널리티(<T2>.<C2>)=100000

5. 쿼리 설명을 다시 클릭합니다.

다른 계획을 가져옵니다. <S2>에 대한 semi-join 이 생성됩니다.

관련 링크

[데이터 연합 관리 도구의 쿼리 계획 뷰 \[페이지 22\]](#)

[시스템 매개 변수를 사용하여 큰 테이블에 조인된 작은 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침 \[페이지 25\]](#)

3.4.4 데이터 연합 관리 도구를 사용하여 연산자가 푸시되고 있는지 여부 확인

쿼리는 일반적으로 데이터 연합 쿼리 엔진 대신 데이터베이스 시스템을 사용하여 연산자를 평가할 때 보다 효율적입니다.

데이터 연합 관리 도구의 [쿼리 모니터링](#) 탭에서 연산자가 푸시되고 있는지 여부를 확인할 수 있습니다.

1. 데이터 연합 관리 도구에서 [쿼리 모니터링](#) 탭을 엽니다.
2. [새로 고침](#) 단추를 클릭하여 가장 최근의 쿼리를 확인합니다.
3. 쿼리를 찾고 하위 쿼리를 검색하여 연산자가 푸시되고 있는지 여부를 확인합니다.
 - 연산자가 하위 쿼리에 나열되면 데이터 소스로 푸시되고 있는 것입니다.
 - 연산자가 상위 쿼리에만 나열되면 푸시되고 있지 않습니다.
데이터 연합 쿼리 엔진에서 연산자를 데이터 소스에 강제로 푸시하도록 하려는 경우 커넥터가 연산자를 허용하도록 데이터 소스에 대한 커넥터 기능을 설정할 수 있습니다.

관련 링크

[데이터 연합 쿼리의 성능 조정 \[페이지 19\]](#)

[데이터 연합 관리 도구의 쿼리 모니터링 탭 \[페이지 11\]](#)

[데이터 연합 관리 도구를 사용하여 관계형 커넥터 및 SAS 커넥터 기능 설정 \[페이지 51\]](#)

3.4.5 시스템 매개 변수를 사용하여 큰 테이블에 조인된 작은 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침

쿼리를 최적화하는 동안 데이터 연합 옵티마이저는 데이터 소스에서 쿼리 엔진으로의 데이터 전송을 줄이려고 합니다. 이를 달성하는 방법 중 하나는 데이터 소스의 큰 테이블에 액세스하는 동안 semi-join 을 생성하는 것입니다. 옵티마이저는 성능 향상이 있을 것으로 예측되는 경우에만 semi-join 을 생성합니다.

semi-join 생성 및 실행은 다음과 같은 시스템 매개 변수와 커넥터 속성에 따라 제어됩니다.

- `ACTIVATE_SEMI_JOIN_RULE`
semi-join 생성 규칙의 활성화 여부입니다. 옵티마이저는 이 매개 변수가 true 로 설정된 경우에만 semi-join 을 생성합니다.
- `MIN_SOURCE_CARDINALITY_THRESHOLD_FOR_SEMI_JOIN_RULE`
옵티마이저가 semi-join 을 생성하는 최소 소스 쿼리 카디널리티입니다. 옵티마이저는 많은 양의 데이터를 반환하는 소스 쿼리에 대해서만 semi-join 을 생성합니다. 소스 쿼리의 예상 카디널리티가 이 매개 변수보다 작을 경우 옵티마이저는 이 소스 쿼리에 대해 semi-join 을 생성하지 않습니다.
- `MIN_ACTIVATION_THRESHOLD_FOR_SEMI_JOIN_RULE`
semi-join 의 용도는 데이터 소스에서 쿼리 엔진으로의 데이터 전송을 줄이는 것입니다. 이 매개 변수는 옵티마이저가 semi-join 을 생성하는 최소 데이터 전송 감소입니다. (semi-join 을 사용하지 않을 경우 행 수 / semi-join 을 사용

할 경우 행 수)로 계산되는 비율을 감소 인수라고 합니다. 감소가 이 매개 변수보다 클 경우 semi-join 이 생성되며, 그렇지 않을 경우 semi-join 이 생성되지 않습니다.

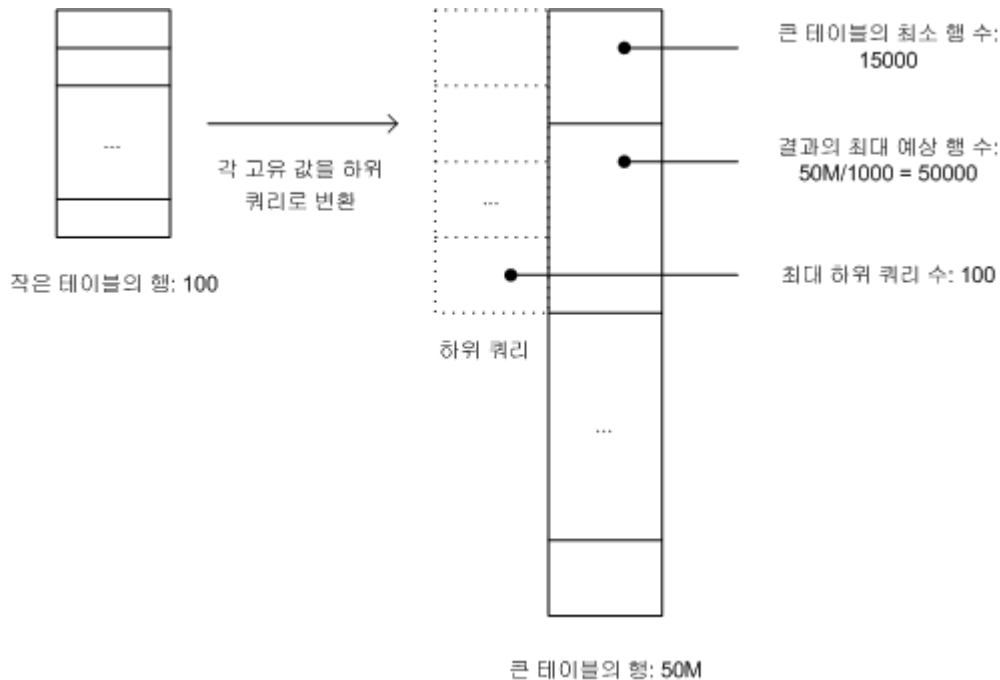


그림 1: 데이터 연합 쿼리 엔진이 매개 변수 `ACTIVATE_SEMI_JOIN_RULE=true`, `MIN_SOURCE_CARDINALITY_THRESHOLD_FOR_SEMI_JOIN_RULE=15000` 및 `MIN_ACTIVATION_THRESHOLD_FOR_SEMI_JOIN_RULE=1000` 을 사용하여 *semi-join* 을 활성화하기로 결정하는 방식

i 노트

데이터 기반에 대한 쿼리를 최적화하기 위해 시스템 매개 변수를 변경한 상태에서 다른 시스템으로 데이터 기반의 수준을 올리려면 시스템 매개 변수의 수준도 올려야 합니다. SAP BusinessObjects Enterprise 용 주기 관리 콘솔에서 이를 수행할 수 있습니다.

예

작은 테이블과 매우 큰 테이블에 대해 실행되는 쿼리에서 *semi-join* 활성화

이 예에서는 100 개의 행이 포함된 작은 테이블과 50000000 개의 행이 포함된 큰 테이블이 있을 경우 *semi-join* 을 활성화하도록 시스템 및 세션 매개 변수를 설정하는 방법을 보여 줍니다. 또한 작은 테이블의 값을 사용하여 큰 테이블의 값을 필터링할 경우 10000 개의 행이 반환된다고 가정합니다.

데이터 연합 프로젝트를 배포한 후에는 통계를 새로 고칩니다. 데이터 연합 관리 도구에서 통계를 새로 고칠 수 있습니다.

`MIN_SOURCE_CARDINALITY_THRESHOLD_FOR_SEMI_JOIN_RULE` 을 15000 으로 설정합니다. 큰 테이블의 행 수가 15000 개를 초과하므로 이 값에 따라 쿼리 엔진에서 *semi-join* 을 사용할 수 있게 됩니다.

`MIN_ACTIVATION_THRESHOLD_FOR_SEMI_JOIN_RULE` 을 1000 으로 설정합니다. 이 값이 권장되는 기본값이며, 다음과 같이 사용됩니다.

큰 테이블의 행 수를 이 수로 나눠 임계값을 계산합니다. 이 경우 임계값은 50000($50M / 1000 = 50000$)입니다. 그러면 쿼리 엔진에서 통계를 확인하는데, 여기서는 *semi-join* 에서 약 10000 개의 행을 반환할 것임을 보여 줍니다. 이 수는 임계값인 50000 보다 적으므로 데이터 연합 응용 프로그램에서 *semi-join* 을 사용할 수 있게 됩니다.

이 값을 너무 낮게 설정하면 유용하지 않은 경우에도 쿼리 엔진에서 *semi-join* 을 사용하게 됩니다. 예를 들어, 이 값을 1 로 설정하면 쿼리 엔진은 *semi-join* 에서 반환하는 행 수가 50000000($50000000 / 1 = 50000000$)개인 경우에도 *semi-join* 을 사용합니다. 이는 전체 테이블 검사를 수행하는 것과 동일합니다.

이 값을 2 로 설정하면 쿼리 엔진은 *semi-join* 에서 반환하는 행 수가 테이블 검사에서 반환하는 행 수의 절반인 경우 *semi-join* 을 사용합니다. 이 기법도 전체 테이블 검사에 비해 크게 유용하지 않습니다.

이 값을 너무 높게 설정하면 쿼리 엔진은 유용한 경우에도 *semi-join* 을 사용하지 않습니다. 예를 들어, 이 값을 500000000 으로 설정하면 쿼리 엔진은 *semi-join* 에서 반환하는 행 수가 1($500000000 / 500000000 = 1$)개인 경우에만 *semi-join* 을 사용합니다.

이 값을 1000 으로 설정하는 것은 일반적으로 결과가 테이블 검사보다 1000 배 작을 경우 *semi-join* 을 활성화하도록 요청하는 것과 동일합니다.

이러한 설정을 통해 쿼리 엔진은 *semi-join* 을 수행하여 속도 및 메모리 사용이 최적화된 쿼리를 실행할 수 있어야 합니다.

관련 링크

[데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기](#) [페이지 30]

3.4.6 시스템 매개 변수를 사용하여 정렬 가능한 데이터가 포함된 큰 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침

쿼리에서 큰 테이블이 반환되고 해당 테이블의 데이터를 정렬할 수 있을 경우 응용 프로그램은 순서 기반 연산자를 사용하여 작업 속도를 향상시킬 수 있습니다. 순서 기반 연산자로는 *merge join* 과 순서 기반 *group by* 가 있습니다.

merge join 은 소스에서 *order by* 연산자를 푸시한 다음 순서가 지정된 결과를 사용하여 조인을 즉시 수행합니다.

이 기법에서는 조인되어야 할 결과를 정렬하지 않습니다. 따라서 순서가 지정되지 않은 결과에 조인을 적용하는 것보다 빠릅니다.

어떤 경우에 *merge join* 이 유용한지 확인

위와 같은 이점이 있기도 하지만 *merge join* 은 다음과 같은 조건이 모두 충족되는 경우에만 유용합니다.

- *semi-join* 이 불가능한 경우
- 쿼리가 조인할 큰 테이블을 반환하는 경우
- 데이터 소스가 *order by* 연산자를 지원하거나 데이터가 *order by* 연산자 사용에 적합한 경우
 - 데이터 소스가 *order by* 연산자를 지원하는지 여부를 파악하려면 데이터 소스의 기능을 확인하면 됩니다. 예를 들어, DB2 는 예측 가능한 Null 값 순서 지정을 지원하지 않습니다.
 - 데이터 정렬 설정을 예측할 수 없어 *order by* 가 지원되지 않는 소스도 있습니다. 예를 들어, DB2 가 예측 가능한 Null 값 순서 지정을 지원하지 않는 상태에서 데이터에 Null 이 없는 것이 확실하면 *merge join* 을 계속 사용할 수 있습니다. 이 경우 강제로 *order by* 를 수행하도록 소스의 기능을 설정합니다.

어떤 경우에 *merge join* 매개 변수를 변경해야 할지 확인

Merge join 은 기본적으로 큰 테이블에 대해 활성화되어 있습니다. 시스템 매개 변수를 사용하여 *merge join* 활성화를 제어할 수 있습니다.

다음과 같은 특정 조건에 따라 매개 변수를 설정해야 할 수도 있습니다.

- 큰 테이블이 있지만 큰 행에서 크기가 분산된 경우 쿼리가 필수 최소 행 수보다 적은 수의 행을 반환하는 경우
- 작은 테이블이 있지만 *merge join* 을 사용하고자 하는 경우

merge join 의 활성화 여부 확인

merge join 이 활성 상태인지 확인하려면 데이터 연합 관리 도구를 사용하여 쿼리 기록을 살펴보고 하위 쿼리에 *order by* 연산자가 포함되어 있는지 확인합니다.

관련 링크

[시스템 매개 변수를 사용하여 순서 기준 연산자의 활성화 제어](#) [페이지 28]

[시스템 매개 변수를 사용하여 큰 테이블에 조인된 작은 테이블에 대해 실행할 쿼리를 최적화하기 위한 지침](#) [페이지 25]

[데이터 연합 관리 도구를 사용하여 관계형 커넥터 및 SAS 커넥터 기능 설정](#) [페이지 51]

3.4.7 시스템 매개 변수를 사용하여 순서 기준 연산자의 활성화 제어

다음 매개 변수를 사용하여 순서 기준 연산자를 트리거할 수 있습니다.

- 옵티마이저 규칙을 활성화하여 유용한 순서 기준 연산자를 검색하려면 서버 매개 변수 `ACTIVATE_ORDER_BASED_OPTIMIZATION_RULE` 을 True 로 설정합니다.
- 서버 매개 변수 `MIN_STORE_CARDINALITY_THRESHOLD_FOR_ORDER_BASED_JOIN_RULE` 및 `MIN_TRANSFER_CARDINALITY_THRESHOLD_FOR_MERGE_JOIN_RULE` 을 설정합니다. 이러한 숫자는 병합 조인 연산자를 선택할 입력 피연산자의 최소 카디널리티(행 수)를 정의합니다. 병합 조인은 한 피연산자에 `minStoreCardForMergeJoin` 에 대한 카디널리티가 있고 다른 피연산자에 `minTransferCardForMergeJoin` 에 대한 카디널리티가 있는 경우에만 선택할 수 있습니다.
- 서버 매개 변수 `MIN_CARDINALITY_THRESHOLD_FOR_GROUP_BY_TRANSFORMATION_RULE` 을 설정합니다. 이 숫자는 `orderBasedGroupBy` 연산자를 선택할 입력 피연산자의 최소 카디널리티를 정의합니다.

i 노트

데이터 기반에 대한 쿼리를 최적화하기 위해 시스템 매개 변수를 변경한 상태에서 다른 시스템으로 데이터 기반의 수준을 올리려면 시스템 매개 변수의 수준도 올려야 합니다. SAP BusinessObjects Enterprise 용 주기 관리 콘솔에서 이를 수행할 수 있습니다.

관련 링크

[데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기](#) [페이지 30]

3.4.8 강제로 데이터 소스 하위 쿼리 병렬 실행

기본적으로 데이터 연합 응용 프로그램에서는 하위 쿼리의 결과를 사용할 준비가 된 경우에만 데이터 소스에 하위 쿼리의 실행을 제출합니다. 이를 통해 데이터 연합 응용 프로그램은 기본 데이터베이스가 쿼리 결과를 캐시해야 할 시간을 단축시키고 작업이 오래 걸릴 경우 시간 초과로 인한 영향이 없도록 합니다.

하지만 강제로 데이터 소스 쿼리를 조기에 실행할 수 있습니다.

union 연산자의 피연산자인 데이터 소스 하위 쿼리의 병렬 실행을 활성화하려면 서버 매개 변수 `ACTIVATE_MULTI_THREADED_UNION_OPERATOR` 를 true 로 설정합니다.

3.4.9 semi-join 실행 전략

소개

데이터 연합 서비스가 *semi-join* 연산자를 적용하여 작은 테이블과 큰 테이블 간의 조인을 최적화하는 경우 다음 전략 중 하나를 사용하여 용량이 큰 테이블의 행 수를 줄일 수 있습니다.

이러한 각각의 전략은 상대적으로 작은 값 목록을 만들어 이 목록에 대해 큰 테이블의 행을 조인합니다. 실행 전략은 이 목록을 만들기 위한 기술적 수단입니다. 모든 데이터 소스가 동일한 기술을 지원하는 것은 아닙니다.

`SEMI_JOIN_EXECUTION_STRATEGIES` 매개 변수를 사용하여 이러한 전략을 활성화 또는 비활성화 하거나 전략의 우선 순위를 변경할 수 있습니다.

전략	설명
IN	데이터 연합 쿼리 엔진은 IN 키워드로 값 목록을 구성합니다.
임시 테이블	데이터 연합 쿼리 엔진은 데이터의 소스에 임시 테이블을 만들어 값의 목록을 구성합니다.
준비된 명령문	데이터 연합 쿼리 엔진은 SQL 준비된 명령문을 사용하여 값 목록을 구성하며, 목록의 각 값은 준비된 명령문에 매개 변수로 전달됩니다.

3.5 특정 커넥터 최적화

3.5.1 SAP NetWeaver BW 에 대한 병렬 쿼리의 콜백 동시성 증가

`jcoServerProperties` 라는 리소스 속성을 사용하여 쿼리 서버가 SAP NetWeaver BW 의 콜백에 대비하는 스레드 수를 증가시킬 수 있습니다.

1. 데이터 연합 관리 도구를 열고 관리자 권한이 있는 사용자 계정을 사용하여 로그인합니다.
2. [커넥터 구성](#) 탭에서 SAP NetWeaver BW 커넥터를 편집합니다.
3. `jcoServerProperties` 라는 커넥터 속성을 `jco.server.connection_count=10` 값으로 설정합니다.

이 속성의 기본값은 2 입니다. 시스템 매개 변수 MAX_CONCURRENT_MEMORY_CONSUMING_QUERIES 가 10 을 초과하는 경우를 제외하고 권장되는 최대값은 10 입니다. 이 경우 스레드 수는 부족하지 않도록 시스템 매개 변수 값보다 커야 합니다.

노트

커넥터 속성은 jcoServerProperties 입니다. 이 속성의 값을 전체 문자열 jco.server.connection_count=10 으로 설정해야 합니다.

관련 링크

[SAP NetWeaver BW 데이터 소스의 커넥터 속성 목록](#) [페이지 44]

3.5.2 SAP NetWeaver BW 에 대한 쿼리에서 응답 패키지 크기 변경

packageSize 라는 리소스 속성을 사용하여 SAP NetWeaver BW 의 쿼리 응답에서 반환되는 데이터의 패키지 크기를 변경할 수 있습니다. 패키지 크기는 패키지당 행 수로 측정됩니다.

패키지 크기가 증가하면 속도가 빨라지지만 메모리 사용은 더 늘어납니다.

반대로, 패키지 크기가 감소하면 속도는 느려지지만 메모리 사용은 줄어듭니다.

1. 데이터 연합 관리 도구를 열고 관리자 권한이 있는 사용자 계정을 사용하여 로그인합니다.
2. **커넥터 구성** 탭을 오른쪽 단추로 클릭하여 SAP NetWeaver BW 커넥터를 편집합니다.
3. packageSize 속성을 편집하고 패키지당 원하는 행 수를 이 속성의 값으로 입력합니다.

자세한 내용은 SAP NetWeaver BW 커넥터 속성 목록에서 packageSize 속성에 대한 설명을 참조하십시오.

관련 링크

[SAP NetWeaver BW 데이터 소스의 커넥터 속성 목록](#) [페이지 44]

3.6 데이터 연합 서비스에 대해 작성한 최적화 설정 수준 올리기

데이터 기반을 하나의 시스템에서 다른 시스템으로(예: 개발 시스템에서 테스트 시스템으로) 마이그레이션하는 작업을 수준 올리기라고 합니다.

사용자가 최적화를 위해 데이터 연합 서비스의 시스템 매개 변수를 변경한 경우, 데이터 기반의 수준을 올리는 동안 시스템 매개 변수의 수준도 올려야 합니다.

SAP BusinessObjects Enterprise 용 주기 관리 콘솔에서도 이를 수행할 수 있습니다.

1. SAP BusinessObjects Enterprise 용 주기 관리 콘솔을 엽니다.
2. **폴더 및 개체** 폴더를 확장한 후 **데이터 연합**을 클릭합니다.
3. **매개 변수** 개체의 수준을 올립니다.

자세한 내용은 Lifecycle Management for SAP BusinessObjects *User Guide* 를 참조하십시오.

4 데이터 소스로 커넥터 구성

4.1 데이터 연합 관리 도구에서 커넥터 정보 보기

1. 데이터 연합 관리 도구를 시작합니다.
2. 커넥터 구성 탭을 클릭합니다.
3. 트리 목록에서 커넥터를 두 번 클릭합니다.
4. 설정을 보려면 일반 정보를 클릭하고, 커넥터 기능을 보려면 기능을 클릭합니다.

4.2 데이터 연합 관리 도구에서 커넥터 속성 변경

1. 데이터 연합 관리 도구를 시작합니다.
2. 커넥터 구성 탭을 클릭합니다.
3. 트리 목록에서 커넥터를 마우스 오른쪽 단추로 클릭한 다음 구성 만들기를 클릭합니다.
4. 나타나는 구성 노드를 두 번 클릭합니다.
5. 구성 속성 탭에서 속성을 두 번 클릭하여 편집하고 값을 변경한 다음 데이터 저장 아이콘을 클릭하여 변경 내용을 저장합니다.

4.3 관계형 데이터 소스에 대한 커넥터 구성

4.3.1 관계형 데이터 소스의 공통 커넥터 속성 목록

아래 표에는 관계형 데이터 소스에 대해 구성할 수 있는 공통 속성이 나열되어 있습니다.

속성	설명
capabilities	데이터베이스에서 지원되는 모든 기능의 목록입니다. 요소는 ; 문자로 구분됩니다(요소 간에 공백 없음). 예 <code>capabilities=fullSQL\=true;outerjoin\=false;rightouterjoin\=true</code>
compCollationCompatible	True/예 또는 False/아니요 데이터 소스의 비교 작업을 위한 데이터 정렬이 데이터 연합 서비스의 현재 설정과 호환되는지를 표시합니다. true 로 설정된 경우 서버는 비교 연산자의 데이터 정렬을 무

속성	설명
	<p>시할 수 있고 숨어는 소스에 안전하게 푸시될 수 있습니다. 기본값은 <code>False</code> 로 설정됩니다.</p> <p>예</p> <p><code>compCollationCompatible=true</code></p>
<code>sortCollationCompatible</code>	<p><code>True</code>/예 또는 <code>False</code>/아니요</p> <p>데이터 소스의 정렬 작업(<code>ORDER BY</code>)을 위한 데이터 정렬이 데이터 연합 서비스의 현재 설정과 호환되는지를 표시합니다. <code>True</code> 로 설정된 경우 서버는 정렬 작업의 데이터 정렬을 무시할 수 있고 (<code>ORDER BY</code>) 식은 소스에 안전하게 푸시될 수 있습니다. 기본값은 <code>False</code> 로 설정됩니다.</p> <p>예</p> <p><code>sortCollationCompatible=true</code></p>
<code>longVarCharMaxSize</code>	<p><code>longvarchar</code> 데이터 유형(텍스트 데이터 유형과 유사)의 크기를 제한합니다.</p>
<code>varCharMaxSize</code>	<p><code>varchar</code> 데이터 유형의 크기를 제한합니다. 기본값은 <code>-1</code> 이며 이는 문자가 잘리지 않음을 의미합니다.</p>
<code>arrayFetchBufferSize</code>	<p>각 배열 반입 전용 버퍼의 최대 크기(바이트)를 정의합니다. 기본값은 <code>65536</code> 입니다.</p>
<code>enableArrayFetchSizeOptimization</code>	<p>배열 반입 크기 최적화를 사용하거나 해제합니다. 기본값은 <code>true</code> 입니다.</p>
<code>maxConnectionIdleTime</code>	<p>연결 풀에 저장되는 최대 유휴 연결 시간입니다. 단위는 밀리 초이며, <code>-1</code> 은 제한 없음을 의미합니다. 이 매개 변수가 설정되지 않으면 기본 연결 서버 풀 시간 제한이 사용됩니다(<code>10mn</code>) <code>100000</code>.</p> <p>가능한 값:</p> <ol style="list-style-type: none"> <code>-1</code>: 시간 제한 없음. 전체 래퍼 수명 시간 동안 풀에서 연결이 유지됩니다. <code>0</code>: 연결이 풀에서 관리되지 않습니다. <code>> 0</code>: 연결이 유휴 상태로 지속될 수 있는 최대 시간 값입니다(단위: 밀리초).
<code>enableUpdateQueries</code>	<p><code>True</code>/예 또는 <code>False</code>/아니요</p> <p>업데이트 쿼리를 실행할 수 있는지 여부를 나타냅니다.</p> <p>기본값은 <code>True</code> 로 설정됩니다.</p>
<code>enableTemporaryTableQueries</code>	<p><code>True</code>/예 또는 <code>False</code>/아니요</p> <p>임시 테이블 쿼리를 실행할 수 있는지 여부를 나타냅니다.</p> <p>기본값은 <code>True</code> 로 설정됩니다.</p>
<code>maxValuesInInClause</code>	<p><code>IN</code> 절에 있는 값의 최대 개수를 지정합니다.</p> <p>기본값은 <code>0</code> 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.</p>

속성	설명
semiJoinMaxQueries	semi-join 연산자가 실행할 수 있는 최대 쿼리 수를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinTempTableMinCardinality	semi-join 연산자의 임시 테이블 전략에서 차원의 최소 카디널리티를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinTempTableMaxCardinality	semi-join 연산자의 임시 테이블 전략에서 차원의 최대 카디널리티를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinExecutionStrategies	semi-join 연산자의 실행 전략 목록을 기본 설정 순서대로 지정합니다. 가능한 값: <ul style="list-style-type: none"> • DEFAULT DEFAULT 는 시스템 매개 변수 SEMI_JOIN_EXECUTION_STRATEGIES 의 값이 사용됨을 의미합니다. • 기본 설정의 순서대로 쉼표로 구분된 I,T,P 의 조합으로서, I 는 IN 쿼리 실행 전략, T 는 임시 테이블 실행 전략, P 는 매개 변수화된 쿼리 실행 전략을 나타냅니다. 예: <ul style="list-style-type: none"> ◦ T,P,I ◦ I,T ◦ P ◦ 전략 없음 I,T,P 중 하나가 없는 경우에는 래퍼에서 해당 실행 전략을 지원하지 않습니다. 참고: NONE 은 래퍼가 지원하는 실행 전략이 없음을 의미합니다. 반환되는 값은 null 또는 빈 문자열이 아니어야 합니다.
allowPartialResults	True/예 또는 False/아니요 이 매개 변수는 maxRows 매개 변수와 함께 사용됩니다. maxRows 가 양의 값으로 설정되면 쿼리에서 지정된 제한을 넘는 수의 행이 반환되어 기본적으로 예외가 throw 됩니다. allowPartialResults 매개 변수를 true 로 설정하여 이 동작을 변경할 수 있습니다. 기본값은 false 입니다.
maxRows	반환할 최대 행 수를 정의합니다. 이 매개 변수는 allowPartialResults 매개 변수와 조합하여 사용됩니다. maxRows 를 양의 값으로 설정하면 쿼리에서 지정된 제한을 넘는 수의 행이 반환되어 기본적으로 예외가 발생합니다. allowPartialResults 매개 변수를 true 로 설정하여 이 동작을 변경할 수 있습니다. 기본값은 제한이 없음을 나타내는 0 입니다.
maxLevelOfFunctionNesting	중첩 기능에서 지원되는 수준의 최대 개수를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.

4.3.2 MySQL 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 MySQL 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
datasourceCompCollation	비교 시 사용할 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가 제외)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.
datasourceSortCollation	정렬 작업에 사용할 소스 데이터 정렬(ORDER BY)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.
datasourceBinaryCollation	비교 시 사용할 이진 데이터 정렬로 평가해야 하는 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 이진 데이터 정렬 의미가 필요한 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.
unicodeStrings	True/예 또는 False/아니요 데이터베이스에 푸시할 문자열 상수에 유니코드 구문을 사용할지 여부를 지정합니다. 기본값은 False로 설정됩니다.

4.3.3 Teradata 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 Teradata 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
sampleSize	SAMPLE 연산자를 사용하여 반환할 최대 행 수를 정의합니다.

관련 링크

[관계형 데이터 소스의 전체 커넥터 기능 목록 \[페이지 52\]](#)

4.3.4 Sybase ASE 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 Sybase ASE 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
setQuotedIdentifier	True/예 또는 False/아니요 setQuotedIdentifier=true 이면 인용 문자열 식별자가 "로 설정됩니다.

관련 링크

4.3.5 SQL Server 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 SQL Server 커넥터에서 구성할 수 있는 특정 속성 목록이 나열되어 있습니다.

속성	설명
datasourceCompCollation	비교 시 사용할 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가 제외)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다. 예 datasourceCompCollation=Latin1_general_ci_ai
datasourceSortCollation	정렬 작업에 사용할 소스 데이터 정렬(ORDER BY)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다. 예 datasourceSortCollation=Latin1_general_ci_as
datasourceBinaryCollation	비교 시 사용할 이진 데이터 정렬로 평가해야 하는 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 이진 데이터 정렬 의미가 필요한 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다. 예 datasourceBinaryCollation=Latin1_general_bin
unicodeStrings	True/예 또는 False/아니요 데이터베이스에 푸시할 문자열 상수에 유니코드 구문을 사용할지 여부를 지정합니다. 기본값은 False 로 설정됩니다.

4.3.6 Generic ODBC 또는 JDBC 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 Generic 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
sqlDialect	데이터베이스에서 지원되는 SQL 언어를 식별합니다. 다음 항목 중 하나입니다.

속성	설명
	<ul style="list-style-type: none"> • sql92 • sql99(이후 사용을 위해 예약됨) • jdbc3(JDBC 구문은 외부 조인에 사용됨) • odbc • oracle • sqlserver • ids(Informix Dynamic Server) • teradata • maxdb • greenplum • postgresql <p>sourceType 매개 변수에서 식별된 대로 소스에서 지원되는 SQL 언어로 기본 설정됩니다. sourceType 이 정의되지 않은 경우 sql92 로 기본 설정됩니다.</p>
supportsCatalog	커넥터가 카탈로그의 개념을 지원하는지 여부를 나타냅니다. 기본값은 true 로 설정됩니다.
supportsSchema	커넥터가 스키마의 개념을 지원하는지 여부를 나타냅니다. 기본값은 true 로 설정됩니다.
supportsBoolean	<p>True/예 또는 False/아니요</p> <p>JDBC 드라이버 또는 데이터베이스에서 첫 번째 클래스 개체로 부울을 지원하지 않는 경우 False 입니다. 이 매개 변수의 기본값은 데이터베이스에 따라 달라집니다. 매개 변수가 지원되는 소스 유형 중 하나인 경우 이 매개 변수는 올바른 값으로 이미 설정되어 있습니다. 그러나 이 매개 변수는 무시할 수 있습니다. 기본값은 False 로 설정됩니다.</p>
useIndexInOrderBy	<p>제출된 쿼리의 ORDER BY 절에서 별칭(열 이름) 대신 인덱스(열 위치)를 사용해야 하는지 여부를 지정합니다. 기본값은 False 입니다(ORDER BY 절에서 별칭을 적절하게 처리하지 않는 데이터베이스 제외).</p> <p>예</p> <p>열 2 와 3 별로 정렬하는 경우 ORDER BY C2, C3 대신 ORDER BY 2, 3 이 생성됩니다.</p>
escapeIdentifierQuoteString	<p>식별자 내에 표시할 때 식별자 인용 문자열</p> <p>(java.sql.DatabaseMetaData#getIdentifierQuoteString 에서 반환)을 이스케이프하는 데 사용되는 문자열을 정의합니다. 기본적으로 이 이스케이프 문자열은 식별자 인용 문자열 자체로 설정됩니다. ""으로 설정된 경우 이스케이프가 수행되지 않습니다.</p>
ignoreKeys	<p>True/예 또는 False/아니요</p> <p>래퍼가 키/외래 키 메타데이터를 가져오기 위해 JDBC 드라이버를 쿼리해서는 안 되는 경우 아니요로 지정합니다. Sun JDBC-ODBC 브리지는 이러한 호출을 지원하지 않으며 이 옵션은 true 로 설정해야 합니다. 기본값: 아니요.</p>

속성	설명
supportsTemporaryTables	True/예 또는 False/아니요 소스가 임시 테이블을 지원하지 않거나 임시 테이블의 생성을 비활성화하려는 경우 False 입니다. 기본값은 False 로 설정됩니다.
supportsTableCardinality	True/예 또는 False/아니요 소스에서 테이블 카디널리티를 계산할 수 없는 경우 False 입니다. 기본값은 true 로 설정됩니다.
supportsColumnCardinality	True/예 또는 False/아니요 소스에서 열 카디널리티를 계산할 수 없는 경우 False 입니다. 기본값은 true 로 설정됩니다.

관련 링크

[관계형 데이터 소스의 전체 커넥터 기능 목록 \[페이지 52\]](#)

4.3.7 Oracle 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 SQL Server 커넥터에서 구성할 수 있는 특정 속성 목록이 나열되어 있습니다.

속성 유형	속성 값	속성 설명
unicodeStrings	True/예 또는 False/아니요	데이터베이스에 푸시할 문자열 상수에 유니코드 구문을 사용할지 여부를 지정합니다. 기본값은 False 로 설정됩니다.

4.3.8 SAP Hana 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 SQL Server 커넥터에서 구성할 수 있는 특정 속성 목록이 나열되어 있습니다.

속성 유형	속성 값	속성 설명
unicodeStrings	True/예 또는 False/아니요	데이터베이스에 푸시할 문자열 상수에 유니코드 구문을 사용할지 여부를 지정합니다. 기본값은 False 로 설정됩니다.

4.3.9 MaxDB 데이터 소스의 특정 커넥터 속성 목록

아래 표에는 SQL Server 커넥터에서 구성할 수 있는 특정 속성 목록이 나열되어 있습니다.

속성 유형	속성 값	속성 설명
unicodeStrings	True/예 또는 False/아니요	데이터베이스에 푸시할 문자열 상수에 유니코드 구문을 사용할지 여부를 지정합니다. 기본값은 False로 설정됩니다.

4.4 SAS 용 커넥터 구성

SAS 를 구성하기 전에 SAS 미들웨어 및 드라이버를 먼저 설치해야 합니다.

SAS 용 미들웨어 및 드라이버 구성에 대한 자세한 내용은 데이터 액세스 가이드를 참조하십시오.

4.4.1 SAS 데이터 소스의 커넥터 속성 목록

아래 표에는 SAS 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
maxConnections	기본 데이터베이스에 대한 최대 동시 연결 수입니다. 0 은 제한 없음을 의미합니다. 기본값은 0 으로 설정됩니다.
maxConnectionIdleTime	연결 풀에 저장되는 최대 유휴 연결 시간입니다. 단위는 밀리초입니다. 0 은 제한 없음을 의미합니다. 기본값은 60000(60 초)으로 설정됩니다.
maxPoolSize	풀에 저장할 최대 유휴(사용 가능한) 연결 수입니다. 0 은 제한 없음을 의미합니다. 기본값은 32 로 설정됩니다.
maxIdlePools	유휴로 설정할 수 있는 최대 풀 수입니다. 이 값에 도달하면 사용되지 않은 가장 오래된 풀이 닫히고 제거됩니다. 0 은 제한 없음을 의미합니다. 기본값은 24 로 설정됩니다.
connectionTestQuery	기본 데이터베이스에 대한 연결이 올바른지 여부를 확인하는 데 사용할 수 있는 SQL 테스트 쿼리입니다. 주의: 이 쿼리는 쉽게 실행할 수 있어야 합니다. 빈 문자열은 테스트 쿼리가 없음을 의미합니다. 기본값은 빈 문자열로 설정됩니다. 예 테스트 쿼리의 예로 <code>SELECT 1 FROM DUAL</code> 이 가능합니다.
connectionFailureDetectionOnError	기본 데이터베이스에서 <code>SQLException</code> 발생 시 수행되어야 하는 연결 오류 감지 종류를 나타내는 키워드입니다.
connectionFailureSQLStates	기본 데이터베이스에서 <code>SQLException</code> 이 발생할 때 연결 오류를 감지하는 데 사용할 수 있는 특정 <code>SQLState</code> 코드 목록입니다. 연결 오류에 대한 표준 코드(2 자 클래스 08 로 시작)는 여기서 지정할 필요가 없습니다. Oracle 의 특정 코드 예: 61000: (ORA-00028: your session has been killed). 요소는 ; 문자로 구분됩니다(요소 간에 공백 없음). 기본값은 비어 있음으로 설정됩니다.

속성	설명
driverProperties	<p>드라이버 속성 목록입니다. 속성은 ; 문자(속성 사이에 공백 없음)로 구분해야 합니다.</p> <p>데이터베이스 연결에 사용 중인 드라이버에 대해 사용할 수 있는 속성과 동일한 속성을 입력할 수 있습니다. 속성 목록은 드라이버 설명서를 참조하십시오.</p> <p>예</p> <pre>driverProperties=selectMethod \=cursor;connectionRetryCount\=2</pre>
sessionProperties	<p>데이터베이스에서 설정된 세션 속성 목록입니다. 속성은 ; 문자(속성 사이에 공백 없음)로 구분해야 합니다.</p> <p>연결 중인 데이터베이스에 대해 사용할 수 있는 속성과 동일한 속성을 입력할 수 있습니다. 속성 목록은 데이터베이스 설명서를 참조하십시오.</p> <p>예</p> <pre>sessionProperties=selectMethod \=cursor;connectionRetryCount\=2</pre>
capabilities	<p>데이터베이스에서 지원되는 모든 기능의 목록입니다. 요소는 ; 문자로 구분됩니다(요소 간에 공백 없음).</p> <p>예</p> <pre>capabilities=fullSQL\=true;outerjoin \=false;rightouterjoin\=true</pre>
useParameterInlining	<p>True 로 설정된 경우, JDBC 래퍼는 <code>java.sql.PreparedStatement</code> 개체가 아니라 <code>java.sql.Statement</code> 개체를 사용하여 매개 변수화된 쿼리를 실행합니다. 매개 변수화된 쿼리는 인라이닝되고 자리 표시자를 상수 값으로 교체합니다. 이 옵션은 잘 작성된(well-prepared) 문을 지원하지 않는 JDBC 드라이버에 유용합니다. 기본값은 False 로 설정됩니다.</p>
castColumnType	<p><code>databasetype=jdbctype</code> 유형 매핑 목록입니다. 이 매개 변수는 드라이버에서 수행된 기본 매핑이 잘못되었거나 미완료된 경우 유용합니다. 참고: 공식적으로 지원되는 데이터베이스의 경우 유형 매핑은 암시적으로 설정되지만 사용자가 이 유형 매핑을 덮어쓸 수 있습니다.</p> <p>예</p> <pre>Oracle JDBC 드라이버의 경우 castColumnType=FLOAT\=FLOAT;BLOB \=BLOB</pre>
enableUpdateQueries	<p>True/예 또는 False/아니요</p> <p>업데이트 쿼리를 실행할 수 있는지 여부를 나타냅니다.</p> <p>기본값은 True 로 설정됩니다.</p>
enableTemporaryTableQueries	<p>True/예 또는 False/아니요</p> <p>임시 테이블 쿼리를 실행할 수 있는지 여부를 나타냅니다.</p>

속성	설명
	기본값은 True 로 설정됩니다.
defaultFetchSize	<p>java.sql.Statement 를 만들 때 설정할 기본 반입 크기입니다. 0 은 반입 크기가 설정되어 있지 않다는 뜻입니다.</p> <p>더 많은 행이 필요할 때 데이터베이스에서 반입해야 하는 행 수에 관한 힌트를 커넥터에 제시합니다.</p> <p>기본값: 0(반입 크기가 설정되어 있지 않음)</p>
compCollationCompatible	<p>True/예 또는 False/아니요</p> <p>데이터 소스의 비교 작업을 위한 데이터 정렬이 데이터 연합 서비스의 현재 설정과 호환되는지를 표시합니다. true 로 설정된 경우 서버는 비교 연산자의 데이터 정렬을 무시할 수 있고 숨어는 소스에 안전하게 푸시될 수 있습니다. 기본값은 False 로 설정됩니다.</p> <p>예</p> <p>compCollationCompatible=true</p>
sortCollationCompatible	<p>True/예 또는 False/아니요</p> <p>데이터 소스의 정렬 작업(ORDER BY)에 대한 데이터 정렬이 데이터 연합 쿼리 서비스의 현재 설정과 호환되는지 여부를 나타냅니다. True 로 설정된 경우 서버는 정렬 작업의 데이터 정렬을 무시할 수 있고 (ORDER BY) 식은 소스에 안전하게 푸시될 수 있습니다. 기본값은 False 로 설정됩니다.</p> <p>예</p> <p>sortCollationCompatible=true</p>
datasourceCompCollation	<p>비교 시 사용할 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가 제외)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.</p> <p>예</p> <p>datasourceCompCollation=Latin1_general_ci_ai</p>
datasourceSortCollation	<p>정렬 작업에 사용할 소스 데이터 정렬(ORDER BY)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가 설정되어 있지 않는 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.</p> <p>예</p> <p>datasourceSortCollation=Latin1_general_ci_as</p>
datasourceBinaryCollation	<p>비교 시 사용할 이진 데이터 정렬로 평가해야 하는 소스 데이터 정렬(LIKE / NOT LIKE 및 함수 평가)입니다. 이 매개 변수는 SQL Server 및 MySQL 이 이진 데이터 정렬 의미가 필요한 쿼리에서 데이터 정렬 절을 추가하는 데 사용됩니다. 이 매개 변수가</p>

속성	설명
	<p>설정되어 있지 않은 경우 이 연산에는 데이터 정렬 절이 생성되지 않습니다. 이 매개 변수는 기본적으로 설정되어 있지 않습니다.</p> <p>예</p> <p><code>datasourceBinaryCollation=Latin1_general_bin</code></p>
<code>sqlDialect</code>	<p>데이터베이스에서 지원되는 SQL 언어를 식별합니다. 다음 항목 중 하나입니다.</p> <ul style="list-style-type: none"> • <code>sql92</code> • <code>sql99</code>(이후 사용을 위해 예약됨) • <code>oracle</code> • <code>sqlserver</code> • <code>jdbc3</code>(JDBC 구문은 외부 조인에 사용됨) • <code>sas</code> <p><code>sourceType</code> 매개 변수에서 식별된 대로 소스에서 지원되는 SQL 언어로 기본 설정됩니다. <code>sourceType</code> 이 정의되지 않은 경우 <code>sql92</code> 로 기본 설정됩니다.</p>
<code>useIndexInOrderBy</code>	<p>제출된 쿼리의 <code>ORDER BY</code> 절에서 별칭(열 이름) 대신 인덱스(열 위치)를 사용해야 하는지 여부를 지정합니다. 기본값은 <code>False</code> 입니다(<code>ORDER BY</code> 절에서 별칭이 적절히 처리되지 않은 데이터베이스 제외).</p> <p>예</p> <p>열 2 와 3 별로 정렬하는 경우 <code>ORDER BY C2, C3</code> 대신 <code>ORDER BY 2, 3</code> 이 생성됩니다.</p>
<code>escapeIdentifierQuoteString</code>	<p>식별자 내에 표시할 때 식별자 인용 문자열 (<code>java.sql.DatabaseMetaData#getIdentiferQuoteString</code> 에서 반환)을 이스케이프하는 데 사용되는 문자열을 정의합니다. 기본적으로 이 이스케이프 문자열은 식별자 인용 문자열 자체로 설정됩니다. ""으로 설정된 경우 이스케이프가 수행되지 않습니다.</p>
<code>ignoreKeys</code>	<p><code>True</code>/예 또는 <code>False</code>/아니요</p> <p>래퍼가 키/외래 키 메타데이터를 가져오기 위해 JDBC 드라이버를 쿼리해서는 안 되는 경우 <code>아니요</code>로 지정합니다. Sun JDBC-ODBC 브리지는 이러한 호출을 지원하지 않으며 이 옵션은 <code>true</code> 로 설정해야 합니다. 기본값: <code>아니요</code>.</p>
<code>transactionIsolation</code>	<p>트랜잭션 격리 수준입니다. 다음 항목 중 하나입니다.</p> <ul style="list-style-type: none"> • <code>TRANSACTION_READ_COMMITTED</code> • <code>TRANSACTION_READ_UNCOMMITTED</code> • <code>TRANSACTION_REPEATABLE_READ</code> • <code>TRANSACTION_SERIALIZABLE</code> <p>기본값: 설정되지 않음</p>
<code>setFetchForwardDirection</code>	<p><code>True</code>/예 또는 <code>False</code>/아니요</p> <p>반입 전달을 명시적으로 설정해야 하는 경우 <code>True</code> 입니다. 기본값: <code>False</code></p>

속성	설명
setReadOnly	<p>True/예 또는 False/아니요</p> <p>연결을 읽기 전용으로 설정하면 안 되는 경우 False 입니다. 기본값: False</p>
metadataFetchMode	<p>SAS 데이터 소스 전용 메타데이터 반입 모드입니다.</p> <p>다음 항목 중 하나입니다.</p> <ul style="list-style-type: none"> eager: 모든 메타데이터가 한 번에 반입됩니다. lazy: 메타데이터가 요청 시 반입됩니다. <p>기본값: lazy</p>
sasWeights	<p>SAS 언어로 쿼리 생성 시 FROM 절에서 테이블 이름과 테이블 순서를 지정하는 데 사용되는 가중치 사이의 매핑입니다. FROM 절의 테이블은 가중치에 따라 내림차순으로 정렬됩니다. 가중치는 기본적으로 테이블 카디널리티로 설정되지만 이 매개 변수를 사용하여 무시할 수 있습니다. 이 순서 지정은 내부 조인에 대해서만 수행됩니다.</p> <p>여기서 테이블 이름은 래퍼에서 가져온 이름입니다. 가중치는 긴 값입니다.</p> <p>이 매개 변수가 지정되지 않거나 지정된 테이블에서 가중치가 정의되지 않은 경우 가중치는 기본적으로 데이터 연합 서비스에서 설정된 대로 테이블의 카디널리티입니다.</p> <p>테이블 이름을 알 수 없는 경우는 무시됩니다.</p> <p>이 매개 변수는 <code>sqlDialect="sas"</code>인 경우에만 고려됩니다.</p> <p>예</p> <pre>sasWeights=EMPLOYEE\=16;DEPARTMENT\=4</pre> <p>이 두 개의 테이블을 조인하여 SAS 에서 쿼리를 푸시하면 EMPLOYEE 테이블은 이 설정을 사용하여 DEPARTMENT 테이블 이전에 나타납니다.</p>
addCompensationPredicates	<p>True/예 또는 False/아니요</p> <p>보정 조건부의 생성을 비활성화하려는 경우 False 입니다.</p> <p>보정 조건부는 null 값에 대한 SQL 의미를 적용하기 위해 추가되는 IS NOT NULL 조건입니다(SAS 는 SQL 과 호환 안 됨).</p> <p>이런 추가 조건은 성능에 영향을 미칠 수 있습니다. 사용자가 null 값에 대한 SQL 의미의 적용에 신경을 쓰지 않으면 이 매개 변수를 false 로 설정할 수 있습니다.</p> <p>이 매개 변수는 <code>sqlDialect="sas"</code>인 경우에만 고려됩니다.</p> <p>기본값은 True 로 설정됩니다.</p>
trimTrailingSpaces	<p>True/예 또는 False/아니요</p> <p>일부 JDBC 드라이버는 공백으로 채워진 메타데이터를 반환하고 있습니다. 이 매개 변수를 예로 설정하면 카탈로그, 스키마, 테이블, 열, 키 및 외래 키 이름에서 추가 공간이 제거됩니다. 이 매개 변수의 기본값은 아니요입니다.</p>
maxValuesInInClause	<p>IN 절에 있는 값의 최대 개수를 지정합니다.</p>

속성	설명
	기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinMaxQueries	semi-join 연산자가 실행할 수 있는 최대 쿼리 수를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinTempTableMinCardinality	semi-join 연산자의 임시 테이블 전략에서 차원의 최소 카디널리티를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinTempTableMaxCardinality	semi-join 연산자의 임시 테이블 전략에서 차원의 최대 카디널리티를 지정합니다. 기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.
semiJoinExecutionStrategies	semi-join 연산자의 실행 전략 목록을 기본 설정 순서대로 지정합니다. 가능한 값은 다음과 같습니다. <ul style="list-style-type: none"> • DEFAULT DEFAULT 는 시스템 매개 변수 SEMI_JOIN_EXECUTION_STRATEGIES 의 값이 사용됨을 의미합니다. • 기본 설정의 순서대로 쉼표로 구분된 I,T,P 의 조합으로서, I 는 IN 쿼리 실행 전략, T 는 임시 테이블 실행 전략, P 는 매개 변수화된 쿼리 실행 전략을 나타냅니다. 예: <ul style="list-style-type: none"> ◦ T,P,I ◦ I,T ◦ P ◦ 전략 없음 I,T,P 중 하나가 없는 경우에는 래퍼에서 해당 실행 전략을 지원하지 않습니다. 참고: NONE 은 래퍼가 지원하는 실행 전략이 없음을 의미합니다. 반환되는 값은 null 또는 빈 문자열이 아니어야 합니다.

4.4.2 카디널리티를 기준으로 from 절에서 테이블을 정렬하여 SAS 쿼리 최적화

SAS 는 from 절에서의 테이블 정렬에 의해 영향을 받습니다. SAS/Share 서버로부터 가장 빠르게 응답을 받으려면 from 의 테이블 이름이 카디널리티를 기준으로 내림차순으로 나타나야 합니다.

데이터 연합 응용 프로그램이 이 정렬 기준으로 테이블을 생성하도록 하여 데이터 연합 응용 프로그램의 통계를 정확하게 유지할 수 있습니다. 데이터 연합 관리 도구를 사용하면 됩니다.

테이블 정렬을 수동으로 제어하려는 경우 SAS JDBC 커넥터에 대해 sasWeights 리소스 속성을 설정할 수도 있습니다.

4.5 SAP NetWeaver BW 에 대한 커넥터 구성

4.5.1 SAP NetWeaver BW 데이터 소스의 커넥터 속성 목록

아래 표에는 SAP NetWeaver BW 커넥터에서 구성할 수 있는 특정 속성이 나열되어 있습니다.

속성	설명
packageSize	<p>콜백을 위한 패키지 크기.</p> <p>SAP NetWeaver BW 에서 데이터 연합 쿼리 엔진으로 반환된 패키지당 행 수입니다.</p> <p>기본값은 200 입니다.</p> <p>예</p> <pre>packageSize=300</pre>
programIDMapping	<p>SAP NetWeaver BW 에서 Data Federator 에 연결하는 데 사용하는 콜백의 프로그램 ID 를 정의합니다. ID 는 서버 이름 => 프로그램 ID 의 매핑 목록으로 제공됩니다. 이 목록은 ';'으로 구분되는 키/값이 포함된 문자열 형식입니다. 키는 서버 이름을, 값은 프로그램 ID 를 나타냅니다. 각 ID 는 SAP NetWeaver BW 에서 만든 RFC 대상의 이름과 일치해야 합니다.</p> <p>이 속성이 정의되지 않으면 Data Federator 에서 RFC 대상이 자동으로 만들어집니다. 이 RFC 대상은 RFC 대상 이름과 동일한 프로그램 ID 를 사용하여 만들어집니다.</p> <p>만들어지는 RFC 대상 형식은 <code><DF_JCO_>+<hostname>+<_>+<counter></code>입니다. <code><hostname></code>은 로컬 호스트의 이름이고 <code><counter></code>는 0 부터 9 사이의 수입니다. <code><hostname></code>의 최대 길이는 23 자로 제한됩니다. 로컬 호스트의 이름이 23 자보다 길면 23 자의 접두사만 RFC 대상 이름에 사용됩니다.</p> <p>속성이 정의되었지만 현재 서버에 대해 나열된 매핑이 없는 경우에는 오류가 반환됩니다.</p> <p>이 속성은 기본값이 없으므로 자동 모드가 사용됩니다.</p> <p>예제 1</p> <pre>MySIA.AdaptiveProcessingServer=RFC1</pre> <p>예제 2</p> <pre>MySIA.DFServer1=RFC1;MySIA.DFServer2=RFC2;...</pre>
useBinaryXML	<p>true 로 설정되면 커넥터와 SAP Netweaver BW 서버 간에 교환한 데이터는 일반 텍스트 형식이 아닌 이진 XML 형식으로 압축됩니다. 이를 통해 성능이 향상됩니다. 기본값은 true 입니다.</p> <p>예</p> <pre>useBinaryXML=true</pre>

속성	설명
checkUnits	<p>true 로 설정되면 단위가 없는 계수를 사용하는 쿼리가 거부되고 오류가 발생합니다. 기본값은 false 이며, 단위 없이도 계수를 사용할 수 있습니다.</p> <p>예</p> <p>checkUnits=false</p>
forcedCapabilities	<p>SAP NetWeaver BW 대신 데이터 연합 쿼리 엔진이 작업을 수행하도록 하려는 경우 SAP NetWeaver BW 커넥터의 기능을 수동으로 제한할 수 있습니다.</p> <p>SAP NetWeaver BW 에 위임할 데이터 연합 쿼리 엔진의 기능을 입력하십시오.</p> <p>유효한 값은 다음과 같습니다.</p> <ul style="list-style-type: none"> • SCAN_ONLY 데이터 연합 쿼리 엔진에서 스캔 기능만 SAP NetWeaver BW 에 위임합니다. • PROJECTIONS_ONLY 데이터 연합 쿼리 엔진에서 프로젝션 기능만 SAP NetWeaver BW 에 위임합니다. • 입력 안 함 - 데이터 연합 쿼리 엔진의 모든 유효 작업을 SAP NetWeaver BW 에 위임합니다. <p>예</p> <p>forcedCapabilities=SCAN_ONLY</p>
jcoDestinationProperties	<p>JCO 대상 속성 목록.; 문자를 사용하여 속성을 구분합니다(공백 없음).</p> <p>사용 가능한 속성은 JCo API 용 API 참조서의 DestinationDataProvider 인터페이스에 설명되어 있습니다.</p>
jcoServerProperties	<p>JCO 서버 속성 목록.; 문자를 사용하여 속성을 구분합니다(공백 없음).</p> <p>사용 가능한 속성은 JCo API 용 API 참조서의 ServerDataProvider 인터페이스에 설명되어 있습니다.</p>
authorityCheck	<p>SAP NetWeaver BW 에서 매개 변수를 설정하여 권한을 검사할지 여부를 표시합니다.</p> <p>SAP NetWeaver BW 의 매개 변수는 SAP NetWeaver BW 에서 다른 작업에 대한 수행 여부를 지정합니다.</p> <p>사용자 계정에 요청된 데이터를 볼 수 있는 권한(읽기)이 있는지 여부를 검사하거나, 권한 검사를 수행하지 않음(없음).</p> <ul style="list-style-type: none"> • true(기본값): SAP NetWeaver BW 가 읽기 권한을 검사합니다. • false: SAP NetWeaver BW 에서 권한을 검사하지 않습니다.
pingTimeout	<p>SAP 서버를 Ping 할 때 사용하는 밀리초 단위의 시간 제한 값. 기본값은 10000 밀리초입니다.</p> <p>예</p> <p>pingTimeout=60000</p>
maxValuesInInClause	<p>IN 절에 있는 값의 최대 개수를 지정합니다.</p> <p>기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.</p>

속성	설명
semiJoinMaxQueries	<p>semi-join 연산자가 실행할 수 있는 최대 쿼리 수를 지정합니다.</p> <p>기본값은 0 이며, 이는 제한이 없거나 제한을 알 수 없음을 의미합니다.</p>
semiJoinExecutionStrategies	<p>semi-join 연산자의 실행 전략 목록을 기본 설정 순서대로 지정합니다.</p> <p>가능한 값은 다음과 같습니다.</p> <ul style="list-style-type: none"> • DEFAULT DEFAULT 는 시스템 매개 변수 SEMI_JOIN_EXECUTION_STRATEGIES 의 값이 사용됨을 의미합니다. • 기본 설정의 순서대로 심포로 구분된 I,T,P 의 조합으로서, I 는 IN 쿼리 실행 전략, T 는 임시 테이블 실행 전략, P 는 매개 변수화된 쿼리 실행 전략을 나타냅니다. 예: <ul style="list-style-type: none"> ◦ T,P,I ◦ I,T ◦ P ◦ 전략 없음 <p>I,T,P 중 하나가 없는 경우에는 래퍼에서 해당 실행 전략을 지원하지 않습니다. 참고: NONE 은 래퍼가 지원하는 실행 전략이 없음을 의미합니다. 반환되는 값은 null 또는 빈 문자열이 아니어야 합니다. 참고: T 및 P 실행 전략은 SAP NetWeaver BW 커넥터에서 사용할 수 없습니다.</p>
enableAuthorizationsFiltering	<p>true 로 설정하면 권한 필터링이 활성화되어 커넥터에서 현재 사용자에게 대해 정의된 SAP BI 권한을 충족시키기 위해 필터가 자동으로 추가됩니다. 권한 필터링이 활성화되면 권한이 없는 데이터에 접근을 시도할 경우 사용자에게 오류가 표시됩니다. 기본값은 false 입니다.</p> <p>예</p> <pre>enableAuthorizationsFiltering=true</pre>
debugReportPrefix	<p>ABAP 보고서 프로그램의 이름에 포함된 문자열로서 최대 길이는 11 입니다. 생성된 프로그램 이름은 Z_RS DRI_DF_TXT_ \${debugReportPrefix}_ID 또는 Z_RS DRI_DF_DBG_ \${debugReportPrefix}_ID 이며 여기서 ID 는 래퍼 쪽에서 생성된 3 자리 숫자 값입니다. 생성된 프로그램은 SAP 전문가가 DF Facade 의 버그를 전달하는 데 사용할 수 있습니다.</p> <p>설정하지 않으면 프로그램이 생성되지 않습니다.</p> <p>예</p> <pre>MY_HOSTNAME</pre>
gatewayHostname	<p>SAP NetWeaver BW 게이트웨이를 호스팅하는 컴퓨터의 이름입니다.</p> <p>지정하지 않으면 RFC 가 실행되어 SAP NetWeaver BW 에서 값을 선택합니다.</p> <p>예</p> <pre>gatewayHostname=server.wdf.sap.corp</pre>

속성	설명
gatewayServiceName	<p>SAP NetWeaver BW 게이트웨이 서비스의 이름 또는 포트 번호입니다.</p> <p>지정하지 않으면 RFC 가 실행되어 SAP NetWeaver BW 에서 값을 선택합니다.</p> <p>예</p> <p>gatewayServiceName=sapgw50</p> <p>예</p> <p>gatewayServiceName=3350</p>

4.5.2 SAP NetWeaver BW 가 데이터 연합 서비스에 연결하는 데 사용하는 콜백 ID 를 수동으로 설정

SAP NetWeaver BW 는 데이터 연합 서비스에 연결하기 위해 콜백 ID 를 사용합니다. 콜백은 SAP NetWeaver BW 커넥터의 첫 번째 쿼리가 실행될 때 자동으로 등록되지만 조직의 보안 정책을 준수하려는 등의 이유로 콜백을 변경하고자 할 수 있습니다.

1. SAP 로그온을 열고 SAP 시스템에 로그인합니다.
2. 트랜잭션 텍스트 필드에 se37 을 입력하고 **실행**을 클릭합니다.
3. 함수 모듈 **RSDRI_DF_CONFIGURE** 를 입력하고 **실행**을 클릭합니다.

그러면 매개 변수 패널이 열립니다.

4. 매개 변수를 다음과 같이 설정합니다.

_ONLY_CHECK	"(비워 둡)
_RFC_DESTINATION	DF_JCO_ <some-hostname> _ <some-sid>
_REMOVE_CONFIGURATION	"(비워 둡)

두 번째 매개 변수에서 **<some-hostname>**을 데이터 연합 서비스를 실행하는 서버를 설치한 컴퓨터의 호스트 이름으로 바꿉니다.

_<some-sid>를 고유 시스템 식별자로 사용하여 **<some-hostname>** 값이 같을 수 있는 여러 연결을 구별합니다.

이 예에서는 DF_JCO_ **<MYHOST>** 가 데이터 연합 관리 도구에서 다시 사용해야 할 고유 식별자입니다.

5. 모듈을 실행합니다.

i 노트

RFC Destination already exists 라는 메시지가 표시될 수 있습니다.

_ONLY_CHECK 확인란의 선택을 취소했는지 확인하십시오.

6. **시스템, 로그오프**를 차례로 클릭합니다.
7. 데이터 연합 관리 도구를 열고 관리자 권한이 있는 사용자 계정을 사용하여 로그인합니다.
8. **커넥터 구성** 탭에서 SAP NetWeaver BW 커넥터를 편집합니다.

9. programIDMapping 속성에서 서버와 SAP NetWeaver BW 에서 사용하는 *L_RFC_DESTINATION* 문자열(프로그램 ID 라고도 함) 사이에 매핑을 추가합니다.

이 경우 programIDMapping 속성 값은 `MySIA.AdaptiveProcessingServer=DF_JCO_MYHOST` 입니다.

자세한 내용은 SAP NetWeaver BW 커넥터 속성 목록에서 programIDMapping 속성에 대한 설명을 참조하십시오.

10. 테이블에 쿼리를 실행하여 데이터가 사용 가능한지 테스트합니다.

관련 링크

[데이터 연합 관리 도구에서 커넥터 속성 변경](#) [페이지 31]

[SAP NetWeaver BW 데이터 소스의 커넥터 속성 목록](#) [페이지 44]

4.5.3 SAP NetWeaver BW 연결에 대한 콜백 ID 정리

현재 최대 callbackProgramID 수는 10 개입니다. 정상적인 실행 과정(데이터 연합 서비스를 실행하는 서버가 갑자기 중지되지 않음)에서는 callbackProgramID 가 SAP 서버에 자동으로 정리됩니다.

서버에서 callbackProgramID 를 더 이상 생성할 수 없을 경우(모두 사용한 경우) 오류가 발생합니다. 시스템이 갑자기 중지되어 콜백 이름이 자동으로 삭제되지 않은 경우 다음 절차에 따라 콜백 이름을 삭제할 수 있습니다.

1. SAP NetWeaver BW 서버에 로그인합니다.
2. sm59 트랜잭션을 입력합니다.
3. [TCP/IP 연결](#)을 클릭합니다.
4. 해당하는 각 연결(<DF_JCO_MYHOST_0> - <DF_JCO_MYHOST_9>)을 클릭한 다음 **삭제** 아이콘을 클릭합니다.

4.5.4 SAP 분석 권한을 사용하여 자동으로 데이터 필터링

SAP NetWeaver BW 에서 실행되는 쿼리는 항상 데이터베이스에서 일련의 데이터를 선택합니다. 권한 관련 특성이 이 데이터의 일부인 경우 쿼리를 실행하는 사용자에게 완전한 선택을 위한 충분한 권한이 있는지 확인해야 합니다. 그렇지 않을 경우 권한 부족 오류 메시지가 반환됩니다.

데이터 연합 서비스에서 분석 권한을 사용하고 큐브에 포함된 각 특성의 인증된 값에 대한 필터를 자동으로 추가하도록 지시할 수 있습니다. 이 기능은 단일 로그인 인증 모드를 사용하여 SAP NetWeaver BW 에 연결하는 경우에 특히 유용합니다. SAP Business Intelligence 플랫폼에 연결하는 각 사용자를 기준으로 데이터를 손쉽게 필터링할 수 있습니다.

기능 활성화

다음 SAP NetWeaver BW 커넥터 리소스 속성을 설정하여 이 기능을 활성화할 수 있습니다.

- enableAuthorizationsFiltering: true / false

데이터 연합 관리 도구를 사용하여 데이터 소스에서 사용하는 리소스를 구성할 수 있습니다.

필수 조건

이 기능은 *Data Federator* 퍼사드라는 SAP NetWeaver BW 의 구성요소에 의존합니다.

SAP NetWeaver BW 측에 대한 필수 조건을 확인하려면 다음 SAP Note 를 참조하십시오. <https://service.sap.com/sap/support/notes/1500945>

원칙

SAP NetWeaver BW 데이터 웨어하우징 워크벤치에서 권한 관련 특성으로 선언된 모든 특성은 자동으로 추가될 필터 집합을 계산하는 데 고려됩니다.

쿼리에 명시적 필터가 포함된 경우 추가 권한 필터가 추가되지 않습니다.

쿼리에 지정된 특성(열)에 대한 필터가 포함되어 있지 않으며 이 특성이 권한 관련 특성일 경우, 필터가 자동으로 인증된 반환 값에 추가됩니다.

계층구조 권한

데이터 연합 서비스는 계층구조를 표시하지 않습니다. 그러나 계층구조에 대한 분석 권한이 데이터 및 인증된 반환 값을 필터링하는 데 고려됩니다.

콜론 권한

값 기반 권한 이외에 집계 권한(콜론 권한이라고도 함)도 정의된 경우, 열이 SQL 쿼리의 SELECT 문에 있지 않더라도 데이터 연합 서비스는 항상 값 기반 권한을 고려합니다.

집계 권한 논리에 대한 자세한 내용은 SAP Note 1140831 - Colon authorization during query execution(<https://service.sap.com/sap/support/notes/1140831>)을 참조하십시오.

여러 계층구조에 대한 권한

특성 값에 여러 개의 계층구조가 있는 경우, 이러한 특성 각각에 대한 권한을 통해 모든 계층구조에서 모든 인증 값을 반환하도록 병합이 수행됩니다.

SAP Business Explorer(BEx)와 비교

SAP NetWeaver BW 데스크톱 응용 프로그램 BEx Query Designer 에서는 분석 권한을 기반으로 쿼리를 정의하고 필터를 추가할 수 있습니다. 그러나 BEx Query Designer 와 SAP BI 데이터 연합 서비스 간에는 약간의 차이가 있습니다.

- BEx Query Designer 를 사용할 경우, Query Designer 가 권한 필터(권한 변수 사용)를 적용할 특성을 선택적으로 지정할 수 있습니다. 데이터 연합 서비스를 사용할 경우, SAP NetWeaver BW 데이터 웨어하우징 워크벤치에서 권한 관련 특성으로 선언된 모든 특성에 권한 필터링이 적용됩니다.

- 데이터 연합 서비스를 사용할 경우, 지정된 특성에 대한 필터가 SQL 쿼리에 명시적으로 포함되어 있으면 이 특성에 대해 권한 필터링이 수행되지 않고 명시적 필터만 사용됩니다. BEx Query Designer 를 사용할 경우, 권한 필터와 사용자가 추가한 명시적 필터를 결합할 수 있습니다.

SAP NetWeaver BW 에 대한 SAP Note

SAP Note 1578089:DBIF: Adding authorizations to filter despite aggregation(<https://service.sap.com/sap/support/notes/1578089>)을 참조하십시오.



예

권한 필터링

BW 의 infocube ZCUBE1 은 6 개의 행만 포함하고 있으며 데이터 소스 BW_ZCUBE1 은 SSO 가 활성화된 상태로 액세스하도록 SAP Business Intelligence 플랫폼에 구성되어 있습니다. 전체 권한을 가진 사용자가 팩트 테이블을 읽는 경우 다음과 같은 SQL 쿼리를 실행합니다.

```
select ZCHA1, ZCHA2, ZCHA3, ZKYF1
from /DF_PROJECT/sources/BW_ZCUBE1/IZCUBE1
```

결과는 다음과 같습니다.

ZCHA1	ZCHA2	ZCHA3	ZKYF1
A	69226	2001	250.0
A	69226	2000	300.0
B	69190	2001	150.0
B	69190	2000	450.0
C	69115	2001	200.0
C	69115	2000	100.0

이제 다른 사용자 JOE 가 ZCHA1 및 ZCHA3 에 대한 전체 권한을 보유하고 있지만 ZCHA2 특성에 대해서는 단일 값 69190 에 액세스할 수 있다고 가정합니다. 이 사용자가 이 기능을 활성화하지 않고 동일한 SQL 쿼리를 실행하면 사용자 JOE 에게 InfoProvider ZCUBE1 에 대한 권한이 없음이라는 오류가 표시됩니다.

이 기능이 활성화되면 SAP Business Intelligence 플랫폼에서 인증된 모든 데이터가 반환되므로 위의 SQL 쿼리 결과는 다음과 같습니다.

ZCHA1	ZCHA2	ZCHA3	ZKYF1
B	69190	2001	150.0
B	69190	2000	450.0

실제로 결과는 ZCHA2 에서 명시적 필터를 사용한 SQL 의 결과와 같습니다.

```
select ZCHA1, ZCHA2, ZCHA3, ZKYF1
from /DF_PROJECT/sources/BW_ZCUBE1/IZCUBE1 where ZCHA2 = 69190
```

4.5.5 다중 소스 유니버스의 SAP NetWeaver BW 연결 아키텍처

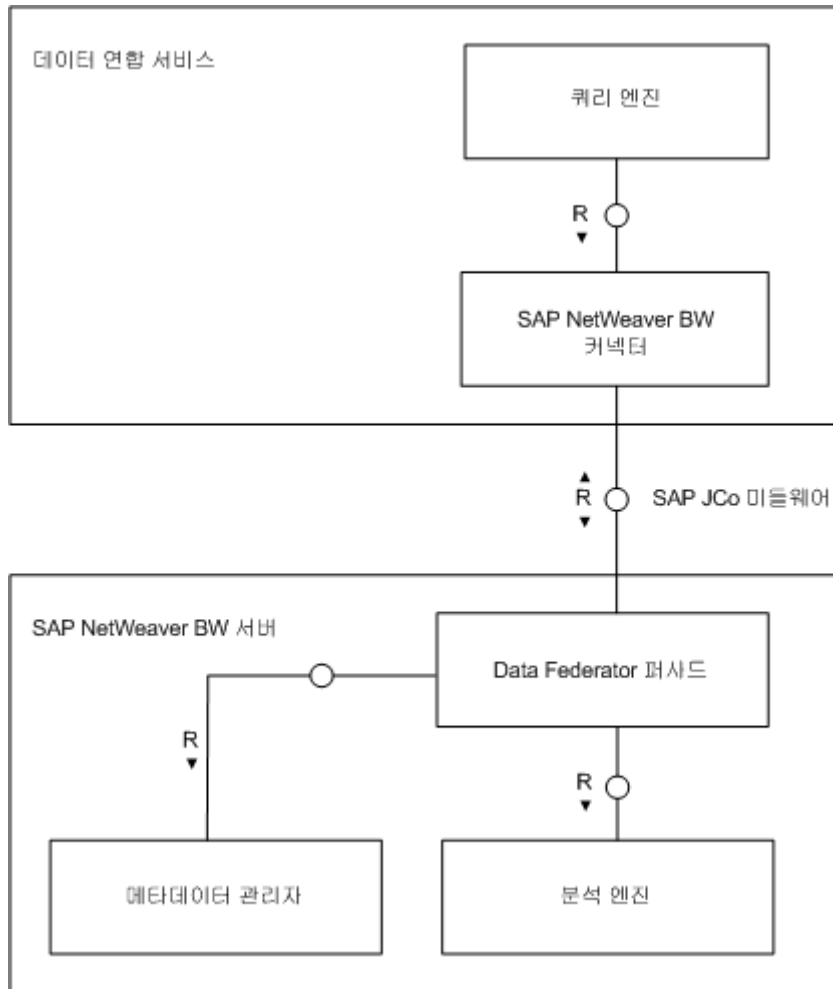
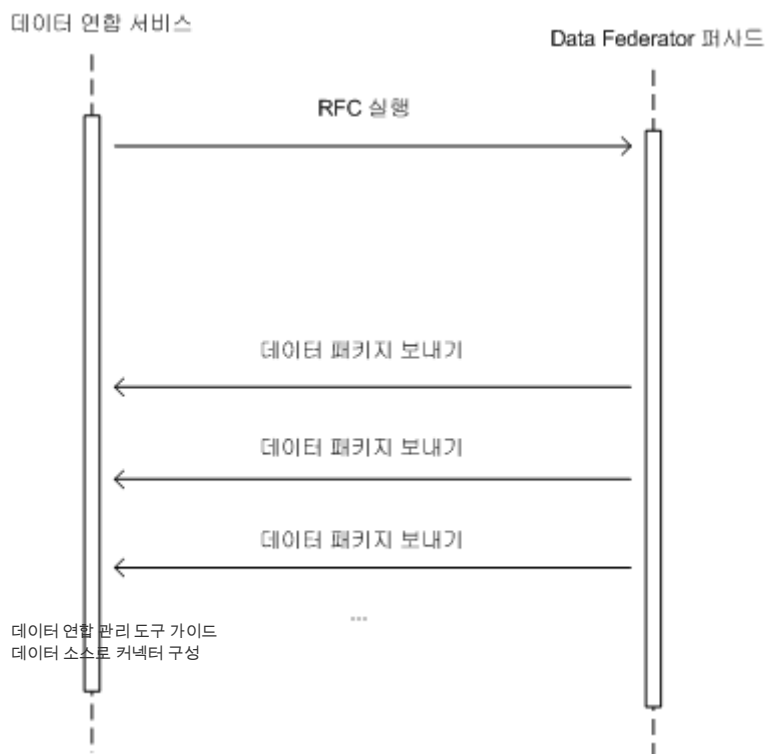


그림 2: 다중 소스 유니버스의 SAP NetWeaver BW 연결 아키텍처

4.5.6 다중 소스 유니버스의 SAP NetWeaver BW 연결 콜백 시퀀스



데이터 연합 쿼리 엔진에서 작업을 직접 실행하거나, 이 작업을 데이터 소스로 위임하도록 커넥터 기능을 설정할 수 있습니다.

일반적으로, 작업을 데이터베이스 시스템에 위임하는 것이 보다 효율적이지만 모든 데이터베이스 시스템이 동일한 연산자를 지원하는 것은 아닙니다. 기능 목록을 통해 데이터 연합 쿼리 엔진은 각 연산자를 위임할 수 있는 데이터 소스가 무엇인지 알 수 있습니다. 이 연산자 위임을 보통 푸시라고 합니다.

i 노트

관계형 커넥터 또는 SAS 커넥터의 기능만 설정할 수 있습니다.

1. 데이터 연합 관리 도구에서 **커넥터 구성** 탭을 클릭합니다.
2. 트리 목록에서 커넥터를 마우스 오른쪽 단추로 클릭한 다음 **구성 만들기**를 클릭합니다.
3. **구성 속성** 탭의 **기능** 행에서 **값** 셀을 클릭하고 기능을 **my-capability=true** 형식으로 입력합니다.

여러 기능이 세미콜론(;)으로 구분되는지 확인합니다. 대부분의 기능에서 true 또는 false 값을 사용할 수 있습니다.

관련 링크

[관계형 데이터 소스의 전체 커넥터 기능 목록](#) [페이지 52]

4.7 관계형 데이터 소스의 전체 커넥터 기능 목록

다음 표에는 커넥터의 기능이 나열되어 있습니다. 이 기능들은 기능 리소스 속성을 구성할 때 사용할 수 있습니다.

i 노트

fullsql 은 기본적으로 모든 기능을 true 로 설정할 수 있는 특수 기능입니다. 그런 다음 필요에 따라 개별 기능을 false 로 별도로 설정할 수 있습니다.

기능	설명
fullsql	모든 기능을 기본적으로 true 로 설정할 수 있습니다. 그런 다음 필요에 따라 개별 기능을 false 로 별도로 설정할 수 있습니다.
project	커넥터의 projection 연산 지원 여부를 지정합니다.
orderby	커넥터의 order by 연산 지원 여부를 지정합니다.
orderbystrings	문자열 열에서 커넥터의 order by 연산 지원 여부를 지정합니다.
distinct	커넥터의 distinct 연산 지원 여부를 지정합니다.
union	커넥터의 union distinct 연산 지원 여부를 지정합니다.
unionall	커넥터의 union all 연산 지원 여부를 지정합니다.
join	커넥터의 join 연산 지원 여부를 지정합니다.
outerjoin	커넥터의 full outer join 연산 지원 여부를 지정합니다.

기능	설명
leftouterjoin	커넥터의 left outer join 연산 지원 여부를 지정합니다.
rightouterjoin	커넥터의 right outer join 연산 지원 여부를 지정합니다.
aggregate	커넥터의 집계 지원 여부를 지정합니다.
aggregatedistinct	커넥터가 distinct 절이 있는 집계를 지원하는지 여부를 지정합니다.
minaggregate	커넥터의 min 집계 함수 지원 여부를 지정합니다.
maxaggregate	커넥터의 max 집계 함수 지원 여부를 지정합니다.
countaggregate	커넥터의 count 집계 함수 지원 여부를 지정합니다.
avgaggregate	커넥터의 average 집계 함수 지원 여부를 지정합니다.
sumaggregate	커넥터의 sum 집계 함수 지원 여부를 지정합니다.
minaggregatedistinct	커넥터가 distinct 절이 있는 min 집계 함수를 지원하는지 여부를 지정합니다.
maxaggregatedistinct	커넥터가 distinct 절이 있는 max 집계 함수를 지원하는지 여부를 지정합니다.
countaggregatedistinct	커넥터가 distinct 절이 있는 count 집계 함수를 지원하는지 여부를 지정합니다.
avgaggregatedistinct	커넥터가 distinct 절이 있는 average 집계 함수를 지원하는지 여부를 지정합니다.
sumaggregatedistinct	커넥터가 distinct 절이 있는 sum 집계 함수를 지원하는지 여부를 지정합니다.
equalitypredicate	커넥터의 equality 조건자 지원 여부를 지정합니다.
comparisonpredicate	커넥터의 inequality 조건자 지원 여부를 지정합니다.
likepredicate	커넥터의 like 조건자 지원 여부를 지정합니다.
nullpredicate	커넥터의 is null 조건자 지원 여부를 지정합니다.
inpredicate	커넥터의 in 조건자 지원 여부를 지정합니다.
arithmeticevaluation	커넥터의 산술 연산 지원 여부를 지정합니다.
booleanevaluation	커넥터의 부울 연산 지원 여부를 지정합니다.
constantevaluation	커넥터의 리터럴 지원 여부를 지정합니다.
emptystringevaluation	커넥터의 빈 문자열 리터럴 지원 여부를 지정합니다.
cancel	커넥터의 쿼리 실행 취소 지원 여부를 지정합니다.
shareconcurrentstatements	커넥터가 단일 연결에서 복수 쿼리(문)를 공유할 수 있는지를 지정합니다.
functionevaluation	커넥터가 함수 평가를 지원할지 여부를 지정합니다.

5 시스템 및 세션 매개 변수 관리

5.1 시스템 및 세션 매개 변수 정보

Data Federator에는 시스템과 세션의 두 가지 매개 변수 유형이 있습니다.

시스템 매개 변수는 실행 중인 Data Federator 쿼리 엔진 인스턴스에서 공유됩니다.

연결 하나마다 세션 매개 변수가 정의됩니다. 이 매개 변수 값은 연결마다 다를 수 있습니다.

각 세션 매개 변수는 이름이 같은 시스템 매개 변수의 기본값을 적용합니다. 세션 매개 변수에 해당하는 시스템 매개 변수의 값을 변경하는 경우 새 값은 새 세션에서만 적용됩니다.

시스템 및 세션 매개 변수를 사용하여 다음과 같은 데이터 연합 쿼리 엔진의 다양한 특성을 구성할 수 있습니다.

- 메모리 사용
- 네트워크 사용
- 쿼리 실행 순서
- 최적화

5.2 데이터 연합 관리 도구를 사용하여 시스템 매개 변수 변경

1. 매개 변수를 관리하기 위해 데이터 연합 관리 도구 인터페이스에 액세스하려면 데이터 연합 관리 도구에 로그인하고 **시스템 매개 변수** 탭을 클릭합니다.
2. 매개 변수를 포함하는 행에서 **현재 값** 상자에 새 값을 입력하고 **Enter** 키를 누릅니다.

5.3 데이터 연합 관리 도구를 사용하여 세션 매개 변수 변경

1. 매개 변수를 관리하기 위해 데이터 연합 관리 도구 인터페이스에 액세스하려면 데이터 연합 관리 도구에 로그인하고 **시스템 매개 변수** 탭을 클릭한 다음 **세션 매개 변수**를 클릭합니다.
2. 매개 변수를 포함하는 행에서 **현재 값** 상자에 새 값을 입력하고 **Enter** 키를 누릅니다.

5.4 데이터 연합 관리 도구를 사용하여 관계형 커넥터 및 SAS 커넥터 기능 설정

커넥터 기능에는 데이터 소스가 지원하는 연산자 종류 등과 같은 항목들이 포함됩니다.

데이터 연합 쿼리 엔진에서 작업을 직접 실행하거나, 이 작업을 데이터 소스로 위임하도록 커넥터 기능을 설정할 수 있습니다.

일반적으로, 작업을 데이터베이스 시스템에 위임하는 것이 보다 효율적이지만 모든 데이터베이스 시스템이 동일한 연산자를 지원하는 것은 아닙니다. 기능 목록을 통해 데이터 연합 쿼리 엔진은 각 연산자를 위임할 수 있는 데이터 소스가 무엇인지 알 수 있습니다. 이 연산자 위임을 보통 푸시라고 합니다.

i 노트

관계형 커넥터 또는 SAS 커넥터의 기능만 설정할 수 있습니다.

1. 데이터 연합 관리 도구에서 **커넥터 구성** 탭을 클릭합니다.
2. 트리 목록에서 커넥터를 마우스 오른쪽 단추로 클릭한 다음 **구성 만들기**를 클릭합니다.
3. **구성 속성** 탭의 **기능** 행에서 **값** 셀을 클릭하고 기능을 **my-capability=true** 형식으로 입력합니다.

여러 기능이 세미콜론(;)으로 구분되는지 확인합니다. 대부분의 기능에서 true 또는 false 값을 사용할 수 있습니다.

관련 링크

[관계형 데이터 소스의 전체 커넥터 기능 목록](#) [페이지 52]

5.5 시스템 매개 변수 목록

시스템 매개 변수	설명
NUM_PARTITIONS_FOR_DISTINCT_OPERATOR	distinct 연산자에 대해 산출하는 첫 번째 수준 파티션의 최적 개수. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 300
MAX_CONCURRENT_MEMORY_CONSUMING_QUERIES	최대 병렬 쿼리 개수. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 2
MAX_CONCURRENT_MEMORY_CONSUMING_OPERATORS	메모리를 사용하는 최대 동시 연산자 수. (이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다. 현재 서버를 다시 시작해야 합니다.) 유형: integer 재시작 필요 여부: 아니요 기본값: 5
EXECUTOR_STATIC_MEMORY	이 매개 변수는 초기화 시 연산자에 할당된 최소 메모리 공간을 나타냅니다. 이에 대해 정확한 값을 지정하거나(예: 'EXECUTOR_STATIC_MEMORY=50M') 실행기 메모리 크기 백분율(예: 'EXECUTOR_STATIC_MEMORY=25%')을 사용할 수 있습니다. 값을 지정할 때 해당 값은 실행기에 할당된 메모리 공간보다 작아야 하며 ('EXECUTOR_TOTAL_MEMORY' 매개 변수 참조), 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.

시스템 매개 변수	설명
	<p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 25%</p>
EXECUTOR_TOTAL_MEMORY	<p>이 매개 변수는 실행기에 할당된 메모리 공간을 나타냅니다. 메모리 크기 값(예: 'EXECUTOR_TOTAL_MEMORY=256M') 또는 JVM 에 의해 할당된 메모리 크기 백분율(예: 'EXECUTOR_TOTAL_MEMORY=80%')을 사용할 수 있습니다. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 80%</p>
EXECUTOR_BUFFER_OVERHEAD	<p>이 매개 변수는 쿼리 실행 중 생성될 수 있는 메모리 오버헤드를 나타냅니다. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 10%</p>
ACCEPT_MORE_CONCURRENT_QUERIES	<p>최대 개수의 동시 쿼리가 있는 경우 부울 값은 예외를 throw 하는 대신 대기 쿼리 목록을 유지해야 합니다(true). 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
MIN_BUFFER_PAGES_PER_OPERATOR	<p>연산자에 반환되는 최소 페이지 수입니다. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 8</p>
EXECUTOR_BUFFER_SIZE	<p>한 페이지 매개 변수의 크기(행 수). 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 128</p>
MAX_BUFFER_SHARE_PER_OPERATOR	<p>최대 동적 버퍼의 최대 공유입니다. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 80</p>

시스템 매개 변수	설명
SEMI_JOIN_DIMENSION_CACHE_MEMORY_SIZE	semi-join 에서 한 차원 캐시에 할당되는 메모리의 양(킬로바이트)입니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 1024
QUERY_HISTORY_SIZE	실행된 쿼리의 리포지토리에 대한 최대 기록 크기 유형: integer 재시작 필요 여부: 아니요 기본값: 10
MAX_SUBQUERIES_IN_HISTORY	기록의 쿼리 집합당 하위 쿼리의 최대 개수입니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 100
MAX_EXECUTIONS_PER_SUBQUERY_IN_HISTORY	기록에 보관된 하위 쿼리의 최대 실행 수입니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 10
MAX_ESTIMATED_SIZE_FOR_STRINGS_OR_DECIMALS	문자열 또는 10 진수 값에 대한 최대 추정 크기(바이트)입니다. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 512
MAX_PARTITIONS_FOR_HASH_OPERATORS	해시 알고리즘에 대해 산출하는 첫 번째 수준 파티션의 최대 개수. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 1987
MIN_SIZE_FOR_BUFFER_HASH_TABLE	Hash Join/Distinct 에 사용되는 버퍼 해시 테이블의 최소 크기입니다(각 항목에 대한 보조 해시). 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다. 유형: integer 재시작 필요 여부: 아니요 기본값: 200
MAX_TEMPORARY_TABLES	커넥터 하나에서 생성되는 고유한 임시 테이블의 최대 개수를 정의합니다. 유형: integer 재시작 필요 여부: 아니요

시스템 매개 변수	설명
	기본값: 1000
SEMI_JOIN_EXECUTION_STRATEGIES	<p>semi-join 연산자의 실행 전략 목록을 기본 설정 순서대로 지정합니다. 가능한 값은 기본 설정의 순서대로 쉼표로 구분된 I,T,P 의 조합으로서, I 는 IN 쿼리 실행 전략, T 는 임시 테이블 실행 전략, P 는 매개 변수화된 쿼리 실행 전략을 나타냅니다. 예: T,P,I,I,T P NONE 전략 없음 I,T,P 중 하나가 없는 경우에는 래퍼에서 해당 실행 전략을 지원하지 않습니다. 참고: NONE 은 래퍼가 지원하는 실행 전략이 없다는 의미입니다. 빈 문자열 또는 null 값은 지원되지 않습니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: I,T,P</p>
MAX_CONJUNCTIONS	<p>조건부의 최대 접속사 수입니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 512</p>
ACTIVATE_MULTI_THREADED_UNION_OPERATOR	<p>Union 연산자의 다중 스레드 구현 사용 여부를 지정합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: false</p>
ACTIVATE_ORDER_BASED_OPTIMIZATION_RULE	<p>true 로 설정된 경우 순서 기반 최적화를 수행하는 모든 규칙이 활성화됩니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
ACTIVATE_PROFITABILITY_BASED_JOIN_ORDERING_RULE	<p>true 로 설정된 경우 수익률을 기반으로 해시 트리를 작성하는 order join 규칙이 활성화됩니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
ACTIVATE_JOIN_DISTRIBUTION_RULE	<p>true 로 설정하면 join distribution 규칙을 활성화합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
ACTIVATE_JOIN_ELIMINATION_RULE	<p>true 로 설정하면 useless join elimination 규칙을 활성화합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: false</p>

시스템 매개 변수	설명
ACTIVATE_SEMI_JOIN_RULE	<p>true 로 설정하면 semijoin 을 생성하는 규칙을 활성화합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
ACTIVATE_SEMI_JOIN_DIMENSION_RUNTIME_CARDINALITY_LIMIT	<p>semi-join 에 대해 런타임에서 계산된 카디널리티 제한의 활성화 여부입니다. 이 옵션이 활성화된 경우 semi-join 차원의 런타임 카디널리티를 예상 값과 비교하고, 런타임 값이 예상 값보다 큰 경우 해당 차원을 삭제합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: true</p>
AVG_SIZE_OF_BUFFER_ROW	<p>한 행의 평균 추정 크기입니다. 이 매개 변수의 새 값은 BufferManager 에 등록된 쿼리가 없어야 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 256</p>
MIN_CARDINALITY_THRESHOLD_FOR_GROUP_BY_TRANSFORMATION_RULE	<p>소스 사용 순서에 의해 GroupBy 노드 제거를 결정하기 위한 고유 값의 최소 카디널리티. <p>0 은 GROUP BY 제거가 항상 수행되어야 함을 의미합니다.</p> <p>유형: long</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 300</p>
NUMBER_OF_PARTITIONS_FOR_MERGE_AGGREGATE_RULE	<p>MergeBasedGroupByAggregate 알고리즘에 사용되는 파티션 수. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 300</p>
MIN_CARDINALITY_FOR_ASYNC_PREFETCH	<p>비동기 프리페치를 결정하는 최소 카디널리티에 대한 매개 변수. -1 은 비동기 프리페치가 허용되지 않음을 의미합니다.</p> <p>유형: long</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 50000</p>
MAX_ORDERING_LIMIT_FOR_ORDER_JOINS_RULE	<p>유추 규칙 OrderJoinsRule 에 대한 매개 변수입니다. 이 값은 산출되는 최대 join ordering 수입니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 1</p>

시스템 매개 변수	설명
MAX_QUEUE_SIZE_LIMIT_FOR_ORDER_JOINS_RULE	<p>유추 규칙 OrderJoinsRule 에 대한 매개 변수입니다. 우선 순위 대기열의 최대 크기로, 검색 공간의 최대 크기를 정의합니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 1024</p>
DEFAULT_DECIMAL_PRECISION	<p>커넥터가 열에 대한 값을 반환하지 않는 경우 Data Federator 쿼리 서버에서 열의 소수점 정밀도에 대해 보고되는 값. 일반적인 상황에서 이 값은 항상 커넥터에 의해 제공됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 27</p>
DEFAULT_DECIMAL_SCALE	<p>커넥터가 열에 대한 값을 반환하지 않는 경우 Data Federator 쿼리 서버에서 열의 소수 자릿수에 대해 보고되는 값. 일반적인 상황에서 이 값은 항상 커넥터에 의해 제공됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 6</p>
MAX_DECIMAL_PRECISION	<p>쿼리 서버에 의해 열의 소수점 정밀도에 대해 보고되는 최대 값</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 40</p>
SCALE_FOR_MAX_DECIMAL_PRECISION	<p>데이터 연합 쿼리 엔진에 의해 최대의 정밀도로 열의 소수 자릿수에 대해 보고되는 값입니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 6</p>
MIN_DECIMAL_SCALE	<p>데이터 연합 쿼리 엔진에 의해 열의 소수 자릿수에 대해 보고되는 최소값입니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 6</p>
DEFAULT_STRING_SIZE	<p>커넥터가 열에 대한 값을 반환하지 않는 경우 Data Federator 쿼리 엔진에서 열의 문자열 크기에 대해 보고되는 값입니다. 일반적인 상황에서 이 값은 항상 커넥터에 의해 제공됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 40</p>

시스템 매개 변수	설명
MAX_STRING_SIZE	<p>데이터 연합 쿼리 엔진에 의해 열의 문자열 크기에 대해 보고되는 최대값입니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 255</p>
MAX_NUMBER_OF_FRACTIONAL_DIGITS_FOR_TOSTRING_DOUBLE_IN_LOCAL	<p>로컬 구분 함수 toStringL(double, varchar) 사용 시 double 의 문자열 표현에 나타나는 최대 소수 자릿수</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 20</p>
NUMBER_OF_PARTITIONS_FOR_HASH_JOIN_OPERATOR	<p>HashJoin/HashOuterJoin 알고리즘에 대한 첫 번째 수준 파티션의 최적 개수 추정치. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 300</p>
MAX_THREADS_IN_UNION_OPERATOR	<p>UNION 연산자에 의해 사용되는 최대 활성 스레드 수입니다. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 2</p>
ACTIVATE_OPTIMIZED_PREPARED_STATEMENTS	<p>쿼리가 모든 실행 또는 준비된 문의 각 실행에 대해 최적화되어 있는지 나타내는 부울 값입니다. True 로 설정된 경우 준비된 문의 모든 실행에 대해 같은 쿼리 계획을 사용합니다. False 로 설정된 경우 준비된 문의 각 실행에 대해 쿼리를 다시 최적화합니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: false</p>
NUMBER_OF_PARTITIONS_FOR_ORDER_AGGREGATE_RULE	<p>OrderBasedGroupByAggregate 알고리즘에 사용되는 파티션 수. 이 매개 변수의 새 값은 후속 쿼리에서 적용됩니다.</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 1987</p>
MAX_CARDINALITY_FOR_HOP_STORE_IN_HASH_JOIN_OPERATOR	<p>HOP 알고리즘 저장을 위한 최대 카디널리티 매개 변수</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 5000</p>

시스템 매개 변수	설명
MIN_STORE_CARDINALITY_THRESHOLD_FOR_ORDER_BASED_JOIN_RULE	<p>순서가 지정된 Merge Join 사용을 정렬하는 최소 저장소 크기 카디널리티에 대한 매개 변수</p> <p>유형: long</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 10000</p>
MIN_TRANSFER_CARDINALITY_THRESHOLD_FOR_MERGE_JOIN_RULE	<p>순서가 지정된 Merge Join 사용을 정렬하는 최소 전송 카디널리티에 대한 매개 변수</p> <p>유형: long</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 30000</p>
THREADPOOL_ACTION_ON_OUT_OF_MEMORY	<p>Memory Tracker 에서 Out of Memory 조건이 발생할 경우 취해지는 조치를 설정하는 특수 매개 변수입니다. 선택 가능한 조치는 freeze, kill&freeze, cancel running, cancel all, none 입니다. freeze 를 선택하면 관리된 쿼리를 실행하는 모든 스레드가 고정됩니다. 이 작업을 통해 특수 외부 도구를 사용하여 가상 시스템 상태를 확인할 수 있습니다. kill&freeze 를 선택하면 현재 실행되고 있는 관리된 쿼리가 중지됩니다(현재 스레드가 중지되고 쿼리가 취소된 후 닫힘). 이 작업을 통해 프로파일러가 작동할 수 있도록 메모리를 확보할 수 있으므로 서버의 상태가 일관되지 않아 쿼리를 더 이상 실행할 수 없는 상황을 방지할 수 있습니다. 쿼리가 취소되면 서버가 가상으로 고정되며 관리된 쿼리를 더 이상 실행할 수 없습니다. cancel running 을 선택하면 현재 관리된 쿼리 및 서버에서 현재 실행된 조치에 대한 쿼리가 모두 취소됩니다. 이 작업을 통해 메모리를 복구하고 서버를 실행 상태로 유지할 수 있습니다. cancel all 을 선택하면 쿼리가 모두 취소됩니다. 취소 실행 시에는 현재 실행 중인 쿼리가 문제의 원인이고 내부 쿼리 서버 오류가 아닐 경우에만 메모리를 확보할 수 있습니다. 참고: 이제부터 ThinDriver 나 원격 서버 연결에서 전송된 쿼리가 모두 관리된 쿼리로 표시됩니다. 관리 콘솔이나 간단한 텍스트 콘솔에서는 관리된 쿼리가 사용되지 않으므로 쿼리가 명시적으로 고정되지 않습니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: cancel all</p>
ACTIVATE_FREEZE_WHEN_OUT_OF_MEMORY	<p>OutOfMemory 가 발생할 경우 취해지는 조치를 설정하는 특수 매개 변수입니다. 설정하면 시스템의 모든 관리된 스레드가 고정됩니다. 관리되지 않는 스레드를 통해 일부 동작은 계속 실행될 수 있지만 시스템 상태는 트러스트할 수 없습니다. 설정하지 않으면 java 프로세스가 중단되고 시스템이 종료됩니다.</p> <p>유형: 부울</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: false</p>
DEFAULT_LOCALE	<p>로캘에 대한 ISO 로캘 코드를 정의합니다. 이 시스템 매개 변수는 세션 매개 변수 LOCALE 의 기본값입니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: en_US</p>
DEFAULT_SORT	<p>sort 데이터 정렬을 정의합니다. 이 시스템 매개 변수는 세션 매개 변수 SORT 의 기본값입니다.</p>

시스템 매개 변수	설명
	<p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 이진</p>
DEFAULT_COMP	<p>comp 데이터 정렬을 정의합니다. 이 시스템 매개 변수는 세션 매개 변수 COMP 의 기본값입니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 이진</p>
OPTIMIZER_COMPUTE_BINDINGS_PARAMETER	<p>ResolveBindings 규칙 속성의 매개 변수를 정의합니다 - 0: 규칙 비활성화, 1: bind join 으로 해결, 2: 캐시 노드로 해결 ...</p> <p>유형: integer</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 2</p>
MIN_SOURCE_CARDINALITY_THRESHOLD_FOR_SEMI_JOIN_RULE	<p>semi join 연산자를 활성화하는 데 필요한 큰 테이블에 대한 카디널리티 임계값을 지정합니다.</p> <p>유형: long</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 15000</p>
MIN_ACTIVATION_THRESHOLD_FOR_SEMI_JOIN_RULE	<p>전체 테이블 검사와 비교하여 Data Federator 에서 semi join 이 유용하다고 고려되는 경우 semi join 에서 반환되는 튜플의 분수입니다. 너무 많은 값을 검색해야 하는 경우 semi join 이 도움이 되지 않을 수도 있으며, 이때 Data Federator 는 테이블 검사를 대신 실행합니다. 예를 들어 테이블에 10M 개의 행이 있는 경우, 최소 활성화 임계값을 1000 으로 설정하면 $10M / 1000 = 10\,000$ 이 됩니다. semi join 을 실행하기 위해 10 000 개 미만의 행을 반입하는 것으로 계산되면 Data Federator 에서 semi join 연산자를 사용합니다. semi join 이 사용되는 빈도를 줄이려면 이 값을 높이고, semi join 이 더 자주 사용되게 하려면 이 값을 낮춥니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 100</p>
MAX_ACTIVATION_LIMIT_FOR_PUSH_AGGREGATE_RULE	<p>소스에 "Group-By" 푸시 없는 초기 카디널리티와 비교하여 Data Federator 에서 "Group-By" 푸시가 유용하다고 고려되는 경우 소스에 "Group-By" 연산자를 푸시하여 반환되는 행의 분수입니다. 너무 많은 값을 검색해야 하는 경우에는 "Group-By" 푸시가 유용하지 않습니다. 예를 들어, 매개 변수가 = 80% (0.80)로 설정되고 "Group-By" 푸시로 인한 새 카디널리티가 초기 카디널리티의 80%를 넘는 경우 "Group-By"가 소스에 푸시되지 않습니다. "Group-By" 연산자가 보다 자주 푸시되도록 하려면 이 값을 높이고 "Group-By" 연산자의 사용 빈도를 줄이려면 이 값을 낮춥니다.</p> <p>유형: 문자열</p> <p>재시작 필요 여부: 아니요</p> <p>기본값: 1.00</p>

시스템 매개 변수	설명
CLUSTER_SYNCHRONIZE_DELAY	두 개의 동기화 이벤트 간 시간(초)을 정의합니다. 결합 허용 모듈에서는 동기화 이벤트를 사용하여 리소스 수정 작업 중 서버 오류로 인해 다른 클러스터 멤버에 통보되지 않은 리포지토리의 수정 사항을 식별합니다. 유형: long 재시작 필요 여부: 아니요 기본값: 3600

관련 링크

[semi-join 실행 전략](#) [페이지 29]

5.6 세션 매개 변수 목록

세션 매개 변수	설명
CATALOG	쿼리에 카탈로그가 지정되지 않은 경우 사용되는 현재 카탈로그를 정의합니다.
SCHEMA	쿼리에 스키마가 지정되지 않은 경우 사용되는 현재 스키마를 정의합니다.
COMP	문자열 비교에 사용되는 데이터 정렬을 정의합니다. SQL 쿼리에서 문자열이 비교되는 방식을 정의하는 데 사용됩니다. 이 매개 변수의 값은 지원되는 데이터 정렬 값 중 하나이거나 키워드 LINGUISTIC입니다. 이 경우 사용된 데이터 정렬은 SORT 매개 변수에서 정의된 데이터 정렬입니다. 기본값은 BINARY입니다. 기본값을 시스템 매개 변수 DEFAULT_COMP로 변경할 수 있습니다.
SORT	문자열 정렬에 사용되는 데이터 정렬을 정의합니다. SQL 쿼리에서 문자열이 정렬되는 방식을 정의하는 데 사용됩니다. 이 매개 변수 값은 지원되는 데이터 정렬 값 중 하나입니다. 기본값은 BINARY입니다. 기본값을 시스템 매개 변수 DEFAULT_SORT로 변경할 수 있습니다.
LOCALE	로캘에 대한 ISO 코드를 정의합니다. 기본값은 en_US입니다. 기본값을 시스템 매개 변수 DEFAULT_LOCALE로 변경할 수 있습니다.
DATA_LOCALE	데이터에 사용할 로캘을 정의합니다. 이 매개 변수는 지역화된 데이터를 반환할 수 있는 커넥터(현재 SAP NetWeaver BW 커넥터)에 의해 사용됩니다.

관련 링크

[데이터 연합 응용 프로그램의 데이터 정렬](#) [페이지 64]

[시스템 매개 변수 목록](#) [페이지 55]

5.7 데이터 연합 응용 프로그램의 데이터 정렬

데이터 정렬은 데이터 정렬 및 비교 방식을 결정하는 일련의 규칙입니다.

데이터 연합 응용 프로그램 및 여기에서 액세스하는 데이터베이스 시스템은 올바른 문자 시퀀스를 정의하는 규칙을 사용하여 문자 데이터를 정렬하고 비교합니다. 대부분의 데이터베이스 시스템에서 데이터베이스 시스템이 대/소문자, 악센트 표시, 전자/반자 또는 카나 문자 유형을 구분해야 할지 여부를 지정하는 옵션을 구성할 수 있습니다.

대/소문자 구분

시스템에서 M 문자를 m 문자와 동일하게 간주하는 경우 대/소문자가 구분되지 않는 것입니다. 컴퓨터에서는 입력값을 구별하기 위해 ASCII 코드를 사용하므로 M 과 m 이 다르게 간주됩니다. M 의 ASCII 값은 77 이지만, m 의 ASCII 값은 109 입니다.

악센트 구분

시스템에서 a 문자와 á 문자를 동일하게 간주하는 경우 악센트가 구분되지 않는 것입니다. 컴퓨터에서는 입력값을 구별하기 위해 ASCII 코드를 사용하므로 a 와 á 가 다르게 간주됩니다. a 의 ASCII 값은 97 이지만, á 의 ASCII 값은 225 입니다.

카나 구분

일본어 카나 문자인 히라가나와 가타카나가 다르게 간주되는 경우 카나가 구분되는 것입니다.

전자/반자 구분

싱글바이트 문자(반자)와 더블바이트 문자(전자)로 표시될 때도 동일하게 표현되는 문자가 다르게 간주되는 경우 전자/반자가 구분되는 것입니다.

관련 링크

[데이터 연합 응용 프로그램에서 지원되는 데이터 정렬](#) [페이지 65]

[이진 데이터 정렬 사용 시 데이터 연합 응용 프로그램에서 소스에 쿼리를 푸시하는 방법을 결정하는 방식](#) [페이지 68]

5.7.1 데이터 연합 응용 프로그램에서 지원되는 데이터 정렬

다음과 같은 데이터 정렬이 DF 에서 지원됩니다.

이진	유니코드 이진 정렬(또는 유니코드 이진 정렬과 호환 가능 - ASCII 문자 집합 정렬은 유니코드 문자 집합 정렬과 호환 가능)
locale_AI_CI	로캘, 악센트 구분 안 함, 대/소문자 구분 안 함
locale_AS_CI	로캘, 악센트 구분, 대/소문자 구분 안 함
locale_AS_CI	로캘, 악센트 구분, 대/소문자 구분 안 함
locale_AI_CS	로캘, 악센트 구분 안 함, 대/소문자 구분
locale_AS_CS	로캘, 악센트 구분, 대/소문자 구분

여기서 로캘은 LN_CY 로 정의됩니다.

- LN - ISO 언어 코드(예: <en>)
- CY - ISO 국가 코드(예: <US>)

i 노트

모든 DF 데이터 정렬은 일본어 가나 구분 안 함 및 전자/반자 구분 안 함입니다.



예

<en_US_AS_CI> - 영어, 미국, 악센트 구분, 대/소문자 구분

관련 링크

[데이터 연합 응용 프로그램의 데이터 정렬](#) [페이지 64]

5.7.2 데이터 연합 SQL 쿼리에 대해 문자열 정렬 및 문자열 비교 동작 설정

sort 및 comp 매개 변수를 사용하여 데이터 연합 쿼리 엔진에서 문자열 정렬 및 비교를 처리하는 방식을 설정할 수 있습니다.

sort 매개 변수는 문자열이 데이터 연합 쿼리 엔진에 의해 정렬되는 방식을 정의하는 데 사용됩니다. sort 매개 변수 값은 지원되는 데이터 정렬 값 중 하나입니다. 기본값은 이진입니다.

comp 매개 변수는 문자열이 SQL 쿼리에서 비교되는 방식을 정의하는 데 사용됩니다. comp 매개 변수 값은 다음 중 하나입니다.

- 지원되는 데이터 정렬 값 중 하나
- Linguistic 키워드: 이 경우 사용된 데이터 정렬은 sort 매개 변수에서 정의된 데이터 정렬입니다.

sort 및 comp 매개 변수는 세션 매개 변수, 시스템 매개 변수 또는 사용자 계정의 속성으로 정의할 수 있습니다.

- sort 또는 comp 매개 변수가 세션 매개 변수로 정의되어 있는 경우 이 값이 현재 연결에 사용됩니다.
- 세션 매개 변수로 정의되어 있지 않은 경우 사용자 계정의 sort 또는 comp 속성이 현재 연결에 사용됩니다.
- 현재 사용자 계정의 속성으로 정의되어 있지 않은 경우 sort 또는 comp 시스템 매개 변수가 현재 연결에 사용됩니다.

sort 및 comp 매개 변수 값은 문자열 값에 적용된 SQL 연산의 결과에 영향을 줍니다. 이 연산은 함수이거나 GROUP BY, ORDER BY 와 같은 SQL 연산자이거나 T.A < e 와 같은 필터 식일 수 있습니다. 아래 표는 comp 및 sort 매개 변수의 대/소문자를 구분하는 SQL 연산자를 요약한 것입니다.

SQL 식	대소문자 구분
=, !=, >, <=, >=	comp 대/소문자 구분
BETWEEN, NOT BETWEEN	comp 대/소문자 구분
CASE	comp 대/소문자 구분
DISTINCT	comp 대/소문자 구분

SQL 식	대소문자 구분
GROUP BY	comp 대/소문자 구분
HAVING	comp 대/소문자 구분
IN, NOT IN	comp 대/소문자 구분
LIKE, NOT LIKE	대/소문자 구분 안 함: 이진만 해당
ORDER BY	sort 대/소문자 구분
UNION ALL	대/소문자 구분 안 함

SQL 함수	대소문자 구분
MAX, MIN	comp 대/소문자 구분
데이터 연합 문자열 함수	대/소문자 구분 안 함: 이진만 해당

예

```
SELECT LASTNAME, count(*)
FROM EMPLOYEE E
WHERE SALARY < 5000 AND DEPARTMENT_NAME =
    <Sales>
GROUP BY LASTNAME
```

표 1: 직원 테이블

성	이름	급여	부서 이름
Smith	John	6000	Sales
Sm Ith	Jo	4000	Sales
Smith	John	2000	Sa Les
Smith	Albert	7000	Sales

comp 매개 변수가 <en_US_AS_CS>인 경우 결과는 다음과 같습니다.

Smith	3
Sm Ith	1

comp 매개 변수가 <en_US_AI_CI>인 경우 결과는 다음과 같습니다.

Smith	4
-------	---

관련 링크

[데이터 연합 응용 프로그램의 데이터 정렬 \[페이지 64\]](#)

[데이터 연합 응용 프로그램에서 지원되는 데이터 정렬 \[페이지 65\]](#)

5.7.3 이진 데이터 정렬 사용 시 데이터 연합 응용 프로그램에서 소스에 쿼리를 푸시하는 방법을 결정하는 방식

데이터 연합 쿼리 엔진의 옵티마이저는 데이터 소스에 SQL 작업을 푸시다운할 수 있는지 여부를 결정하기 위해 푸시다운 분석을 수행합니다.

데이터 정렬이 이진일 경우 쿼리 엔진은 데이터 소스의 SQL 기능만 검사하여 특정 데이터 소스에 하위 쿼리를 푸시할지 여부를 결정합니다.

따라서 일반적인 경우 쿼리 엔진은 기본 데이터 소스가 데이터 연합 응용 프로그램의 이진 데이터 정렬과 호환되는 기본 데이터 정렬을 사용하는 것으로 간주합니다.

SQLServer, MySQL 및 Oracle 에 대해서만 소스의 기본 데이터 정렬이 이진 정렬과 호환되지 않은 경우에도 데이터 연합 쿼리 엔진이 이진 데이터 정렬을 사용하도록 강제할 수 있습니다. 이진 데이터 정렬에 대한 리소스 매개 변수를 구성하는 방법과 관련된 자세한 내용은 MySQL, SQLserver, Oracle 설명서를 참조하십시오.

관련 링크

[데이터 연합 응용 프로그램의 데이터 정렬 \[페이지 64\]](#)

[데이터 연합 SQL 쿼리에 대해 문자열 정렬 및 문자열 비교 동작 설정 \[페이지 66\]](#)

[데이터 연합 응용 프로그램에서 지원되는 데이터 정렬 \[페이지 65\]](#)

6 SQL 구문 참조

6.1 데이터 연합 쿼리 엔진용 쿼리 언어

가능한 경우 데이터 연합 응용 프로그램은 표준 SQL-92 구문에 맞춥니다. 하지만, 데이터 연합 쿼리 엔진의 문에서 일부 요소가 사용되거나 영향을 미치는 방식을 이해하는 것이 중요합니다. 이 단원에서는 개체 관리, 데이터 형식, select 및 식을 포함하여 데이터 연합 응용 프로그램에 의해 구현된 SQL-92의 요소를 설명합니다.

6.1.1 식별자 및 명명 규칙

테이블을 포함한 스키마와 카탈로그를 제공하여 테이블을 참조합니다. 카탈로그, 스키마 및 테이블은 점(.)으로 구분해야 합니다.



예

테이블의 이름 정의

정규화된 이름을 사용하여 테이블을 참조해야 합니다. 정규화된 이름은 카탈로그 이름, 스키마 이름 및 테이블 이름으로 구성됩니다.

- c.s.t
- "c"."s"."t"

카탈로그 또는 스키마가 기본적으로 정의되어 있는 경우, 테이블에 대한 참조에서 카탈로그 또는 스키마의 이름을 생략할 수 있습니다.

관련 링크

[큰따옴표 구분 기호 사용](#) [페이지 71]

[기본 카탈로그 및 스키마](#) [페이지 70]

6.1.1.1 카탈로그

카탈로그는 명명된 스키마 그룹입니다. 카탈로그의 이름은 그 이름에 속한 스키마의 이름을 정규화합니다. 쿼리에서 카탈로그 이름을 명시적으로 밝히거나 기본 카탈로그를 설정합니다.

관련 링크

[기본 카탈로그 및 스키마](#) [페이지 70]

6.1.1.2 스키마

SQL 스키마는 테이블 또는 뷰의 명명된 그룹입니다. 스키마는 카탈로그에 따라 다릅니다. 스키마 이름은 스키마가 속한 카탈로그 내에서 고유해야 합니다.

스키마 식별자는 기본 카탈로그가 설정되어 있지 않을 때 절대 경로이거나, 기본 카탈로그 디렉터리에서의 상대 경로입니다.

데이터 연합 관리 도구에서 세션 매개 변수를 통해 기본 스키마를 설정할 수 있습니다.

관련 링크

[기본 카탈로그 및 스키마](#) [페이지 70]

6.1.1.3 테이블

테이블은 한 스키마에 연결됩니다. 테이블 이름은 테이블이 속한 스키마 내에서 고유해야 합니다.

테이블은 카탈로그 이름, 스키마 이름 및 테이블 이름으로 식별되어야 합니다. 표준 SQL 구문에서는 .(마침표)로 구분되는 카탈로그 이름, 스키마 이름 및 테이블 이름을 연결하여 테이블 식별자를 구성합니다.

기본 카탈로그 및/또는 기본 스키마가 설정되어 있을 때, 테이블 식별자에서 카탈로그 이름과 스키마 이름을 생략할 수 있습니다.

관련 링크

[기본 카탈로그 및 스키마](#) [페이지 70]

6.1.1.4 열

테이블은 열의 집합으로 설명됩니다. 열 이름은 열이 속한 테이블 내에서 고유해야 합니다. 표준 SQL 구문에서는 마침표 "."로 구분되는 열 이름과 테이블 식별자를 연결하여 열 식별자를 구성합니다.

6.1.1.5 기본 카탈로그 및 스키마

데이터 연합 관리 도구에서 세션 매개 변수를 통해 기본 카탈로그 또는 스키마를 지정할 수 있습니다. 기본 카탈로그를 지정하면 테이블 이름을 정규화하지 않고 쿼리를 보낼 수 있습니다.

테이블을 참조하려면	기본 카탈로그가 다음인 경우	그리고 기본 스키마가 다음인 경우	정규화된 이름 사용
c.s.t	c		s.t
"c1".s.t	"c1"		s.t
c.s.t	c	s	t

관련 링크

[데이터 연합 관리 도구를 사용하여 시스템 매개 변수 변경](#) [페이지 54]

6.1.1.6 큰따옴표 구분 기호 사용

파서에서 식별자를 잘못 해석하지 않도록 하려면, 카탈로그, 스키마, 테이블 및 열 이름에 영숫자 문자 이외의 문자가 포함되어 있을 때 이들 이름에 대해 큰따옴표 구분 기호를 사용해야 합니다.

올바른 표시	"c1/c2"."sche+ma"."Tab-le1".col1
잘못된 표시	/c1/c2.sche+ma.Tab-le1.col1

참조 정보는 [개체 식별자 및 숫자 상수](#) [페이지 77]를 참조하십시오.

6.1.2 데이터 연합 쿼리 엔진에 사용된 데이터 형식

데이터 연합 쿼리 엔진에서는 각각의 열, 로컬 변수, 식 및 매개 변수에 관련 데이터 형식이 있습니다. 데이터 형식은 정수 데이터, 문자 데이터, 날짜 및 시간 데이터 또는 10 진수 데이터와 같이, 개체가 보유할 수 있는 데이터의 크기와 구조에 대한 정의입니다.

개체와 관련된 데이터 형식은 해당 개체의 다음 3 가지 특성을 정의합니다.

- 데이터 형식: 개체에 포함된 데이터의 종류
- 길이 및 크기: 값의 길이 또는 크기
- 배율 및 정밀도: 숫자의 배율과 정밀도(숫자 데이터 형식만 해당)

기존의 데이터베이스에서는 길이, 정밀도 및 배율이 저장된 값의 속성을 정의하므로 열을 만들 때 이들을 설정합니다. 데이터 연합 쿼리 엔진은 가상 데이터베이스이며 어떤 값도 저장하지 않습니다. 따라서 길이, 정밀도 및 배율은 스키마 정의 시 정의되지 않습니다. 그 값은 사용 중인 소스 테이블에서 동적으로 유추됩니다.

관련 링크

[알려진 데이터 형식](#) [페이지 71]

[식에 유추 입력](#) [페이지 73]

[배율과 정밀도](#) [페이지 74]

6.1.2.1 알려진 데이터 형식

데이터 연합 쿼리 엔진은 java.sql.Types 에 정의된 표준 SQL 형식을 지원합니다. 다음은 지원되는 데이터 형식 목록입니다.

- BIT
- DATE
- TIMESTAMP
- TIME
- INTEGER
- DOUBLE
- DECIMAL
- VARCHAR

- NULL

모든 데이터베이스가 같은 데이터 형식을 사용하거나 같은 방식으로 해석하는 것은 아니기 때문에, 쿼리 엔진은 공통 데이터베이스 형식과 쿼리 엔진 사이의 매핑을 표준화했습니다.

관련 링크

<http://java.sun.com/j2se/1.3/docs/api/java/sql/Types.html>

6.1.2.2 JDBC 데이터 형식에 데이터 연합 쿼리 엔진 유형 매핑

다음 표에는 데이터 연합 쿼리 엔진에서 사용하는 내부 데이터 형식과 데이터 연합 JDBC 드라이버에서 반환되는 JDBC 데이터 형식 사이의 대응 관계가 자세히 설명되어 있습니다.

데이터 연합 데이터 형식	JDBC 데이터 형식
BIT	BIT
DATE	DATE
TIMESTAMP	TIMESTAMP
TIME	TIME
INTEGER	INTEGER
DOUBLE	DOUBLE
DECIMAL	DECIMAL
VARCHAR	VARCHAR
NULL	NULL

6.1.2.3 JDBC 데이터 형식에서 데이터 연합 데이터 형식으로 매핑

JDBC 데이터 소스에 액세스할 때, 데이터 연합 쿼리 엔진은 JDBC 드라이버에서 반환되는 JDBC 형식을 내부 데이터 연합 응용 프로그램 데이터 형식으로 매핑합니다. 다음 표에는 JDBC 데이터 형식과 매핑에 사용되는 데이터 연합 형식 사이의 대응 관계가 자세히 설명되어 있습니다.

JDBC 데이터 형식	데이터 연합 데이터 형식
TINYINT, SMALLINT, INTEGER, DECIMAL(정밀도 <= 10 및 배율 = 0)	INTEGER
BIT	BIT
REAL, FLOAT, DOUBLE	DOUBLE

JDBC 데이터 형식	데이터 연합 데이터 형식
BIGINT, DECIMAL, NUMERIC	DECIMAL
VARCHAR, LONGVARCHAR, CHAR	VARCHAR
DATE	DATE
TIME	TIME
TIMESTAMP	TIMESTAMP
NULL 및 다른 모든 JDBC 형식	NULL

6.1.2.4 날짜 및 시간 변환

데이터 연합 쿼리 엔진은 날짜를 '1970-01-01'로 설정하여 TIME 데이터를 TIMESTAMP 데이터로 변환합니다.



시간을 타임스탬프로 변환

TIME '12:01:01'은 TIMESTAMP '1970-01-01 12:01:01.0'으로 변환됩니다.

데이터 연합 쿼리 엔진은 시간 00:00:00.000000000 을 추가하여 DATE 데이터를 TIMESTAMP 로 변환합니다.



날짜를 타임스탬프로 변환

DATE '1999-01-01'은 TIMESTAMP '1999-01-01 00:00:00.000000000'으로 변환됩니다.

6.1.2.5 식에 유추 입력

두 식에 다른 데이터 형식이 있을 때는 데이터 형식의 우선 순위를 적용하여 이 두 식을 산술 연산자와 결합하는 식의 결과 데이터 형식을 결정합니다.

데이터 연합 쿼리 엔진은 형식 간에 다음과 같은 우선 적용 순서를 사용합니다.

```

NULL
VARCHAR
INTEGER
DOUBLE
DECIMAL

```

6.1.2.6 배율과 정밀도

식 결과의 길이, 배율 및 정밀도는 결과의 형식으로부터 유추됩니다. VARCHAR 또는 DECIMAL 결과 형식의 경우, 길이, 배율 및 정밀도는 입력 식과 입력 식을 결합한 함수 및 연산자의 배율과 정밀도로부터 유추됩니다.

아래 표에는 모든 데이터 연합 식에 대한 벡터(길이, 정밀도, 배율)가 나와 있습니다.

업 형식	고정 한계(길이, 정밀도, 배율)
BIT	(1, 1, 0)
INTEGER	(11, 10, 0)
DOUBLE	(22, 15, 0)
DATE	(10, 0, 0)
TIMESTAMP	(29, 9, 0)
TIME	(8, 0, 0)
NULL	(0, 0, 0)
DECIMAL	유추됨
VARCHAR	정밀도와 배율은 항상 (0, 0)임 길이는 유추됨

6.1.3 문

SQL 쿼리를 작성하여 데이터 연합 쿼리 엔진에 저장된 데이터를 검색하거나 조작할 수 있습니다. 쿼리는 다음의 몇 가지 형태로 작성할 수 있습니다.

- 데이터 연합 관리 도구 - 데이터 연합 쿼리 엔진 상단에 있는 그래픽 사용자 인터페이스(GUI)
- 명령줄 SQL 응용 프로그램
- SELECT 문을 작성할 수 있는 다른 호환 가능한 유틸리티
- 클라이언트 또는 중간 계층 기반 응용 프로그램(예: Microsoft Visual Basic 응용 프로그램) - SQL Server 테이블의 데이터를 바인딩된 컨트롤(예: 모눈)로 매핑 가능

6.1.3.1 SELECT 문

쿼리는 다양한 방법으로 사용자와 상호 작용하지만 수행하는 작업은 모두 같습니다. 즉, SELECT 문의 결과 집합을 사용자에게 표시합니다.

SELECT 문은 데이터 연합 쿼리 엔진에서 데이터를 검색하여 하나 이상의 결과 집합으로 사용자에게 반환합니다. 결과 집합은 SELECT 문의 데이터를 표 형식으로 정렬한 것입니다. 결과 집합도 SQL 테이블과 마찬가지로 열과 행으로 구성되어 있습니다.

SELECT 문의 전체 구문은 복잡하지만 대부분의 SELECT 문은 결과 집합에 대해 다음 네 가지 기본 속성을 기술합니다.

- 결과 집합의 열 개수와 특성
- 데이터를 제공하는 테이블의 이름
- 소스 테이블 행이 SELECT 에 적합하기 위해 충족되어야 하는 조건. 이 조건을 충족시키지 못하는 행은 무시됩니다.
- 결과 집합 행이 정렬되는 순서



SELECT 문

다음 SELECT 문은 단가가 40 달러를 초과하는 제품의 제품 ID 와 이름, 정가를 검색합니다.

```
SELECT <ProductID>, <Name>, <ListPrice>
FROM <Production.Product>
WHERE <ListPrice> > <$40>
ORDER BY <ListPrice> ASC
```

- SELECT 절
SELECT 키워드 다음에 나열된 열 이름(<ProductID>, <Name> 및 <ListPrice>)은 SELECT 목록을 구성합니다. 이 목록은 결과 집합에 세 개의 열이 있고, 각 열마다 FROM 절에 지정된 테이블(<Product 테이블>)에서 연결된 열의 이름, 데이터 유형 및 크기가 있다고 지정합니다. FROM 절은 한 개의 테이블만 지정하기 때문에 SELECT 문의 모든 열 이름은 해당 테이블의 열을 참조합니다.
- FROM 절
FROM 절은 <Product> 테이블을 데이터를 검색할 테이블로 나열합니다.
- WHERE 절
WHERE 절은 이 SELECT 문에 적합한 <Product> 테이블의 행만이 <ListPrice> 열의 값이 <\$40>를 초과하는 행이라는 조건을 지정합니다.
- ORDER BY 절
ORDER BY 절은 <ListPrice> 열의 값에 따라 결과 집합을 오름차순(ASC)으로 정렬하도록 지정합니다.

6.1.3.2 데이터 연합 쿼리 엔진에서 지원되는 SQL-92 문

데이터 연합 쿼리 엔진은 데이터 조작 언어(DML)와 프로시저 및 명령 목록을 지원합니다. 지원되는 SELECT 문의 특정 집합은 특별히 언급되지 않는 한, 전체 표준 SQL-92 구문입니다. 특히 외부 조인에 대한 SQL-92 문법과 외부 조인에 대한 JDBC 구문이 모두 지원됩니다.

관련 링크

[SELECT 절의 문법](#) [페이지 78]

6.1.4 식

이 단원에서는 데이터 연합 SQL 구문의 식에 대해 자세히 설명합니다.

6.1.4.1 식에 사용되는 함수

최근 함수 목록을 확인하려면 정보 디자인 도구 사용자 가이드 문서에서 다중 소스 사용 유니버스의 SAP BusinessObjects SQL 함수 참조를 확인하십시오.

6.1.4.2 식에 사용되는 연산자

식에 사용되는 연산자는 하나 또는 그 이상의 간단한 식을 결합하여 더 복잡한 식을 형성합니다.

연산자 이름	설명
+(더하기)	숫자 형식에 대해서는 "더하기"를, VARCHAR 형식에 대해서는 "연결"을 의미하는 산술 연산자.
-(빼기)	"빼기"를 의미하는 산술 연산자.
*(곱하기)	"곱하기"를 의미하는 산술 연산자.
/(나누기)	"나누기"를 의미하는 산술 연산자.
%(모듈로)	산술 연산자. 이 연산자는 나누기의 나머지(정수)를 반환합니다. 예를 들어, 12 를 5 로 나눈 나머지는 2 이므로 $12 \% 5 = 2$ 입니다.
** (거듭제곱)	산술 연산자. 이 연산자는 지정된 거듭제곱에 대해 지정된 식의 값을 반환합니다.
=(같음)	"같음"을 의미하는 비교 연산자.
>(보다 큼)	"보다 큼"을 의미하는 비교 연산자.
<(보다 작음)	"보다 작음"을 의미하는 비교 연산자.
>=(크거나 같음)	"크거나 같음"을 의미하는 비교 연산자.
<=(작거나 같음)	"작거나 같음"을 의미하는 비교 연산자.
<>(같지 않음)	"같지 않음"을 의미하는 비교 연산자.
ALL	비교 집합이 전부 TRUE 인 경우 TRUE 인 논리 연산자.
AND	두 BOOLEAN 식이 모두 TRUE 인 경우 TRUE 인 논리 연산자.
ANY	비교 집합 중 어느 하나가 TRUE 인 경우 TRUE 인 논리 연산자.

연산자 이름	설명
BETWEEN	피연산자가 어떤 범위 내에 있는 경우 TRUE 인 논리 연산자.
EXISTS	하위 쿼리에 행이 포함된 경우 TRUE 인 논리 연산자.
IN	피연산자가 목록에 있는 식 중 하나와 같은 경우 TRUE 인 논리 연산자.
LIKE	피연산자가 어떤 패턴과 일치하는 경우 TRUE 인 논리 연산자.
NOT	다른 BOOLEAN 연산자의 값을 반전하는 논리 연산자.
OR	어느 한 BOOLEAN 식이 TRUE 인 경우 TRUE 인 논리 연산자.
SOME	비교 집합 중 일부가 TRUE 인 경우 TRUE 인 논리 연산자.
+(양수)	숫자 값이 양수인 단항 연산자.
-(음수)	숫자 값이 음수인 단항 연산자.

6.1.4.3 연산자 우선 순위 수준

복잡한 식에 연산자가 여러 개 있을 때, 연산자 우선 순위에 따라 연산 수행 순서가 결정됩니다. 실행 순서는 결과 값에 큰 영향을 미칠 수 있습니다.

연산자 우선 순위 수준은 다음과 같습니다. 우선 순위 수준이 높은 연산자가 수준이 낮은 연산자보다 먼저 적용됩니다.

- +(양수), -(음수)
- *(곱하기), /(나누기), %(모듈로), **(거듭제곱)
- +(더하기), (+ 연결), -(빼기)
- =, >, <, >=, <=, <>(비교 연산자)
- NOT
- AND
- OR
- ALL, ANY, BETWEEN, IN, LIKE, SOME

6.1.4.4 개체 식별자 및 숫자 상수

식별자와 상수의 이름은 영문자로 시작해야 하고 영문자와 밑줄만 사용해야 합니다. 하지만, 큰따옴표(")로 둘러싸는 경우에는 식별자/상수 이름에 어떤 문자든 사용할 수 있습니다.

예를 들어, 식별자 이름으로 ABC_12 또는 "%any name you like\$#%"를 사용할 수 있습니다.

다음 표에는 식별자 및 숫자 상수에 대한 데이터 연합 구문이 나와 있습니다.

입력 대상	사용할 정의	예를 들면, 다음과 같습니다.
정수	INTEGER: nnn(하나 또는 그 이상의 숫자만 사용)	12 14 15
이중 또는 소수	DOUBLE/ DECIMAL: nn.nn(하나 이상의 숫자 뒤에 점을 쓰고, 그 뒤에 하나 이상의 숫자를 씀)	12.3 13.222 11.3
날짜	DATE: {d 'yyyy-mm-dd'}	{d '2005-03-28'}
시간	TIME: {t 'hh:mm:ss'}	{t '01:10:12'}
타임스탬프	TIMESTAMP: {ts 'yyyy-mm-dd hh:mm:ss.ffff'}	{ts '2005-03-28 01:11:34.23222'}
문자열 또는 가변 길이 문자열 (Varchar)	작은따옴표 내부의 모든 문자열	'asdasdas'
간단한 식별자	영문자로 시작하고 그 뒤에는 영문자, 숫자 및 밑줄로 구성된 임의의 조합으로 된 모든 문자열	ABC_12
특수 문자를 포함한 식별자	큰따옴표 내부의 모든 문자열	"!%any name you like\$#\$\$"

6.1.5 설명

SQL 문에 주석을 추가하려면 텍스트 앞에 이중 하이픈(--) 또는 파운드 기호(#)를 사용합니다. 주석은 해당 줄의 끝에서 끝납니다.

6.2 SELECT 절의 문법

다음 단원에서는 데이터 연합 쿼리 엔진과 함께 사용되는 전체 SQL Select 절 문법을 자세히 설명합니다.

```

start      :=  ( query ) ( ";" )? <EOF>

query      :=  ( <WITH> withListElement ( "," withListElement ) * )?
              SQLSelectFromWhere (
                ( <UNION> | <INTERSECT> | <EXCEPT> ) ( <DISTINCT> | <ALL> )?
                SQLSelectFromWhere QueryExpression )?
                ( <ORDER> <BY> orderByTerms ( "," orderByTerms ) * )?

```

```

QueryExpression :=
    ( ( <UNION> | <INTERSECT> | <EXCEPT> ) ( <DISTINCT> | <ALL> )?
      SQLSelectFromWhere ) *

withListElement := anyIdentifier <AS> ( WITHView | nativeQuery )

WITHView      := "(" query ")"

nativeQuery := <NATIVE> "("
    dataSourceIdentifier ","
    nativeQueryStatement ","
    columnSpecificationList
    ( "," paramSpecificationList )? ")"

dataSourceIdentifier := anyIdentifier

nativeQueryStatement := quotedString

columnSpecificationList := columnSpecification ( "," ( columnSpecification ) ) *
paramSpecificationList := paramSpecification ( "," ( paramSpecification ) ) *
columnSpecification := anyIdentifier columnDataType

paramSpecification := ( ( ( ( <DATE_LITERAL> | <TIME_LITERAL> |
    <TIMESTAMP_LITERAL> ) )
    | quotedString ) columnDataType )
    | <NULL_LITERAL>

columnDataType := identifier ( "(" integerLiteral ( "," integerLiteral )? ")" )?
integerLiteral := <INT_LITERAL>

SQLSelectFromWhere :=
    <SELECT> ( <DISTINCT> )? ( selectExpression ( "," selectExpression ) * |
    ( <MULT> ) )
    ( fromClause
    ( <WHERE> disjunction )?
    ( <GROUP> <BY> ( additiveTerm ) ( "," additiveTerm ) * )?
    ( <HAVING> disjunction )? )

fromClause := ( <FROM> tableReferenceList )

tableReferenceList := ( tableReference ( "," tableReference ) * )

tableReference := tableReferenceAtomicTerm ( qualifiedJoinPart ) *

tableReferenceAtomicTerm := ( tablePrimary
    | jdbcOuterJoin
    | "(" query ")" ( ( <AS> )? ( identifier | delimitedIdentifier ) )?
    | "(" tableReference ")" ( ( <AS> )?
    identifier ( "(" projectAlias ( "," projectAlias ) * ")" )? )? )

tablePrimary := ( table ( ( <AS> )? ( tableAlias ) )? )

table := ( anyIdentifier ( "." anyIdentifier ( "." anyIdentifier )? )? )

qualifiedJoinPart := ( ( <NATURAL> )? ( joinType )?
    <JOIN> tableReferenceAtomicTerm ( joinSpecification )? )

jdbcOuterJoin := "{" <OUTER_JOIN_JDBC> jdbcOuterJoinPart "}"

jdbcOuterJoinPart := tableReferenceAtomicTerm
    ( outerJoinType <OUTER> <JOIN> ( jdbcOuterJoinPart ) joinSpecification )?

joinType := ( ( <INNER> ) | ( <CROSS> ) | ( outerJoinType ( <OUTER> )? ) )

outerJoinType := ( <LEFT> | <RIGHT> | <FULL> )

```

```

joinSpecification := ( joinCondition | namedColumnsJoin )
joinCondition    := ( <ON> disjunction )
namedColumnsJoin := ( <USING> "(" addUsing ( "," addUsing )* ")" )
addUsing         := columnName
projectAlias     := ( anyIdentifier )
selectExpression := ( ( tableStar )
    | ( disjunction ( ( <AS> )? anyIdentifier )? ) )
tableStar        := table "." <MULT>
functionTermJdbc := ( "{" <FUNCTION_JDBC> (
    ( identifier )
    | ( <LEFT> )
    | ( <RIGHT> ) ) "(" ( disjunction ( "," disjunction )* )? ")" "}" )
functionTerm     := ( (
    ( identifier ) |
    ( <LEFT> )
    | ( <RIGHT> ) )
    "(" ( ( <DISTINCT> | <ALL> )?
    ( disjunction ( "," disjunction )* | <MULT> ) )? ")" )
analyticFunctionPart := ( <OVER> "("
    ( <PARTITION> <BY> ( variable ) ( "," variable )* )?
    <ORDER> <BY> ( ( variable ( <ASC> | <DESC> )? ) )
    ( "," ( variable ( <ASC> | <DESC> )? ) )* ")" )
disjunction := ( conjunction ( <OR> conjunction )* )
conjunction := ( negationTerm ( <AND> negationTerm )* )
escapeChar  := quotedString
quotedString := <QUOTED_STRING_LITERAL>
anyIdentifier := <IDENTIFIER>
    | <DELIMITED_IDENTIFIER>
delimitedIdentifier := <DELIMITED_IDENTIFIER>
identifier := <IDENTIFIER>
columnName := anyIdentifier
negationTerm := ( <NOT> )? ( ( comparisonTerm ) | ( <EXISTS> "(" query ")" ) )
comparisonTerm := additiveTerm ( <COMPARISON_OPERATOR> (
    ( additiveTerm )
    | ( ( ( <ANY> ) | ( <SOME> ) | ( <ALL> ) ) "(" query ")" ) )
    | ( <BETWEEN> additiveTerm <AND> additiveTerm )
    | ( inValuesOrQuery )
    | <LIKE> additiveTerm ( <ESCAPE> escapeChar )?
    | <IS> ( <NULL_LITERAL> | <NOT> <NULL_LITERAL> )
    | <NOT> (
        <BETWEEN> additiveTerm <AND> additiveTerm
        | <LIKE> additiveTerm ( <ESCAPE> escapeChar )? ) )?
nativeExpression := <NATIVE> <EXPRESSION> "("
    dataSourceIdentifier ","
    columnDataType ","
    quotedString bindingArgumentList ")"

```

```

bindingArgumentList := ( "," additiveTerm ) *

inValuesOrQuery := ( ( <NOT> )? <IN> "(" ( inValues ) | ( query ) ) ")" )

inValues      := ( signedConstant ( "," signedConstant ) * )

additiveTerm   := ( factor ( ( <PLUS> | <MINUS> ) factor ) * )

factor := unaryTerm ( (
    <MULT>
  | <DIVIDE>
  | <POWER>
  | <INT_DIVIDE>
  | <MOD> ) unaryTerm ) *

unaryTerm := atomicTerm
           | <PLUS> atomicTerm
           | <MINUS> atomicTerm

variable   := ( anyIdentifier
  ( "." anyIdentifier
    ( "." anyIdentifier
      ( "." anyIdentifier )? )? )? )

variableFullName := anyIdentifier (
  "." anyIdentifier
    ( "." anyIdentifier
      ( "." anyIdentifier )? )? )?

constant := <BOOL_LITERAL>
          | <INT_LITERAL>
          | <FLOAT_LITERAL>
          | <SCIENTIFIC_NOTATION_LITERAL>
          | <DATE_LITERAL>
          | <TIMESTAMP_LITERAL>
          | <TIME_LITERAL>
          | <NULL_LITERAL>
          | quotedString
          | <PARAMETER>

signedConstant := <BOOL_LITERAL>
                 | ( <PLUS> | <MINUS> )? ( <INT_LITERAL> | <FLOAT_LITERAL> )
                 | <SCIENTIFIC_NOTATION_LITERAL>
                 | <DATE_LITERAL>
                 | <TIMESTAMP_LITERAL>
                 | <TIME_LITERAL>
                 | <NULL_LITERAL>
                 | quotedString
                 | <PARAMETER>

atomicTerm := functionTerm ( analyticFunctionPart )?
            | functionTermJdbc
            | variable
            | constant
            | "(" disjunction ")"
            | caseExpression
            | coalesceExpression
            | castExpression
            | convertFunction
            | nativeExpression

caseExpression := ( <CASE> ( ( additiveTerm ( (
    <WHEN> additiveTerm <THEN> additiveTerm ) + )
  | ( ( <WHEN> disjunction <THEN> additiveTerm ) + ) )
  ( <ELSE> additiveTerm )? <END> )

coalesceExpression := ( <COALESCE> "(" additiveTerm ( ( "," additiveTerm ) ) +

```

```

)" )

castExpression := ( <CAST> "(" disjunction <AS> identifier ")" )

convertFunction := ( <CONVERT> "(" disjunction "," identifier ")" )

tableAlias := ( delimitedIdentifier | identifier )

orderByTerms := ( variableFullName | integerLiteral ) ( <ASC> | <DESC> )?

bindingFunction := ( variable <COMPARISON_OPERATOR> additiveTerm )

startStoredProcedure := ( procedureCall ) ( ";" )? <EOF>

procedureCall := <CALL> anyIdentifier ( ( "(" procedureArguments ")" )
| ( procedureArguments ) )

procedureArguments := ( procedureArgument ( "," procedureArgument )* )?

procedureArgument := ( procedureConstant )
| ( <CAST> "(" procedureConstant <AS> identifier ")" )

procedureConstant := (
    <BOOL_LITERAL>
  | <INT_LITERAL>
  | <FLOAT_LITERAL>
  | <SCIENTIFIC_NOTATION_LITERAL>
  | <DATE_LITERAL>
  | <TIMESTAMP_LITERAL>
  | <TIME_LITERAL>
  | <NULL_LITERAL>
  | quotedString
  | <PARAMETER> )

<DEFAULT> TOKEN [IGNORE_CASE] : {
<FROM: "from">
| <SELECT: "select">
| <DISTINCT: "distinct">
| <WHERE: "where">
| <GROUP: "group">
| <ORDER: "order">
| <BY: "by">
| <HAVING: "having">
| <DESC: "desc">
| <ASC: "asc">
| <AS: "as">
| <UNION: "union">
| <INTERSECT: "intersect">
| <EXCEPT: "except">
| <WITH: "with">
| <USING: "using">
| <ON: "on">
| <MERGE: "merge">
| <MERGING: "merging">
| <NATIVE: "native">
| <EXPRESSION: "expression">
| <NATURAL: "natural">
| <JOIN: "join">
| <CROSS: "cross">
| <INNER: "inner">
| <OUTER: "outer">
| <LEFT: "left">
| <RIGHT: "right">
| <FULL: "full">
| <ESCAPE: "escape">
| <OUTER_JOIN_JDBC: "oj">
| <FUNCTION_JDBC: "fn">
| <OVER: "over">

```

```

| <PARTITION: "partition">
| <CASE: "case">
| <WHEN: "when">
| <THEN: "then">
| <ELSE: "else">
| <END: "end">
| <COALESCE: "coalesce">
| <CALL: "call">
| <CAST: "cast">
| <CONVERT: "convert">
}

<DEFAULT> TOKEN [IGNORE_CASE] : {
    <NULL_LITERAL: "null">
}

<DEFAULT> TOKEN [IGNORE_CASE] : {
    <BOOL_LITERAL: "true" | "false">
}

<DEFAULT> TOKEN [IGNORE_CASE] : {
    <AND: "and">
    | <OR: "or">
    | <IN: "in">
    | <ANY: "any">
    | <SOME: "some">
    | <ALL: "all">
    | <EXISTS: "exists">
    | <BETWEEN: "between">
    | <COMPARISON_OPERATOR: ">" | ">=" | "<" | "<=" | "=" | "<>">
    | <LIKE: "like">
    | <NOT: "not">
    | <MULT: "*">
    | <PLUS: "+">
    | <MINUS: "-">
    | <DIVIDE: "/">
    | <INT_DIVIDE: "//">
    | <POWER: "***">
    | <MOD: "%">
    | <IS: "is">
    | <PARAMETER: "?">
}

<DEFAULT> SPECIAL : {
    <SINGLE_LINE_COMMENT: ("#" | "--") (~["\n", "\r"])* ("\n" | "\r" | "\r\n")*>
}

<DEFAULT> TOKEN : {
    <INT_LITERAL: ([ "0"-"9" ])+>

    | <FLOAT_LITERAL: ([ "0"-"9" ])+ "." ([ "0"-"9" ])+>

    | <SCIENTIFIC_NOTATION_LITERAL: ("-" | "+")? ([ "0"-"9" ])+ ( "." ([ "0"-"9" ])+ )?
      | ( "." ([ "0"-"9" ])+ ) ("e" | "E") ("-" | "+")? ([ "0"-"9" ])+>

    | <DATE_LITERAL: "{" (" ")* "d" (" ")* "\" <DIGIT> <DIGIT> <DIGIT> <DIGIT>
      "-" <DIGIT> <DIGIT> "-" <DIGIT> <DIGIT> "\" (" ")* "}">

    | <TIME_LITERAL: "{" (" ")* "t" (" ")* "\" <DIGIT> <DIGIT>
      ":" <DIGIT> <DIGIT> ":" <DIGIT> <DIGIT> "\" (" ")* "}">

    | <TIMESTAMP_LITERAL: "{" (" ")* "ts" (" ")* "\"
      <DIGIT> <DIGIT> <DIGIT> <DIGIT> "-" <DIGIT> <DIGIT> "-" <DIGIT> <DIGIT> " "
      <DIGIT> <DIGIT> ":" <DIGIT> <DIGIT> ":" <DIGIT> <DIGIT>

```

```

    ("." (<DIGIT>)*)? "\" (' ')* ">

| <DELIMITED_IDENTIFIER: "\" (~["\"", "\n", "\r"] | "\"\"")* "\">

| <QUOTED_STRING_LITERAL: "\" (' ' (~["\""] | "\"'\"")* "\">

| <IDENTIFIER: <LETTER> (<LETTER> | <DIGIT>)*>

| <#URLCHAR: [":", "?", ".", "/", "@", "_", "-", "+", "%", "!"]>

| <#LETTER: ["$", "A"-"Z", "_", "a"-"z",
    "\u00c0"-" \u00d6", "\u00d8"-" \u00f6", "\u00f8"-" \u00ff", "\u0100"-" \u024f",
    "\u0370"-" \u052f", "\u0530"-" \u05ff", "\u0600"-" \u06ff", "\u0900"-" \u10ff",
    "\u1100"-" \u11f9", "\u1e00"-" \u1ef9", "\u0100"-" \u1fff", "\u3040"-" \u319f",
    "\u3200"-" \u32fe", "\u3300"-" \u33fe", "\u3400"-" \u3d2d", "\u4e00"-" \u9fff",
    "\uac00"-" \ud7a3", "\uf900"-" \ufa2d", "\ufb00"-" \ufb4f", "\ufb50"-" \ufdfb",
    "\ufe70"-" \ufefc", "\uff00"-" \uffff"]>

| <#DIGIT: ["0"-"9", "\u0660"-" \u0669", "\u06f0"-" \u06f9", "\u0966"-" \u096f",
    "\u09e6"-" \u09ef", "\u0a66"-" \u0a6f", "\u0ae6"-" \u0aef", "\u0b66"-" \u0b6f",
    "\u0be7"-" \u0bef", "\u0c66"-" \u0c6f", "\u0ce6"-" \u0cef", "\u0d66"-" \u0d6f",
    "\u0e50"-" \u0e59", "\u0ed0"-" \u0ed9", "\u1040"-" \u1049"]>
}

```


7 용어집

7.1 용어 및 설명

이 단원에서는 데이터 연합 응용 프로그램 및 설명서에서 사용되는 용어에 대해 설명합니다.

용어 또는 구문	정의
커넥터	데이터 연합 쿼리 엔진의 데이터 소스 연결을 가능하게 하는 드라이버
팬아웃	열 간의 관계에서 첫 번째 열의 각 항목과 관련된 두 번째 열의 평균 항목 수입입니다.
<i>merge join</i>	데이터 연합에서 사용되는 연산으로, 테이블을 조인하는 데 걸리는 시간을 단축시키기 위해 두 개의 큰 데이터 테이블을 조인하기 전에 정렬하는 것입니다.
푸시(동사)	데이터 연합 엔진 내에서 작업을 수행하는 대신 소스 데이터베이스에 일부 작업 수행을 요청하는 것입니다. 일반적으로 소스 데이터베이스에서 작업을 수행하는 것이 데이터 연합 엔진에서 작업을 수행하는 것보다 효율적입니다.
<i>semi-join</i>	두 테이블 간의 연산으로, 두 번째 테이블에 있는 하나 이상의 행과 일치하는 첫 번째 테이블의 행을 반환합니다. 즉, 두 번째 테이블의 행을 사용하여 첫 번째 테이블이 필터링됩니다.
통계	데이터 연합에 사용되는 소스에 저장된 데이터의 숫자 정보입니다. 테이블의 예상 항목 수, 열의 예상 고유 값 수 또는 열 간 각 값의 평균 관계 수 등이 통계에 해당합니다.

8 문제 해결

8.1 데이터 연합 서비스 로그 정보

데이터 연합 서비스는 SAP BusinessObjects Enterprise 플랫폼의 적응 처리 서버에서 호스트합니다.

서비스를 호스트하는 적응 처리 서버에서 데이터 연합 서비스에 대한 로그를 찾을 수 있습니다.

SAP BusinessObjects Enterprise 관리자 가이드에서 SAP BusinessObjects Enterprise 서버의 로깅에 대한 문서를 참조하십시오.

8.2 SAP NetWeaver BW 데이터 소스에서 장기 실행 쿼리로 인한 연결 종료

SAP NetWeaver BW 데이터 소스에서 쿼리를 10 분 이상 실행할 경우 아무런 메시지 없이 연결이 끊어집니다.

원인

SAP NetWeaver BW 의 기본 제한 시간 값이 너무 짧아 쿼리를 실행할 수 없습니다.

조치

1. 다음과 같이 제한 시간 값을 늘립니다.
2. SAP 시스템에 로그인합니다.
3. 트랜잭션 텍스트 필드에 rz11 을 입력한 다음 실행합니다.
4. rdisp/max_wprun_time 매개 변수를 표시합니다.
5. **값 변경**을 클릭하고 매개 변수를 600 보다 큰 값으로 설정하여 보고서를 실행합니다.
값이 초 단위로 설정됩니다.

8.3 SAP NetWeaver BW 커넥터의 NoClassDefFoundError: CpicDriver 오류

NoClassDefFoundError: com.sap.conn.rfc.driver.CpicDriver 예외가 발생합니다.

원인

이 예외는 SAP Java Connector(JCo)의 종속성이 호스트에 설치되어 있지 않아서 발생할 수 있습니다. JCo 는 SAP NetWeaver 연결을 위해 데이터 연합 쿼리 엔진에서 사용하는 미들웨어입니다. 누락된 종속성은 Microsoft Visual Studio 2005 C/C++ 런타임 라이브러리의 집합입니다.

조치

Microsoft Visual Studio 2005 C/C++ 런타임 라이브러리를 설치합니다. 자세한 내용은 <https://service.sap.com/sap/support/notes/684106> 을 참조하십시오.

www.sap.com/contactsap

© 2013 SAP AG 및 SAP 계열사. 모든 권리 보유.

본 발행물의 어떠한 부분도 SAP AG의 명시적 허가 없이는 어떠한 형태나 목적으로도 재생산 또는 배포할 수 없습니다. 본 문서의 정보는 사전 예고 없이 변경될 수 있습니다.

SAP AG 및 그 유통업자가 판매하는 일부 소프트웨어 제품에는 다른 소프트웨어 공급업체가 소유한 소프트웨어 구성 요소가 포함되어 있습니다. 국가별 제품 명세는 다를 수 있습니다.

본 문서는 SAP AG 및 계열사("SAP 그룹")에 정보 전달 목적으로만 제공되며 어떠한 것도 진술하거나 보증하지 않습니다. SAP 그룹은 본 문서의 오류나 누락 부분에 대한 책임을 지지 않습니다. SAP 그룹 제품 및 서비스 대한 유일한 보증은 해당 제품 및 서비스와 함께 제공되는 보증서에 명시된 내용으로 제한됩니다. 본 문서의 어떤 내용도 추가 보증의 근거로 해석할 수 없습니다.

SAP 및 본 문서에서 언급된 기타 SAP 제품, 서비스와 해당 로고는 독일 및 기타 국가에서 사용되는 SAP AG의 상표 또는 등록 상표입니다.

추가 상표 정보 및 통지는 <http://www.sap.com/corporate-en/legal/copyright/index.epx>에서 확인하십시오.