IBM SPSS Analytic Server
버전 2

사용자 안내서

IBM

참고

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 반드시 39 페이지의 『주의사항』의 정보를 읽으십시오.

제품 정보

이 개정판은 새 개정판에 별도로 명시하지 않는 한 IBM SPSS Analytic Server의 2, 릴리스 0, 수정사항 0 및 모든 후속 릴리스와 수정에 적용됩니다.

목차

제 1 장 버전 2 사용자를 위한 새로운 사항

Analytic Server 콘솔

새 레이아웃

레이아웃이 변경되어 아코디언이 아닌 홈 페이지에서 페이지에 액세스할 수 있습니다.

데이터 소스

- 데이터 소스의 사용자 정의 속성을 정의하고 다른 애플리케이션에서 작성한 사용자 정의 속성을 볼 수 있습니다.
- 데이터 소스의 메타데이터를 작성할 때 모든 데이터 값의 스캔을 시작하여 카테고리 값과 범위 한계를 판별할 수 있습니다. 모든 데이터 값을 스캔하면 메타데이터가 올바른지 확인할 수 있지만 데이터 소스에 여러 필드와 레코드가 있으면 시간이 다소 걸릴 수 있습니다.
- 추가 데이터 소스 유형에 대한 지원이 있습니다.

  파일 컨텐츠 유형

  추가 파일 컨텐츠 유형의 지원에는 추가 설정 및 구문 분석기 형식이 포함됩니다. 데이터 소스에서 각 파일의 필드에 대해 구문 분석된 순서를 정의할 수도 있습니다. 디렉토리를 데이터 소스에 추가할 때 해당 디렉토리 또는 서브디렉토리 내에 있는 파일을 선택하는 규칙을 지정할 수 있습니다.

  부분 구조화 파일

  웹 로그와 같이 구분된 텍스트 파일의 구조보다 적은 구조가 있지만 정규식을 통해 레코드 및 필드로 추출할 수 있는 데이터가 포함된 파일이 있습니다.

  압축 파일

  지원되는 압축 형식으로는 Gzip, Deflate, Bz2, Snappy 및 IBM CMX가 있습니다. 또한 이전에 설명한 압축 형식을 사용하는 시퀀스 파일이 지원됩니다.

  다른 형식의 텍스트 기반 파일

  단일 텍스트 기반 데이터 소스에 이제 텍스트 분석을 위해 다른 형식(PDF, Microsoft Word 등)의 문서가 포함될 수 있습니다.

  SPSS Statistics 파일

  SPSS Statistics 파일(*.sav, *.zsav)은 데이터 모델이 포함된 2진 파일입니다.

  분할 가능한 2진 형식 파일(*.asbf)

  이 파일 유형이 Analytic Server의 결과인 경우가 있습니다. 예를 들어, 분석에 목록 값이 있는 필드를 사용해야 하는 경우입니다.

시퀀스 파일

시퀀스 파일(*.seq)은 키/값 쌍으로 구조화된 텍스트 파일입니다. 이 파일은 일반적으로 MapReduce 작업에서 중개 형식으로 사용됩니다.

데이터베이스 컨텐츠 유형

Analytic Server가 Greenplum, MySQL 및 Sybase IQ를 사용할 수 있도록 구성된 경우 이를 데이터 소스로 정의할 수 있습니다.

HCatalog 컨텐츠 유형

Analytic Server가 Apache Cassandra, MongoDB 및 Oracle NoSQL을 사용할 수 있도록 구성된 경우 이를 데이터 소스로 정의할 수 있습니다.

Geospatial 컨텐츠 유형

shape 파일 또는 온라인 맵 서비스를 사용하여 데이터 소스를 지리로 정의할 수 있습니다.

분석

새 SPSS Modeler 기능

병합 순위 지정된 조건에 따라 병합할 수 있도록 지원이 추가되었습니다.

시계열 시계열 처리와 TCM(temporal causal model)의 분산 작성 및 스코어링에 대한 지원이 추가되었습니다. SPSS Modeler의 AS 시간 구간, 스트리밍 TCM 및 TCM 노드를 확인하십시오.

공간 데이터

지리 좌표계 처리와 GSAR(geospatial association rules) 및 STP(spatio-temporal point process) 모델의 분산 작성 및 스코어링에 대한 지원이 추가되었습니다. SPSS Modeler에서 리프로젝션, 연관 규칙, STP 노드를 참조하십시오.

클러스터링

2단계 클러스터 모델의 분산 작성과 스코어링에 대한 지원이 추가되었습니다. SPSS Modeler의 TwoStep-AS 노드를 참조하십시오.

기존 SPSS Modeler 기능에 대한 지원이 개선되었습니다.

집계 문자열 필드는 널이 아닌 값의 최소값, 최대값 및 개수를 사용하여 집계할 수 있습니다. 최적화 탭의 숫자 필드에 대략의 순서 통계(중앙값, 사분위수)가 지원됩니다.

병합 키 없이 키로 병합 및 조건으로 병합을 위한 지원이 추가되었습니다. 예를 들어, 글로벌 평균을 생성하는 경우입니다.

앙상블 모델링

트리, 선형 및 신경망 모형을 위한 앙상블 모형의 작성 알고리즘이 개선되어 균일한 크기의 블록에서 임의로 분산되어 있지 않은 데이터를 더 잘 처리할 수 있습니다.

제 2 장 Analytic Server 콘솔

Analytic Server는 데이터 소스와 프로젝트를 관리하는 데 사용하는 씬 클라이언트 인터페이스를 제공합니다.

로그인

1. 브라우저의 주소 표시줄에서 Analytic Server의 URL을 입력하십시오. 이 URL은 서버 관리자에게서 얻을 수 있습니다.
2. 서버에 로그온하는 데 사용할 사용자 이름을 입력하십시오.
3. 지정된 사용자 이름과 연관된 비밀번호를 입력하십시오.

로그인 후에는 콘솔 홈이 표시됩니다.

콘솔 탐색

- 헤더에는 제품 이름, 현재 로그인되어 있는 사용자 이름, 도움말 시스템 링크가 표시됩니다. 현재 로그인된 사용자의 이름은 로그아웃 링크가 포함된 드롭 다운 목록의 머리글입니다.
- 컨텐츠 영역에는 콘솔 홈에서 수행할 수 있는 조치가 표시됩니다.

데이터 소스

데이터 소스는 분석할 데이터 세트를 정의하는 레코드 콜렉션에 데이터 모델이 추가된 것입니다. 레코드의 소스는 HDFS의 파일(구분된 텍스트, 고정 너비 텍스트, EXCEL), 데이터베이스 또는 HCatalog일 수 있습니다. 데이터 모델은 데이터 분석에 필요한 모든 메타데이터(필드 이름, 저장 공간, 측정 수준 등)를 정의합니다. 데이터 소스 소유자는 데이터 소스에 대한 액세스를 부여하거나 제한할 수 있습니다.

데이터 소스 목록

기본 데이터 소스 페이지에서는 현재 사용자가 멤버로 속해 있는 데이터 소스 목록을 제공합니다.

- 데이터 소스의 이름을 클릭하여 세부사항을 표시하고 특성을 편집합니다.
- 검색 영역에 입력하여 목록을 필터링해서 이름에 검색 문자열이 포함된 데이터 소스만 표시합니다.
- **새로 만들기** 단추를 클릭하면 **새 데이터 소스 추가** 대화 상자에서 지정한 이름 및 컨텐츠 유형을 가진 새 데이터 소스가 작성됩니다.
  - 데이터 소스에 지정할 수 있는 이름에 대한 제한사항은 31 페이지의 『이름 지정 규칙』의 내용을 참조하십시오.
  - 사용 가능한 컨텐츠 유형은 파일, 데이터베이스, HCatalog 및 Geospatial입니다.

    참고: HCatalog 옵션은 Analytic Server가 해당 데이터 소스와 작업할 수 있도록 구성된 경우에만 사용 가능합니다.

    참고: 컨텐츠 유형은 일단 선택된 후에는 편집할 수 없습니다.

- 데이터 소스를 제거하려면 **삭제**를 클릭하십시오. 이 조치는 데이터 소스와 연관된 모든 파일을 그대로 남겨 둡니다.
- 목록을 업데이트하려면 **새로 고치기**를 클릭하십시오.
- 조치 드롭 다운 목록에서는 선택한 조치를 수행합니다.
  1. **내보내기**를 선택하면 데이터 소스의 아카이브를 생성하여 로컬 파일 시스템에 저장합니다.
  2. **가져오기**를 선택하면 내보내기 조치로 생성된 아카이브를 가져옵니다.
  3. **복제**를 선택하면 데이터 소스의 사본을 생성합니다.

개별 데이터 소스 세부사항

컨텐츠 영역은 여러 섹션으로 나뉘며, 이는 데이터 소스의 컨텐츠 유형에 따라 달라질 수 있습니다.

세부사항

모든 컨텐츠 유형에 공통적인 설정입니다.

이름 데이터 소스의 이름을 표시하는 편집 가능한 텍스트 필드입니다.

표시 이름

다른 애플리케이션에서 표시되는 데이터 소스의 이름을 보여주는 편집 가능한 텍스트 필드입니다. 비어 있는 경우, 이름을 표시 이름으로 사용합니다.

설명 데이터 소스에 대한 설명 텍스트를 제공하는 편집 가능한 텍스트 필드입니다.

공용 여부

모든 사용자가 데이터 소스를 볼 수 있는지(선택됨) 또는 사용자 및 그룹이 멤버로 명시적으로 추가되어야 하는지(선택 취소) 여부를 표시하는 선택란입니다.

사용자 정의 속성

애플리케이션에서 사용자 정의 속성을 사용하여 데이터 소스에 특성을 연결할 수 있습니다(예: 임시 데이터 소스 여부). 이러한 속성은 애플리케이션에서 데이터 소스를 사용하는 방법을 파악할 수 있도록 Analytic Server 콘솔에 표시됩니다.

저장을 클릭하면 설정의 현재 상태가 저장됩니다.

공유

모든 컨텐츠 유형에 공통적인 설정입니다.

사용자 및 그룹을 작성자로 추가하여 데이터 소스의 소유권을 공유할 수 있습니다.

- 텍스트 상자에 입력하면 이름에 검색 문자열이 포함되어 있는 사용자 및 그룹이 필터링됩니다. 작성자 목록에 추가하려면 **멤버 추가**를 클릭하십시오.
- 작성자를 제거하려면 멤버 목록에서 사용자 또는 그룹을 선택하고 **멤버 제거**를 클릭하십시오.

참고: 관리자는 멤버로 나열되는 여부에 관계없이 모든 데이터 소스에 대해 읽기 및 쓰기 액세스 권한을 가집니다.

파일 입력

파일 컨텐츠 유형이 있는 데이터 소스를 정의하는 데 고유한 설정입니다.

파일 뷰어

데이터 소스에 포함될 사용 가능한 파일을 표시합니다. Analytic Server 프로젝트 구조 내에서 파일을 보려면 **프로젝트** 모드를 선택하고 데이터 소스에 저장된 파일을 보려면 **데이터 소스**를 선택하고 파일 시스템(일반적으로 HDFS)을 보려면 **파일 시스템**을 선택하십시오. 하나의 폴더 구조를 찾아볼 수는 있지만 HDFS는 편집 불가능하며 **Projects** 모드에서는 정의된 프로젝트 내에서만 루트 레벨로 항목을 삭제하거나 파일을 추가하거나 폴더를 작성할 수 없습니다. 프로젝트를 작성, 편집 또는 삭제하려면 프로젝트를 사용하십시오.

- **업로드**를 클릭하면 현재 데이터 소스 또는 프로젝트/하위 폴더로 파일을 업로드합니다. 단일 디렉토리에서 여러 파일을 찾거나 선택할 수 있습니다.
- **새 폴더**를 클릭하면 새 폴더 이름 대화 상자에서 지정한 이름의 새 폴더가 현재 폴더 아래에 작성됩니다.
- 선택한 파일을 로컬 파일 시스템으로 다운로드하려면 **다운로드**를 클릭하십시오.
- 선택한 파일/폴더를 제거하려면 **삭제**를 클릭하십시오.

데이터 소스 정의에 포함된 파일

이동 단추를 사용하여 데이터 소스에서 선택한 파일과 폴더를 추가하거나 제거하십시오. 데이터 소스에서 선택된 파일 또는 폴더 각각에 대해 파일 읽기에 필요한 지정 사항을 정의하려면 설정을 클릭하십시오.

데이터 소스에 여러 개의 파일이 포함된 경우, 해당 파일들이 공통 메타데이터를 공유해야 합니다. 즉, 각 파일에 같은 수의 필드가 있어야 하고 필드는 개별 파일에서 같은 순서대로 구문 분석되어 있고 각 필드는 전체 파일에서 같은 저장 공간을 사용해야 합니다. 파일이 서로 일치하지 않으면 콘솔에서 미리보기 및 메타데이터를 작성하지 못하거나 Analytic Server에서 파일을 읽을 때 유효한 값을 유효하지 않은(null) 것으로 구문 분석할 수 있습니다.

데이터베이스 선택

레코드 컨텐츠를 포함하는 데이터베이스에 대한 연결 매개변수를 지정하십시오.

데이터베이스

연결하려는 데이터베이스의 유형을 선택하십시오. DB2, Greenplum, MySQL, Netezza, Oracle, SQL Server, Sybase IQ 또는 TeraData에서 선택하십시오. 찾고 있는 유형이 목록에 없으면 서버 관리자에게 문의하여 적절한 JDBC 드라이버를 사용하여 Analytic Server를 구성하십시오.

서버 주소

데이터베이스를 호스팅하는 서버의 URL을 입력하십시오.

서버 포트

데이터베이스가 청취하는 포트 번호입니다.

데이터베이스 이름

연결하려는 데이터베이스의 이름입니다.

사용자 이름

데이터베이스의 비밀번호가 보호된 경우 사용자 이름을 입력하십시오.

비밀번호

데이터베이스의 비밀번호가 보호된 경우 사용자의 비밀번호를 입력하십시오.

테이블 이름

사용하려는 데이터베이스의 테이블 이름을 입력하십시오.

최대 동시 읽기 수

데이터 소스에 지정된 테이블에서 읽을 수 있도록 Analytic Server에서 데이터베이스로 전송할 수 있는 병렬 쿼리 수에 대한 제한을 입력하십시오.

HCatalog 선택

Apache HCatalog에서 관리되는 데이터에 액세스하기 위한 매개변수를 지정하십시오.

데이터베이스

HCatalog 데이터베이스의 이름입니다.

테이블 이름

사용하려는 데이터베이스의 테이블 이름을 입력하십시오.

필터 테이블이 파티션된 테이블로 작성된 경우 테이블에 대한 파티션 필터입니다. HCatalog 필터링은 유형 문자열의 Hive 파티션 키에서만 지원됩니다.

참고: !=, <> 및 LIKE 연산자는 특정 Hadoop 배포에서 작동하지 않습니다. 이는 HCatalog와 해당 배포 간의 호환성 문제 때문입니다.

HCatalog 필드 맵핑

HCatalog 요소 대 데이터 소스 필드의 맵핑을 표시합니다. 필드 맵핑을 수정하려면 편집을 클릭하십시오.

참고: Hive 테이블에서 데이터를 공개하는 HCatalog 기반 데이터 소스를 작성한 후에 다수의 데이터 파일에서 Hive 테이블을 생성하면 Analytic Server에서 데이터 소스의 데이터를 읽을 때마다 시간이 많이 지연될 수 있습니다. 이와 같은 상황이 발생하면 더 적은 데이터 파일을 사용해 Hive 테이블을 다시 빌드하고 파일 수를 400 미만으로 줄이십시오.

Geospatial 선택

지리 데이터에 액세스하기 위한 매개변수를 지정하십시오.

Geospatial 유형

지리 데이터는 온라인 맵 서비스 또는 shape 파일에서 가져올 수 있습니다.

맵 서비스를 사용 중인 경우 해당 서비스의 URL을 지정하고 사용할 맵 레이어를 선택하십시오.

shape 파일을 사용 중인 경우에는 shape 파일을 업로드하십시오.

미리보기 및 메타데이터

데이터 소스 설정을 지정한 후 미리보기 및 메타데이터를 클릭하여 데이터 소스 지정 사항을 검사하고 확인하십시오.

결과 파일 또는 데이터베이스 컨텐츠 유형이 있는 데이터 소스를 Analytic Server에서 실행되는 스트림의 결과로 추가할 수 있습니다. **쓰기 가능 설정**을 선택하여 추가할 수 있게하고 다음을 수행하십시오.

- 데이터베이스 컨텐츠 유형이 있는 데이터 소스의 경우 결과 데이터가 작성되는 결과 데이터베이스 테이블을 선택하십시오.
- 파일 컨텐츠 유형이 있는 데이터 소스의 경우 다음을 수행하십시오.
  1. 새 파일이 작성되는 결과 폴더를 선택하십시오.

     팁: 각 데이터 소스에 별도의 폴더를 사용하여 파일과 데이터 소스 간의 연관을 쉽게 추적하십시오.
  2. 파일 형식으로 **CSV**(Comma Separated Variable) 또는 **분할 가능한 2진 형식**을 선택하십시오.
  3. 선택적으로 **시퀀스 파일 작성**을 선택하십시오. 그러면 다운스트림 MapReduce 작업에 사용할 수 있는 분할 가능한 압축 파일을 작성할 때 유용합니다.
  4. **줄 바꾸기를 이스케이프할 수 있음**을 선택하면 결과 파일에서 데이터의 줄 바꾸기를 문자열 "\n"으로 작성하고 문자열 "\n"은 "\\n"으로 작성합니다. 선택하지 않으면 결과 파일에서 문자열 "\n"이 "\n"으로 작성되고 줄 바꾸기가 있으면 오류가 발생합니다.
  5. 압축 형식을 선택하십시오. 이 목록에는 Analytic Server 설치에 사용하도록 구성된 모든 형식이 포함됩니다.

     참고: 일부 압축 형식과 파일 형식을 조합하면 결과를 분할할 수 없게 되어 추가 MapReduce 처리에 적합하지 않습니다. Analytic Server에서는 사용자가 이러한 조합을 선택하면 결과 섹션에 경고를 생성합니다.

설정(파일 데이터 소스)

설정 대화 상자에서는 파일 기반 데이터를 읽는 데 필요한 지정 사항을 정의할 수 있습니다. 이 설정은 선택한 모든 파일과 **파일 선택** 탭의 기준에 일치하는 선택한 폴더 내의 모든 파일에 적용됩니다.

파일에 올바르지 않은 구문 분석기 설정을 지정하면 콘솔에서 미리보기 및 메타데이터를 작성하지 못하거나 Analytic Server에서 파일을 읽을 때 유효한 값을 유효하지 않은(null) 것으로 구문 분석할 수 있습니다.

설정 탭

설정 탭에서는 파일 유형과 파일 유형에 해당하는 구분 문석기 설정을 지정할 수 있습니다.

지원되는 파일 형식에 대한 압축 파일을 사용하여 데이터 소스를 정의할 수 있습니다. 지원되는 압축 형식으로는 Gzip, Deflate, Bz2, Snappy 및 IBM CMX가 있습니다.

구분된 파일 유형

구분된 파일은 자유 필드 텍스트 파일이며 레코드에 일정 개수의 필드가 포함되어 있지만 필드당 문자 수는 다릅니다. 구분된 파일의 확장자는 일반적으로 *.csv 또는 *.tab입니다. 자세한 정보는 9 페이지의 『구분된 파일 유형 설정』을 참조하십시오.

고정 파일 유형

고정 필드 텍스트 파일은 필드가 구분되어 있지 않은 파일이지만 고정된 길이이며 같은 위치에서 시작합니다. 고정 필드 텍스트 파일의 확장자는 일반적으로 *.dat입니다. 자세한 정보는 10 페이지의 『고정 파일 유형 설정』을 참조하십시오.

부분 구조화 파일 유형

부분 구조화 파일(예: *.log)은 정규식을 통해 필드로 맵핑할 수 있는 예측 가능한 구조를 사용하지만 구분된 파일 만큼 구조화되어 있지는 않습니다. 자세한 정보는 11 페이지의 『부분 구조화 파일 유형 설정』을 참조하십시오.

텍스트 분석 파일 유형

텍스트 분석 파일은 SPSS Text Analytics를 사용하여 분석할 수 있는 문서(예: *.doc, *.pdf 또는 *.txt)입니다.

비어 있는 행 건너뛰기
: 추출된 텍스트 컨텐츠에서 비어 있는 행을 무시할 것인지 지정합니다. 기본값은 **아니오**입니다.

행 구분 문자
: 줄 바꾸기를 정의하는 문자열을 지정합니다. 기본값은 줄 바꾸기 문자 "\n"입니다.

SPSS Statistics 파일 유형

SPSS Statistics 파일(*.sav, *.zsav)은 데이터 모델이 포함된 2진 파일입니다. 이 파일 유형은 설정 탭에서 추가 설정이 필요하지 않습니다.

분할 가능한 2진 형식 파일 유형

파일 유형이 분할 가능한 2진 형식 파일(*.asbf)임을 지정합니다. 이 파일 유형이 Analytic Server의 결과인 경우가 있습니다. 예를 들어, 분석에 목록 값이 있는 필드를 사용해야 하는 경우입니다. 이 파일 유형은 설정 탭에서 추가 설정이 필요하지 않습니다.

시퀀스 파일 유형

시퀀스 파일(*.seq)은 키/값 쌍으로 구조화된 텍스트 파일입니다. 이 파일은 일반적으로 MapReduce 작업에서 중개 형식으로 사용됩니다.

Excel 파일 유형

파일 유형이 Microsoft Excel 파일(*.xls, *.xlsx)임을 지정합니다. 자세한 정보는 12 페이지의 『Excel 파일 유형 설정』을 참조하십시오.

구분된 파일 유형 설정:

구분된 파일 유형에 다음 설정을 지정할 수 있습니다.

문자 세트 인코딩

파일의 문자 인코딩입니다. Java 문자 세트 이름(예: "UTF-8", "ISO-8859-2", "GB18030")을 선택하거나 지정하십시오. 기본값은 **UTF-8**입니다.

필드 구분자

필드 경계를 표시하는 하나 이상의 문자입니다. 각 문자는 독립된 구분자로 간주됩니다. 예를 들어, **쉼표 및 탭**을 선택하면(또는 **기타**를 선택하고 **,\t**를 입력하는 경우) 쉼표 또는 탭이 필드 경계를 표시합니다. 제어 문자가 필드를 구분하는 경우 여기에 지정된 문자는 제어 문자 이외에 구분자로 처리됩니다. 제어 문자가 필드를 구분하지 않는 경우 기본값은 ","이며 그렇지 않으면 기본값이 비어 있는 문자열입니다.

제어 문자가 필드를 구분함

LF 및 CR을 제외한 ASCII 제어 문자가 필드 구분자로 처리되는지 여부를 설정합니다. 기본값은 **아니오**입니다.

첫 번째 행에 필드 이름이 있음

첫 번째 행을 사용하여 필드 이름을 판별할지 여부를 설정합니다. 기본값은 **아니오**입니다.

건너뛸 처음 문자 수

파일 시작 부분에서 건너뛸 문자 수. 비음수 정수입니다. 기본값은 0입니다.

공백 병합

인접하여 발생한 여러 개의 공백 및/또는 탭을 단일 필드 구분자로 간주할지 여부를 설정합니다. 공백도 탭도 필드 구분자가 아닌 경우에는 효과가 없습니다. 기본값은 **예**입니다.

행의 끝 주석 문자

행의 끝 주석을 표시하는 하나 이상의 문자입니다. 레코드에서 문자와 문자 다음에 오는 모든 사항이 무시됩니다. 각 문자는 독립된 주석 마커로 간주됩니다. 예를 들어, "/*"는 슬래시 또는 별표가 주석을 시작한다는 의미입니다. "//"와 같이 여러 문자 주석 마커를 정의하는 것은 불가능합니다. 빈 문자열은 주석 문자가 정의되지 않았음을 표시합니다. 정의된 경우 따옴표가 처리되거나 건너뛸 처음 문자를 건너뛰기 전에 주석 문자를 검사합니다. 기본값은 빈 문자열입니다.

올바르지 않은 문자

올바르지 않은 문자(인코딩에서 문자에 해당하지 않는 바이트 시퀀스)를 처리하는 방법을 판별합니다. 빈 문자열은 해당 문자가 버려짐을 표시하고 비어 있지 않은 문자열(보통 단일 문자)은 해당 문자가 문자열의 컨텐츠로 대체됨을 표시합니다. 기본값은 빈 문자열입니다.

작은따옴표

작은따옴표(아포스트로피)의 처리를 지정합니다. 기본값은 **유지**입니다.

유지 작은따옴표에 특별한 의미가 없고 다른 문자와 같이 처리됩니다.

삭제 작은따옴표가 따옴표로 묶이지 않는 한 삭제됩니다.

쌍 작은따옴표가 인용 문자로 처리되고 작은따옴표 쌍 사이의 문자는 특별한 의미를 잃게 됩니다(인용된 것으로 간주됨). 작은따옴표 자체가 작은따옴표로 묶인 문자열 내에서 발생할 수 있는지 여부는 **따옴표가 이중화로 인용될 수 있음** 설정으로 판별됩니다.

큰따옴표

큰따옴표의 처리 방법을 지정합니다. 기본값은 **쌍**입니다.

유지 큰따옴표에 특별한 의미가 없고 다른 문자와 같이 처리됩니다.

삭제 큰따옴표를 따옴표로 묶지 않으면 삭제됩니다.

쌍 큰따옴표가 인용 문자로 처리되고 큰따옴표 쌍 사이의 문자는 특별한 의미를 잃게 됩니다(인용된 것으로 간주됨). 큰따옴표 자체가 큰따옴표로 묶인 문자열 내에서 발생할 수 있는지 여부는 **따옴표가 이중화로 인용될 수 있음** 설정으로 판별됩니다.

따옴표가 이중화로 인용될 수 있음

쌍으로 설정된 경우 큰따옴표로 묶인 문자열에서 큰따옴표를 나타낼 수 있고 작은따옴표로 묶인 문자열에서 작은따옴표를 나타낼 수 있는지 여부를 표시합니다. **예**인 경우 큰따옴표가 이중화를 통해 큰따옴표로 묶인 문자열 내에서 이탈되고 작은따옴표가 이중화를 통해 작은따옴표로 묶인 문자열 내에서 이탈됩니다. **아니오**인 경우 큰따옴표로 묶인 문자열 내에서 큰따옴표를 따옴표로 묶거나 작은따옴표로 묶인 문자열 내에서 작은따옴표를 따옴표로 묶는 방법은 없습니다. 기본값은 **예**입니다.

줄 바꾸기를 이스케이프할 수 있음

파일을 읽을 때 구문 분석기에서 문자열 "\n"을 줄 바꾸기로 해석할 수 있는지 표시합니다. 줄 바꾸기가 이스케이프되지 않으면 "\n"을 문자열로 읽습니다. 줄 바꾸기가 이스케이프되면 "\n"을 ASCII 줄 바꾸기 문자로 읽으며 "\\n"은 문자열 "\n"으로 읽습니다. 기본값은 **아니오**입니다.

고정 파일 유형 설정:

고정 파일 유형에 다음 설정을 지정할 수 있습니다.

문자 세트 인코딩

파일의 문자 인코딩입니다. Java 문자 세트 이름(예: "UTF-8", "ISO-8859-2", "GB18030")을 선택하거나 지정하십시오. 기본값은 **UTF-8**입니다.

올바르지 않은 문자

올바르지 않은 문자(인코딩에서 문자에 해당하지 않는 바이트 시퀀스)를 처리하는 방법을 판별합니다. 빈 문자열은 해당 문자가 버려짐을 표시하고 비어 있지 않은 문자열(보통 단일 문자)은 해당 문자가 문자열의 컨텐츠로 대체됨을 표시합니다. 기본값은 빈 문자열입니다.

레코드 길이

레코드를 정의하는 방법을 표시합니다. **줄 바꾸기로 구분함**이면 레코드가 줄 바꾸기, 파일 시작 또는 파일 종료에 의해 정의(구분)됩니다. **특정 길이**인 경우, 레코드가 레코드 길이(바이트)로 정의됩니다. 양수값을 지정합니다.

건너뛸 초기 레코드

파일 시작 부분에서 건너뛸 레코드 수. 음수가 아닌 정수를 지정합니다. 기본값은 0입니다.

필드 이 섹션에서는 파일의 필드를 정의합니다. **필드 추가**를 클릭하고 필드 이름, 필드 값이 시작하는 열, 필드 값의 길이를 지정하십시오. 파일의 열은 0부터 시작하여 번호가 지정됩니다.

부분 구조화 파일 유형 설정:

부분 구조화 파일의 설정은 파일 컨텐츠를 필드로 맵핑하는 규칙으로 구성됩니다.

규칙 테이블

개별 규칙에서 레코드의 정보를 추출하여 필드를 작성합니다. 이러한 규칙은 규칙 테이블에서 함께 데이터 소스의 각 레코드에서 추출할 수 있는 모든 필드를 정의합니다.

테이블의 규칙은 각 레코드에 순서대로 적용됩니다. 테이블의 모든 규칙이 레코드와 일치하면 레코드 처리에 다른 규칙 테이블이 필요하지 않으며 다음 레코드가 처리됩니다. 테이블에 일치하지 않는 규칙이 있으면 테이블의 이전 규칙으로 추출한 모든 필드 값을 버립니다. 다른 규칙 테이블이 있으면 해당 테이블의 규칙을 레코드에 적용합니다. 레코드와 일치하는 테이블이 없으면 미스매치 규칙이 적용됩니다.

미스매치

규칙 테이블과 일치하지 않는 레코드에 대해 **건너뛰기**를 선택하거나 레코드의 모든 필드 값을 **결측**(널)으로 설정할 수 있습니다.

규칙 내보내기

현재 표시되는 규칙 테이블을 저장하여 재사용할 수 있습니다. 내보낸 테이블을 서버에 저장할 수 있습니다.

규칙 가져오기

저장된 규칙 테이블을 현재 표시되는 규칙 테이블로 가져올 수 있습니다. 그러면 해당 테이블에 대해 정의한 모든 규칙을 덮어쓰기 때문에 새 테이블을 작성하고 규칙 테이블을 가져오는 것이 좋습니다.

규칙 편집기

규칙 편집기에서는 단일 필드의 추출 규칙을 생성할 수 있습니다.

익명 캡처 그룹

필드 캡처 규칙에서는 일반적으로 이전에 규칙이 중지된 위치에서 레코드의 데이터 추출을 시작합니다. 부분 구조화 데이터 소스의 두 필드 간에 잘못된 정보가 있는 경우, 따라서 다음 필드가 시작되는 위치에 구문 분석기를 배치하는 익명 캡처 그룹을 정의하면 유용합니다. **익명 캡처 그룹**을 선택하면 캡처 그룹에 대한 이름 지정 및 레이블 지정 제어를 사용하지 않지만 나머지 대화 상자는 정상적으로 작동합니다.

필드 이름

필드 이름을 입력하십시오. 이는 데이터 소스 메타데이터를 정의하는 데 사용합니다. 필드 이름은 규칙 테이블 내에서 고유해야 합니다.

규칙 이름

선택적으로 규칙의 설명 레이블을 입력하십시오.

설명 선택적으로 규칙에 대한 자세한 설명을 입력하십시오.

규칙 정의

규칙을 정의하는 두 가지 방법이 있습니다.

추출 규칙에 제어 사용

추출 규칙을 간단하게 작성할 수 있게 합니다.

1. 필드 데이터의 추출 시작점을 지정하십시오. **현재 위치**는 이전 규칙이 중지된 위치에서 시작되며 **다음까지 건너뛰기**가 레코드 시작점에서 시작되어 텍스트 상자에 지정된 문자에 도달할 때까지 모든 문자를 무시합니다. 필드 데이터에 시작점의 문자를 포함하려면 **포함**을 선택하십시오.
2. **캡처** 드롭 다운에서 필드 캡처 그룹을 선택하십시오.
3. 선택적으로 필드 데이터의 추출을 중지할 지점을 선택하십시오. **공백**을 사용하면 공백 문자(예: 공백 또는 탭)가 발견될 때 중지하고 **문자**를 사용하면 지정된 문자열에서 중지합니다. 필드 데이터에 중지 위치의 문자를 포함하려면 **포함**을 선택하십시오.

regexp 규칙 수동 정의

정규식 구문을 작성하는 데 익숙한 경우 이 옵션을 선택하십시오. **Regexp** 텍스트 상자에 정규식을 입력하십시오.

필드 캡처 그룹 추가

나중에 사용할 정규식을 저장할 수 있습니다. 저장된 캡처 그룹은 **캡처** 드롭 다운에 표시됩니다.

규칙 편집기에서는 규칙 테이블의 모든 이전 규칙을 적용한 후에 이 규칙에 따라 첫 번째 레코드에서 추출한 데이터의 미리보기를 표시합니다.

Excel 파일 유형 설정:

Excel 파일에 다음 설정을 지정할 수 있습니다.

워크시트 선택

데이터 소스로 사용할 Excel 워크시트를 선택합니다. 워크시트의 이름이나 숫자 색인(첫 번째 워크시트의 색인은 0임)을 지정하십시오. 기본값은 첫 번째 워크시트를 사용하는 것입니다.

가져오기를 위한 데이터 범위 선택

비어 있지 않은 첫 번째 행이나 명시적 셀 범위로 시작하는 데이터를 가져올 수 있습니다.

- **비어 있지 않은 첫 번째 행으로 범위가 시작합니다**. 비어 있지 않은 첫 번째 셀을 찾아서 데이터 범위의 왼쪽 상단에 이를 사용합니다.

- 또는 행과 열을 사용해 셀의 명시적 범위를 지정합니다. 예를 들어, Excel 범위 A1:D5를 지정하려면 첫 번째 필드에 A1을 입력하고 두 번째에 D5를 입력할 수 있습니다(또는 R1C1 및 R5C4). 공백 행을 포함한 지정된 범위의 모든 행이 리턴됩니다.

첫 번째 행에 필드 이름이 있음

선택할 셀 범위의 첫 번째 행에 필드 이름이 포함되는지 지정합니다. 기본값은 **아니오**입니다.

공백 행이 발견되면 읽기 중지

한 개 이상의 공백 행이 발견되면 레코드 읽기를 중지할 것인지 워크시트 끝까지 공백 행을 포함한 모든 데이터를 읽을 것인지 지정합니다. 기본값은 **아니오**입니다.

형식

형식 탭에서는 구문 분석된 필드의 형식화 정보를 정의할 수 있습니다.

필드 변환 설정

공백 자르기

문자열 필드의 시작 및/또는 끝 부분에서 공백 문자를 제거합니다. 기본값은 **없음**입니다. 다음 같이 지원됩니다.

없음 공백 문자를 제거하지 않습니다.

왼쪽 문자열의 시작 부분에서 공백 문자를 제거합니다.

오른쪽 문자열의 끝 부분에서 공백 문자를 제거합니다.

둘 다 문자열의 시작 또는 끝 부분에서 공백 문자를 제거합니다.

로케일 로케일을 정의합니다. 기본값은 서버 로케일입니다. 로케일 문자열은 <language>[_country[_variant]]로 지정되어야 합니다. 여기서,

language

ISO-639에 정의된 올바른 두 문자(소문자)로 된 코드입니다.

country

ISO-3166에 정의된 올바른 두 문자(대문자)로 된 코드입니다.

variant

벤더 또는 브라우저 특정 코드입니다.

소수점 구분자

소수점 부호로 사용되는 문자를 설정합니다. 기본값은 로케일별 설정입니다.

기호 그룹화

수천 개의 구분 문자에 사용되는 로케일 고유 문자가 사용되어야 하는지 여부를 설정합니다.

기본 날짜 형식

기본 날짜 형식을 정의합니다. 유니코드 LDML(Locale Data Markup Language) 지정 사항에서 정의하는 모든 형식 패턴이 지원됩니다.

기본 시간 형식
기본 시간 형식을 정의합니다.

기본 시간소인
기본 시간소인 형식을 정의합니다.

기본 시간대
시간대를 설정합니다. 기본값은 UTC입니다. 이 설정은 명시적으로 지정된 시간대가 없는 시간 및 시간소인 필드에 적용됩니다.

필드 대체

이 섹션에서는 개별 필드에 형식화 지시사항을 지정할 수 있습니다. 데이터 모델에서 필드를 선택하거나 필드 이름을 입력하고 **추가**를 클릭하여 개별 지시사항이 있는 필드의 목록에 이를 추가하십시오. 목록에서 제거하려면 **제거**를 클릭하십시오. 목록에서 선택한 필드는 다음 필드 특성을 설정할 수 있습니다.

저장 공간
필드의 저장 공간을 설정하십시오.

소수점 구분자
실수 저장 공간이 있는 필드의 경우, 소수점 부호로 사용할 문자를 설정하십시오. 기본값은 로케일별 설정입니다.

기호 그룹화
정수 또는 실수 저장 공간이 있는 필드에서 천단위 구분 기호에 로케일별 문자를 사용할 것인지 설정합니다.

형식 날짜, 시간 또는 시간소인 저장 공간이 있는 필드에 대한 형식을 설정합니다. 드롭 다운 목록에서 형식을 선택하십시오.

필드 순서 탭

구분된 Excel 파일 유형의 필드 순서 탭에서는 파일의 필드에 대한 구문 분석된 순서를 정의할 수 있습니다. 실제 필드 순서는 파일에서 다를 수 있지만 일관적인 데이터 모델을 작성하려면 구문 분석된 필드 순서가 동일해야 하므로 이는 데이터 소스에 여러 개의 파일이 있는 경우 중요합니다.

고정 및 부분 구조화 파일 유형의 경우 설정 탭에서 순서를 정의합니다.

데이터 소스에 한 개의 파일이 있거나 모든 파일에 같은 필드 순서를 사용하면 기본 **필드 순서가 데이터 모델과 일치함**을 사용할 수 있습니다. 데이터 소스에 여러 개의 파일이 있지만 파일의 필드 순서가 일치하지 않으면 파일 구문 분석에 **특정 필드 순서**를 정의하십시오.

1. 정렬된 목록에 필드를 추가하려면 필드 이름을 입력하거나 데이터 모델에서 제공하는 목록에서 필드를 선택하십시오. **모두 추가**를 클릭하면 데이터 모델의 모든 필드를 한 번에 추가할 수 있습니다. 필드 이름은 정렬된 목록에 한 번만 추가됩니다.
2. 화살표 단추를 사용하여 필드를 원하는 대로 정렬하십시오.

특정 필드 순서를 사용하면 목록에 추가되지 않은 모든 필드가 이 파일의 결과 세트의 일부가 아닙니다. 이 대화 상자에 나열되지 않은 데이터 모델에 필드가 있으면 결과 세트에서 값이 널입니다.

폴더 탭

폴더에 대한 구문 분석기 설정을 지정할 때 폴더 탭에서 폴더의 파일 중 데이터 소스에 포함할 파일을 선택할 수 있습니다.

선택한 폴더의 모든 파일 일치

데이터 소스에 폴더 최상위 레벨의 모든 파일이 포함됩니다. 하위 폴더의 파일은 포함되지 않습니다.

정규식을 사용하여 파일 일치

데이터 소스에 지정된 정규식과 일치하는 폴더의 최상위 레벨에 있는 모든 파일이 포함됩니다. 하위 폴더의 파일은 포함되지 않습니다.

Unix 글로빙 표현식을 사용하여 파일 일치(잠재적으로 재귀적임)

데이터 소스에는 지정된 Unix 글로빙 표현식과 일치하는 모든 파일이 포함됩니다. 표현식에는 선택한 폴더의 하위 폴더에 있는 파일을 포함할 수 있습니다.

HCatalog 필드 맵핑

HCatalog 스키마

지정된 테이블의 구조를 표시합니다. HCatalog는 고도로 구조화된 데이터 세트를 지원할 수 있습니다. 이러한 데이터에서 Analytic Server 데이터 소스를 정의하려면 구조가 단순 행과 열로 구성되어야 합니다. 스키마의 요소를 선택하고 이동 단추를 클릭하여 분석할 필드에 맵핑하십시오.

트리 노드 중 일부는 맵핑되지 않습니다. 예를 들어, 복합 유형의 맵 또는 배열은 "상위"로 취급되어 직접 맵핑할 수 없습니다. HCatalog 배열 또는 맵의 개별 단순 요소는 별도로 추가해야 합니다. 이러한 노드는 트리에서 ...:array:struct 또는 ...:map:struct로 끝나는 레이블로 식별될 수 있습니다.

예를 들어, 다음과 같습니다.

- 정수 배열의 경우, 필드를 배열: bigintarray[45] 내의 값에 지정할 수 있지만 배열 자체: bigintarray에는 지정할 수 없습니다.
- 맵의 경우, 필드를 맵: datamap["key"] 내의 값에 지정할 수 있지만 맵 자체: datamap에는 지정할 수 없습니다.
- 정수 배열의 경우, 필드를 값 bigintarrayarray[45][2]에 지정할 수 있지만 배열 자체인 bigintarrayarray[45]에 지정할 수 없습니다.

따라서 필드를 배열이나 맵 요소에 지정할 때 요소 정의에 색인이나 키: bigintarray[index] 또는 bigintmap["key"]가 있어야 합니다.

필드 맵핑

HCatalog 요소

편집할 셀을 두 번 클릭하십시오. HCatalog 요소가 배열 또는 맵인 경우 셀을 편집해야 합니

다. 배열에 대해서는 필드에 맵핑할 배열의 멤버에 해당하는 정수를 지정하십시오. 맵에 대해서는 필드에 맵핑할 키에 해당하는 따옴표로 묶인 문자를 지정하십시오.

맵핑 필드

Analytic Server 데이터 소스에 나타나는 필드입니다. 편집할 셀을 두 번 클릭하십시오. 맵핑 필드 열의 중복 항목 값은 허용되지 않으므로 오류로 표시됩니다.

저장 공간

필드의 저장 공간입니다. 저장 공간은 HCatalog에서 도출되며 편집할 수 없습니다.

참고: 미리보기 및 메타데이터를 클릭하여 Hcatalog 데이터 소스를 완료하면 편집 옵션이 표시되지 않습니다.

원시 데이터

HCatalog에 저장된 레코드를 표시합니다. 이를 통해 HCatalog 스키마를 필드에 맵핑하는 방법을 판별할 수 있습니다.

참고: HCatalog 선택에 지정된 필터링은 원시 데이터 보기에 적용됩니다.

HCatalog 데이터 소스 사용

Analytic Server는 HCatalog 데이터 소스를 지원합니다. 이 절에서는 여러 기본 NoSQL 데이터베이스를 사용으로 설정하는 방법에 대해 설명합니다.

Apache Accumulo

Analytic Server는 Apache Accumulo로 된 기반 컨텐츠가 있는 HCatalog 데이터 소스를 지원합니다.

Apache Accumulo 분산 키/값 저장소는 Google의 BigTable 디자인 기반의 데이터 저장 공간 및 검색 시스템이며 Apache Hadoop, Zookeeper 및 Thrift 위에 작성됩니다. Apache Accumulo에서는 셀 기반 액세스 제어 양식에서 BigTable 디자인에 고급 개선사항을 제공하며 키/값 쌍을 데이터 관리 프로세스의 다양한 지점에서 수정할 수 있는 서버측 프로그래밍 메커니즘을 제공합니다.

Hive에서 외부 Apache Accumulo 테이블을 작성하려면 다음 구문을 사용하십시오.

```
set accumulo.instance.id=<instance_name>;
set accumulo.user.name=<user_name>;
set accumulo.user.pass=<user_password>;
set accumulo.zookeepers=<zookeeper_host_port>;

CREATE EXTERNAL TABLE <hive_table_name>(<table_column_specifications>)
STORED BY 'com.ibm.spss.hcatalog.AccumuloStorageHandler'
WITH SERDEPROPERTIES (
'accumulo.columns.mapping' = '<family_and_qualifier_mappings>',
'accumulo.table.name' = '<Accumulo_table_name>')
TBLPROPERTIES (
    "accumulo.instance.id"="<instance_name>",
    "accumulo.zookeepers"="<zookeeper_host_port>"
);
```

예를 들어, 다음과 같습니다.

```
set accumulo.instance.id=<id>;
set accumulo.user.name=admin;
set accumulo.user.pass=test;
set accumulo.zookeepers=<host>:<port>;

CREATE EXTERNAL TABLE acc_drug1n(rowid STRING,age STRING,sex STRING,bp STRING,
    cholesterol STRING,na STRING,k STRING,drug STRING)
STORED BY 'com.ibm.spss.hcatalog.AccumuloStorageHandler'
WITH SERDEPROPERTIES (
'accumulo.columns.mapping' = 'rowID,drug|age,drug|sex,drug|bp,drug|cholesterol,
    drug|na,drug|k,drug|drug',
'accumulo.table.name' = 'drug1n')
TBLPROPERTIES (
    "accumulo.instance.id"="<id>",
    "accumulo.zookeepers"="<host>:<port>"
);
```

참고: 지정된 Accumulo 테이블에 사용할 Accumulo 사용자 이름과 비밀번호는 인증된 Analytic Server 사용자의 사용자 이름 및 비밀번호와 일치해야 합니다.

Apache Cassandra

Analytic Server는 Apache Cassandra로 된 기반 컨텐츠가 있는 HCatalog 데이터 소스를 지원합니다.

Cassandra에서는 구조화된 키-값 저장소를 제공합니다. 키는 열 계열로 그룹화된 여러 값으로 맵핑됩니다. 열 계열은 데이터베이스를 작성할 때 결정되지만 열을 언제든 계열에 추가할 수 있습니다. 또한 열은 지정된 키에만 추가되므로 임의의 계열에서 서로 다른 키는 다른 개수의 열을 가질 수 있습니다. 개별 키의 열 계열의 값은 함께 저장됩니다.

Cassandra 테이블을 정의하는 두 가지 방법이 있습니다. 레거시 Cassandra 명령행 인터페이스(cassandra-cli)와 새 CQL 쉘(csqlsh)을 사용하는 것입니다.

레거시 CLI를 사용하여 테이블을 작성한 경우 다음 구문을 사용하여 외부 Apache Cassandra 테이블을 Hive에 작성하십시오.

```
CREATE EXTERNAL TABLE <hive_table_name>  (<column specifications>)
STORED BY 'com.ibm.spss.hcatalog.CassandraStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "<cassandra_column_family>",
"cassandra.host"="<cassandra_host>","cassandra.port" = "<cassandra_port>")
TBLPROPERTIES ("cassandra.ks.name" = "<cassandra_keyspace>");
```

예를 들어, 다음 CLI 테이블 정의가 있다고 가정합니다.

```
create keyspace test
with placement_strategy = 'org.apache.cassandra.locator.SimpleStrategy'
and strategy_options = [{replication_factor:1}];

create column family users with comparator = UTF8Type;

update column family users with
        column_metadata =
        [
        {column_name: first, validation_class: UTF8Type},
        {column_name: last, validation_class: UTF8Type},
```

```
        {column_name: age, validation_class: UTF8Type, index_type: KEYS}
        ];

assume users keys as utf8;

set users['jsmith']['first'] = 'John';
set users['jsmith']['last'] = 'Smith';
set users['jsmith']['age'] = '38';
set users['jdoe']['first'] = 'John';
set users['jdoe']['last'] = 'Dow';
set users['jdoe']['age'] = '42';

get users['jdoe'];
```

... Hive 테이블 DDL은 다음과 같이 표시됩니다.

```
CREATE EXTERNAL TABLE cassandra_users  (key string, first string, last string, age string)
STORED BY 'com.ibm.spss.hcatalog.CassandraStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "users",
"cassandra.host"="<cassandra_host>","cassandra.port" = "9160")
TBLPROPERTIES ("cassandra.ks.name" = "test");
```

CQL을 사용하여 테이블을 작성한 경우, 다음 구문을 사용하여 외부 Apache Cassandra 테이블을 Hive에 작성하십시오.

```
CREATE EXTERNAL TABLE <hive_table_name>  (<column specifications>)
STORED BY 'com.ibm.spss.hcatalog.CassandraCqlStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "<cassandra_column_family>",
"cassandra.host"="<cassandra_host>","cassandra.port" = "<cassandra_port>")
TBLPROPERTIES ("cassandra.ks.name" = "<cassandra_keyspace>");
```

예를 들어, 다음 CQL3 테이블 정의의 경우 다음과 같습니다.

```
CREATE KEYSPACE TEST WITH REPLICATION
= { 'class' : 'SimpleStrategy', 'replication_factor' : 2 };
USE TEST;

CREATE TABLE bankloan_10(
  row int,
  age int,
  ed int,
  employ int,
  address int,
  income int,
  debtinc double,
  creddebt double,
  othdebt double,
  default int,
  PRIMARY KEY(row)
);

INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (1,41,3,17,12,176,9.3,11.359392,5.008608,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (2,27,1,10,6,31,17.3,1.362202,4.000798,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (3,40,1,15,14,55,5.5,0.856075,2.168925,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (4,41,1,15,14,120,2.9,2.65872,0.82128,0);
```

```
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (5,24,2,2,0,28,17.3,1.787436,3.056564,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (6,41,2,5,5,25,10.2,0.3927,2.1573,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (7,39,1,20,9,67,30.6,3.833874,16.668126,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (8,43,1,12,11,38,3.6,0.128592,1.239408,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (9,24,1,3,4,19,24.4,1.358348,3.277652,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (10,36,1,0,13,25,19.7,2.7777,2.1473,0);
```

... Hive 테이블 DDL은 다음과 같습니다.

```
CREATE EXTERNAL TABLE cassandra_bankloan_10 (row int, age int,ed int,employ int,address int,
   income int,debtinc double,creddebt double,othdebt double,default int)
STORED BY 'com.ibm.spss.hcatalog.CassandraCqlStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "bankloan_10","cassandra.host"="<cassandra_host>",
   "cassandra.port" = "9160")
TBLPROPERTIES ("cassandra.ks.name" = "test");
```

Apache HBase

Analytic Server는 Apache HBase로 된 기반 컨텐츠가 있는 HCatalog 데이터 소스를 지원합니다.

Apache HBase는 Hadoop 및 HDFS 위에서 오픈 소스로 분산되어 있는 버전화된 열 지향 저장소입니다.

Hive에서 외부 HBase 테이블을 작성하려면 다음 구문을 사용하십시오.

```
CREATE EXTERNAL TABLE <tablename>(<table_column_specifications>)
STORED BY 'com.ibm.spss.hcatalog.HBaseStorageHandler'
WITH SERDEPROPERTIES ("hbase.columns.mapping" = "<column_mapping_spec>")
TBLPROPERTIES("hbase.table.name" = "<hbase_table_name>")
```

예를 들어, 다음과 같습니다.

```
CREATE EXTERNAL TABLE hbase_drug1n(rowid STRING,age STRING,sex STRING,bp STRING,
    cholesterol STRING,na STRING,k STRING,drug STRING)
STORED BY 'com.ibm.spss.hcatalog.HBaseStorageHandler'
WITH SERDEPROPERTIES ("hbase.columns.mapping" = ":key,drug:age,drug:sex,drug:bp,
    drug:cholesterol,drug:na,drug:k,drug:drug")
TBLPROPERTIES("hbase.table.name" = "drug1n");
```

참고: HBase 테이블을 작성하는 방법은 Apache HBase 참조 안내서 (http://hbase.apache.org/book.html)를 확인하십시오.

참고: 데이터베이스 이름을 앞에 표시하여 데이터베이스 유형을 표시하는 것이 좋습니다. 예를 들어, 데이터베이스 이름을 HB_drug1n으로 지정하여 HBase 데이터베이스인 것을 표시하거나 ACC_drug1n으로 지정하여 Accumulo 데이터베이스인 것을 표시하십시오. 그러면 Analytic Server 콘솔에서 HCatalog 파일을 선택할 때 도움이 됩니다.

MongoDB

Analytic Server는 MongoDB로 된 기본 컨텐츠가 있는 HCatalog 데이터 소스를 지원합니다.

MongoDB는 오픈 소스 문서 데이터베이스이며 C++로 작성된 최신 NoSQL 데이터베이스입니다. 이 데이터베이스에는 동적 스키마가 있는 JSON 스타일 문서를 저장합니다.

Hive에서 외부 MongoDB 테이블을 작성하려면 다음 구문을 사용하십시오.

```
create external table <hive_table_name>(<column specifications>)
stored by "com.ibm.spss.hcatalog.MongoDBStorageHandler"
with serdeproperties ( "mongo.column.mapping" = "<MongoDB to Hive mapping>" )
tblproperties ( "mongo.uri" = "'mongodb://<host>:<port>/<database>.<collection>" );
```

예를 들어, 다음과 같습니다.

```
create external table mongo_bankloan
(age bigint,ed bigint,employ bigint, address bigint,income bigint,
    debtinc double, creddebt double,othdebt double,default bigint)
STORED BY 'com.ibm.spss.hcatalog.MongoDBStorageHandler'
with serdeproperties ( 'mongo.column.mapping'
= '{"age":"age","ed":"ed","employ":"employ","address":"address",
    "income":"income","debtinc":"debtinc","creddebt":"creddebt","othdebt":"othdebt",
 "default":"default"}' )
tblproperties ('mongo.uri'='mongodb://9.48.11.162:27017/test.bankloan');
```

Oracle NoSQL

Analytic Server는 Oracle NoSQL로 된 기본 컨텐츠가 있는 HCatalog 데이터 소스를 지원합니다.

Oracle NoSQL 데이터베이스는 분산 키-값 데이터베이스입니다. 데이터는 키-값 쌍으로 저장되어 해시된 기본 키 값을 기반으로 특정 저장 공간 노드에 작성됩니다. 저장 공간 노드는 고가용성을 위해 복제합니다. 고객 애플리케이션은 데이터 읽기 및 쓰기를 위해 Java/C API를 사용하여 작성됩니다.

SerDe 및 테이블 매개변수

Oracle NoSQL 저장 공간 핸들러는 다음 매개변수를 지원합니다.

SERDEPROPERTIES 매개변수

kv.major.keys.mapping
쉼표로 구분된 주요 키 목록입니다. 필수입니다.

kv.minor.keys.mapping
쉼표로 구분된 보조 키 목록입니다. 선택사항입니다.

kv.parent.key
쿼리 시 "하위" 키-값 쌍이 리턴되는 상위 키를 지정합니다. 주요 키 경로는 부분 경로여야 하며 보조 키 경로는 비어 있어야 합니다. 선택사항입니다.

kv.avro.json.key
Avro 스키마로 정의된 값을 보유하는 데 사용하는 보조 키 이름입니다. 보조키가 정의되지 않은 경우(일반적임), 기본값으로 "value"를 사용합니다. 매개변수가 정의되지 않은 경우 값이 JSON 문자열로 리턴됩니다. 선택사항입니다.

kv.avro.json.keys.mapping.column

주요/보조 키-값 쌍의 Hive 열 이름을 정의합니다. Hive 열의 유형은 map<string,string>이어야 합니다. 선택사항이며,

TABLEPROPERTIES 매개변수

kv.host.port

Oracle NoSQL 데이터베이스의 IP 주소 및 포트 번호입니다. 필수입니다.

kv.name

Oracle NoSQL 키-값 저장소의 이름입니다. 필수입니다.

예제: 단순 Avro 스키마

데이터 레이아웃은 Apache Avro 직렬화 프레임워크를 사용하여 모델링됩니다. 이 접근법을 따르려면 다음과 같이 Avro 스키마를 작성하십시오.

```
{"type":"record",
 "name":"DrugSchema",
 "namespace":"avro",
 "fields":[
        {"name":"id","type":"string","default":""},
        {"name":"age","type":"string","default":""},
        {"name":"sex","type":"string","default":""},
        {"name":"bp","type":"string","default":""},
        {"name":"drug","type":"string","default":""}
]}
```

이 스키마는 Oracle NoSQL 데이터베이스에 등록되어야 하며 작성된 데이터에는 아래와 같이 스키마에 대한 참조가 포함되어야 합니다.

```
put -key /drugstore_avro/1 -value
   "{\"id\":\"1\",\"age\":\"23\",\"sex\":\"F\",\"bp\":\"HIGH\",\"drug\":\"drugY\"}"
   -json avro.DrugSchema
put -key /drugstore_avro/2 -value
   "{\"id\":\"2\",\"age\":\"47\",\"sex\":\"M\",\"bp\":\"LOW\",\"drug\":\"drugC\"}"
   -json avro.DrugSchema
put -key /drugstore_avro/3 -value
   "{\"id\":\"3\",\"age\":\"47\",\"sex\":\"M\",\"bp\":\"LOW\",\"drug\":\"drugC\"}"
   -json avro.DrugSchema
put -key /drugstore_avro/4 -value
   "{\"id\":\"4\",\"age\":\"28\",\"sex\":\"F\",\"bp\":\"NORMAL\",\"drug\":\"drugX\"}"
   -json avro.DrugSchema
put -key /drugstore_avro/5 -value
   "{\"id\":\"5\",\"age\":\"61\",\"sex\":\"F\",\"bp\":\"LOW\",\"drug\":\"drugY\"}"
   -json avro.DrugSchema
```

Hive에 데이터를 공개하려면 외부 테이블을 작성하고 SERDEPROPERTIES 섹션에 추가 특성 **kv.avro.json.key**를 지정하십시오. 이 특성 값은 보조 키의 이름이며 보조 키가 정의되지 않은 경우 사전 정의된 이름 **value**입니다.

```
CREATE EXTERNAL TABLE oracle_json(id string, age string, sex string, bp string, drug string)
  STORED BY 'com.ibm.spss.hcatalog.OracleKVStorageHandler'
  WITH SERDEPROPERTIES ("kv.major.keys.mapping" = "drugstore_avro,keyid",
        "kv.parent.key"="/drugstore_avro","kv.avro.json.key" = "value")
  TBLPROPERTIES ("kv.host.port" = "<hostname>:5000", "kv.name" = "kvstore");
```

oracle_json에서 select *를 실행하면 다음 결과가 발생합니다.

```
select * from oracle_json;

1 23 F HIGH drugY
5 61 F LOW drugY
3 47 M LOW drugC
2 47 M LOW drugC
4 28 F NORMAL drugX
```

oracle_json 테이블을 Analytic Server 콘솔에서 사용하여 Oracle NoSQL 데이터 소스를 작성할 수 있습니다.

예제: 복합 키

이제 다음 Avro 스키마를 사용할 수 있습니다.

```
{ "type": "record",
  "name": "DrugSchema",
  "namespace":"avro",
  "fields": [
      {"name": "age", "type": "string", "default": ""}, // age
      {"name": "bp", "type":"string","default":""},     // blood pressure
      {"name": "drug", "type": "int", "default": ""},    // drug administered
]}
```

또한 키는 다음과 같이 모델링된다고 가정합니다.

```
/u/<sex (M/F)>/<patient ID>
```

그리고 다음 명령을 사용하여 데이터 저장소를 작성합니다.

```
put -key /u/F/1 -value
  "{\"age\":\"23\",\"bp\":\"HIGH\",\"drug\":\"drugY\"}" -json avro.DrugSchema
put -key /u/M/2 -value
  "{\"age\":\"47\",\"bp\":\"LOW\",\"drug\":\"drugC\"}" -json avro.DrugSchema
put -key /u/M/3 -value
  "{\"age\":\"47\",\"bp\":\"LOW\",\"drug\":\"drugC\"}" -json avro.DrugSchema
put -key /u/F/4 -value
  "{\"age\":\"28\",\"bp\":\"NORMAL\",\"drug\":\"drugX\"}" -json avro.DrugSchema
put -key /u/F/5 -value
  "{\"age\":\"61\",\"bp\":\"LOW\",\"drug\":\"drugY\"}" -json avro.DrugSchema
```

주요 키의 성별 및 사용자 ID 정보를 보존하려면 SERDEPROPERTIES 매개변수 **kv.avro.json.keys.mapping.column**을 추가하여 테이블을 작성해야 합니다. 매개변수 값은 map<string,string> 유형의 Hive 열 이름이어야 합니다. 맵의 키는 **kv.*.keys.mapping** 특성에 지정된 레코드 키 이름이며 이 값은 실제 키 값입니다. 테이블 작성 DDL이 아래에 표시됩니다.

```
CREATE EXTERNAL TABLE oracle_user
   (keys map<string,string>, age string, bp string, drug string)
  STORED BY 'com.ibm.spss.hcatalog.OracleKVStorageHandler'
  WITH SERDEPROPERTIES ("kv.major.keys.mapping" = "DrugSchema,sex,patientid",
      "kv.parent.key" = "/u",
      "kv.avro.json.key" = "value",
      "kv.avro.json.keys.mapping.column" = "keys")
  TBLPROPERTIES ("kv.host.port" = "<hostname>:5000", "kv.name" = "kvstore");
```

oracle_user에서 select *를 실행하면 다음 결과가 발생합니다.

```
select * from
        oracle_user; {"user":"u","gender":"m"," userid":"125"} joe smith 77 13
      {"user":"u","gender":"m"," userid":"129"} jeff smith 67 27
      {"user":"u","gender":"m"," userid":"127"} jim smith 78 11
      {"user":"u","gender":"f"," userid":"131"} jen schmitt 70 20
      {"user":"u","gender":"m"," userid":"130"} jed schmidt 60 31
      {"user":"u","gender":"f"," userid":"128"} jan smythe 79 10
      {"user":"u","gender":"f"," userid":"126"} jess smith 76 12
```

oracle_user 테이블을 Analytic Server 콘솔에서 사용하여 Oracle NoSQL 데이터 소스를 작성할 수 있습니다. Avro 스키마의 열 이름뿐만 아니라 sex 및 patientid 키도 데이터 소스에서 해당 필드를 정의하는 데 사용할 수 있습니다.

범위 스캔

Analytic Server는 상위 키의 범위를 추가로 제한할 수 있도록 하위 범위 외에도 주요 키의 상위 접두부를 기반으로 범위 스캔을 지원합니다.

상위 키는 리턴될 "하위" 키-값 쌍의 접두부를 지정합니다. 접두부가 비어 있으면 저장소에서 모든 키가 페치됩니다. 접두부가 비어 있지 않으면 주요 키 경로가 부분 경로여야 하며 보조 키 경로는 비어 있어야 합니다. 상위 키는 **com.ibm.spss.ae.hcatalog.range.parent** 데이터 소스 속성으로 저장됩니다.

하위 범위는 하위 범위의 주요 경로 구성요소에 대한 상위 키의 범위를 추가로 제한합니다. 하위 범위 시작 키는 **com.ibm.spss.ae.hcatalog.range.start**로 저장되며 하위 범위 종료 키는 **com.ibm.spss.ae.hcatalog.range.end**로 저장됩니다. 시작 키의 길이는 종료 키와 같거나 짧아야 합니다. 하위 범위 매개변수는 선택적입니다.

XML 데이터 소스

Analytic Server는 HCatalog를 통한 XML 데이터 지원을 제공합니다.

예제

1. 다음 규칙에 따라 Hive DDL(Data Definition Language)을 통해 XML 스키마를 Hive 데이터 유형으로 맵핑합니다.

   ```
   CREATE [EXTERNAL] TABLE <table_name> (<column_specifications>)
   ROW FORMAT SERDE "com.ibm.spss.hive.serde2.xml.XmlSerDe"
   WITH SERDEPROPERTIES (
       ["xml.processor.class"="<xml_processor_class_name>",]
       "column.xpath.<column_name>"="<xpath_query>",
   ```

```
     ...
    ["xml.map.specification.<element_name>"="<map_specification>"
     ...
    ]
)
STORED AS
    INPUTFORMAT "com.ibm.spss.hive.serde2.xml.XmlInputFormat"
    OUTPUTFORMAT "org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat"
[LOCATION "<data_location>"]
TBLPROPERTIES (
    "xmlinput.start"="<start_tag ",
    "xmlinput.end"="<end_tag>"
);
```

참고: XML 파일을 Bz2 압축으로 압축한 경우 INPUTFORMAT을 com.ibm.spss.hive.serde2.xml.SplittableXmlInputFormat으로 설정해야 합니다. CMX 압축을 사용하여 압축된 경우, com.ibm.spss.hive.serde2.xml.CmxXmlInputFormat으로 설정해야 합니다.

예를 들어, 다음 XML이 있다고 가정합니다.

```
<records>
  <record customer_id="0000-JTALA">
    <demographics>
      <gender>F</gender>
      <agecat>1</agecat>
      <edcat>1</edcat>
      <jobcat>2</jobcat>
      <empcat>2</empcat>
      <retire>0</retire>
      <jobsat>1</jobsat>
      <marital>1</marital>
      <spousedcat>1</spousedcat>
      <residecat>4</residecat>
      <homeown>0</homeown>
      <hometype>2</hometype>
      <addresscat>2</addresscat>
    </demographics>
    <financial>
      <income>18</income>
      <creddebt>1.003392</creddebt>
      <othdebt>2.740608</othdebt>
      <default>0</default>
    </financial>
  </record>
</records>
```

...이 경우 다음 Hive DDL에서 이 XML을 표시할 수 있습니다.

```
CREATE TABLE xml_bank
   (customer_id STRING, demographics map<string,string>, financial map<string,string>)
ROW FORMAT SERDE 'com.ibm.spss.hive.serde2.xml.XmlSerDe'
WITH SERDEPROPERTIES (
    "column.xpath.customer_id"="/record/@customer_id",
    "column.xpath.demographics"="/record/demographics/*",
    "column.xpath.financial"="/record/financial/*"
)
STORED AS
```

```
    INPUTFORMAT 'com.ibm.spss.hive.serde2.xml.XmlInputFormat'
    OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat'
TBLPROPERTIES (
    "xmlinput.start"="<record customer",
    "xmlinput.end"="</record>"
);
```

자세한 정보는『XML 대 Hive 데이터 유형 맵핑』을 참조하십시오.

2. Analytic Server 콘솔에서 HCatalog 컨텐츠 유형을 사용하는 Analytic Server 데이터 소스를 작성하십시오.

제한사항

- 현재 XPath 1.0 지정 사항만 지원됩니다.
- 요소 및 속성의 규정된 이름에서 로컬 파트는 Hive 필드 이름을 처리할 때 사용합니다. 네임스페이스 접두부는 무시합니다.

XML 대 Hive 데이터 유형 맵핑: XML로 모델링된 데이터는 아래 표시된 규칙을 사용하여 Hive 데이터 유형으로 변환할 수 있습니다.

구조

XML 요소를 Hive 구조 유형으로 직접 맵핑하여 모든 속성을 데이터 멤버로 만들 수 있습니다. 요소의 컨텐츠는 기본 또는 복합 유형의 추가 멤버가 됩니다.

XML 데이터

```
<result name="ID_DATUM">03.06.2009</result>
```

Hive DDL 및 원시 데이터

```
struct<name:string,result:string>
```

```
{"name":"ID_DATUM", "result":"0.3.06.2009"}
```

배열

요소의 XML 시퀀스는 기본 또는 복합 유형의 Hive 배열로 표시될 수 있습니다. 다음 예제에서는 XML <result> 요소의 컨텐츠를 사용하여 문자열 배열을 정의하는 방법을 보여줍니다.

XML 데이터

```
<result>03.06.2009</result>
<result>03.06.2010</result>
<result>03.06.2011</result>
```

Hive DDL 및 원시 데이터

```
result array<string>
```

```
{"result":["03.06.2009","03.06.2010",...]}
```

맵

XML 스키마는 맵에 원시 지원을 제공하지 않습니다. XML의 모델링 맵에 대한 세 가지 공통 접근법이 있습니다. 여러 접근법을 사용하기 위해 다음 구문을 사용합니다.

```
"xml.map.specification.<element_name>"="<key>-><value>"
```

이 구문의 설명은 다음과 같습니다.

element_name
맵 항목으로 취급할 XML 요소의 이름입니다.

key 맵 항목 키 XML 노드

value 맵 항목 값 XML 노드

지정된 XML 요소의 맵 지정 사항은 Hive 테이블 작성 DDL의 SERDEPROPERTIES 섹션에 정의해야 합니다. 키와 값은 다음 구문을 사용하여 정의할 수 있습니다.

@attribute
@attribute 지정 사항을 사용하면 속성 값을 맵의 값 또는 키로 사용할 수 있습니다.

element
요소 이름은 키 또는 값으로 사용할 수 있습니다.

#content
요소의 컨텐츠는 키 또는 값으로 사용할 수 있습니다. 맵 키는 기본 유형이어야 하므로 복합 컨텐츠가 문자열로 변환됩니다.

XML 및 해당 Hive DDL과 원시 데이터로 맵을 표시하는 접근법은 다음과 같습니다.

요소 이름 대 컨텐츠
요소 이름은 키로 컨텐츠는 값으로 사용됩니다. 이는 공통 기술 중 하나이며 기본적으로 XML을 Hive 맵 유형으로 맵핑할 때 사용합니다. 이 접근법은 문자열 유형의 맵 키만 사용할 수 있는 제한이 있습니다.

XML 데이터

```
<entry1>value1</entry1>
<entry2>value2</entry2>
<entry3>value3</entry3>
```

맵핑, Hive DDL 및 원시 데이터

이 경우, 기본적으로 요소 이름을 키로 컨텐츠를 값으로 사용하므로 맵핑을 지정하지 않아도 됩니다.

```
result map<string,string>
```

```
{"result":{"entry1": "value1", "entry2": "value2", "entry3": "value3"}}
```

속성 대 요소 컨텐츠
속성 값을 키로 요소 컨텐츠를 값으로 사용합니다.

XML 데이터

```
<entry name="key1">value1</entry>
<entry name="key2">value2</entry>
<entry name="key3">value3</entry>
```

맵핑, Hive DDL 및 원시 데이터

```
"xml.map.specification.entry"="@name->#content"

result map<string,string>

{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

속성 대 속성

XML 데이터

```
<entry name="key1" value="value1"/>
<entry name="key2" value="value2"/>
<entry name="key3" value="value3"/>
```

맵핑, Hive DDL 및 원시 데이터

```
"xml.map.specification.entry"="@name->@value"

result map<string,string>

{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

복합 컨텐츠

기본 유형으로 사용하는 복합 컨텐츠는 <string>이라는 루트 요소를 추가하여 유효한 XML 문자열로 변환됩니다. 다음의 XML을 고려해 보십시오.

```
<dataset>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</dataset>
```

XPath 표현식 /dataset/*를 사용하면 여러 <value> XML 노드가 리턴됩니다. 대상 필드가 기본 유형인 경우, 구현 시 <string> 루트 노드를 추가하여 쿼리 결과가 유효한 XML로 변환됩니다.

```
<string>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</string>
```

참고: 쿼리 결과가 단일 XML 요소이면 구현 시 루트 요소 <string>을 추가하지 않습니다.

텍스트 컨텐츠

XML 요소의 텍스트 컨텐츠에 공백만 있으면 이를 무시합니다.

미리보기 및 메타데이터(데이터 소스)

미리보기 및 메타데이터를 클릭하면 레코드 샘플과 데이터 소스의 데이터 모델이 표시됩니다. 여기서 기본 메타데이터 정보를 검토할 수 있습니다.

미리보기

미리보기 탭은 레코드 및 해당 필드 값의 소규모 표본을 표시합니다.

편집

편집 탭에서는 기본 필드 메타데이터가 표시됩니다. 파일 컨텐츠 유형의 데이터 소스는 작은 표본 레코드에서 데이터 모델을 작성하므로 이 탭에서 필드 메타데이터를 수동으로 편집할 수 있습니다. HCatalog 컨텐츠 유형의 데이터 소스는 HCatalog 필드 맵핑을 기반으로 데이터 모델이 생성되므로 이 탭에서 필드 저장 공간을 편집할 수 없습니다.

필드 필드 이름을 두 번 클릭하여 필드를 편집할 수 있습니다.

측정 지정된 필드에서 데이터의 특성에 대해 설명하는 데 사용되는 측정 수준입니다.

역할 필드가 머신 학습 프로세스에 대한 입력(예측변수 필드)인지 또는 대상(예측 필드)인지를 모델링 노드에 알리는 데 사용됩니다. 둘 다 및 없음도 파티션과 함께 사용 가능한 역할입니다. 파티션은 학습, 검증 및 검증을 위해 레코드를 별도의 표본으로 파티셔닝하는 데 사용되는 필드를 표시합니다. 분할 값은 각각의 가능한 필드 값에 별도의 모델이 작성됨을 지정합니다. 빈도는 필드 값을 각 레코드의 빈도 가중치로 사용하도록 지정합니다. 레코드 ID는 결과에서 레코드를 식별하는 데 사용합니다.

저장 공간

저장 공간은 데이터가 필드에 저장되는 방법을 설명합니다. 예를 들어, 값이 1 및 0인 필드는 정수 데이터를 저장합니다. 이는 데이터 사용에 대해 설명하는 측정 수준과 다르며 저장 공간에 영향을 미치지 않습니다. 예를 들어, 값이 1 및 0인 정수 필드에 대한 측정 수준을 플래그로 설정할 수 있습니다. 일반적으로 1 = True 및 0 = False를 표시합니다.

값 범주형 측정이 있는 필드의 개별 값이나 연속 측정이 있는 필드의 값의 범위를 표시합니다.

구조 필드의 레코드에 단일 값(기본)이 있는지 값 목록이 있는지 표시합니다.

깊이 목록의 깊이를 표시합니다. 0은 기본 목록이며 1은 목록의 목록이며 이와 같이 계속됩니다.

모든 데이터 값 스캔

이 옵션으로 데이터 소스 데이터 값의 스캔을 시작 및 취소하여 카테고리 값과 범위 한계를 판별할 수 있습니다. 스캔이 진행 중이면 단추를 클릭하여 **데이터 스캔 취소**를 실행하십시오. 모든 데이터 값을 스캔하면 메타데이터가 올바른지 확인할 수 있지만 데이터 소스에 여러 필드와 레코드가 있으면 시간이 다소 걸릴 수 있습니다.

프로젝트

프로젝트는 입력을 저장하고 작업 결과에 액세스하기 위한 작업공간입니다. 프로젝트는 파일 및 폴더를 포함하기 위한 최상위 레벨 조직 구조입니다. 프로젝트는 개별 사용자 및 그룹과 공유할 수 있습니다.

프로젝트 목록

기본 프로젝트 페이지에서는 현재 사용자가 멤버로 속해 있는 프로젝트의 목록을 제공합니다.

- 프로젝트의 이름을 클릭하여 세부사항을 표시하고 특성을 편집합니다.
- 검색 영역에 입력하여 목록을 필터링해서 이름에 검색 문자열이 포함된 프로젝트만 표시합니다.
- **새로 만들기**를 클릭하면 **새 프로젝트 추가** 대화 상자에 지정하는 이름으로 새 프로젝트가 작성됩니다. 프로젝트에 지정할 수 있는 이름에 대한 제한사항은 31 페이지의 『이름 지정 규칙』의 내용을 참조하십시오.
- 선택한 프로젝트를 제거하려면 **삭제**를 클릭하십시오. 이 조치를 수행하면 프로젝트가 제거되고 HDFS에서 프로젝트와 연관된 모든 데이터가 삭제됩니다.
- 목록을 업데이트하려면 **새로 고치기**를 클릭하십시오.

개별 프로젝트 세부사항

컨텐츠 영역은 접을 수 있는 **세부사항**, **공유**, **파일** 및 **버전** 섹션으로 나누어져 있습니다.

세부사항

이름 프로젝트의 이름을 표시하는 편집 가능한 텍스트 필드입니다.

표시 이름
다른 애플리케이션에서 표시되는 프로젝트의 이름을 보여주는 편집 가능한 텍스트 필드입니다. 비어 있는 경우, 이름을 표시 이름으로 사용합니다.

설명 프로젝트에 대한 설명 텍스트를 제공하는 편집 가능한 텍스트 필드입니다.

보관할 버전 수
버전 수가 지정된 수를 초과하는 경우 자동으로 가장 이전에 커미트된 프로젝트 버전을 삭제합니다. 기본값은 25입니다.

참고: 정리 프로세스는 즉시 실행되지 않지만 백그라운드로 20분 마다 실행됩니다.

공용 여부
모든 사용자가 프로젝트를 볼 수 있는지(선택됨) 또는 사용자 및 그룹이 멤버로 명시적으로 추가되어야 하는지(선택 취소) 여부를 표시하는 선택란입니다.

저장을 클릭하면 설정의 현재 상태가 저장됩니다.

공유 사용자와 그룹을 작성자나 뷰어로 추가하여 프로젝트를 공유할 수 있습니다.

- 텍스트 상자에 입력하면 이름에 검색 문자열이 포함되어 있는 사용자 및 그룹이 필터링됩니다. 공유 수준을 선택하고 **멤버 추가**를 클릭하여 멤버 목록에 추가하십시오.
  - 작성자는 프로젝트의 전체 멤버이고 프로젝트와 프로젝트 내에 있는 폴더 및 파일을 수정할 수 있습니다. 이러한 그룹의 사용자와 멤버는 IBM® SPSS® Modeler를 통해 Analytic Server에 연결할 때 이 프로젝트에 대한 쓰기(Analytic Server 내보내기 노드) 액세스 권한이 있습니다.
  - 뷰어는 프로젝트의 폴더와 파일을 볼 수 있으며 프로젝트 내에서 오브젝트에 대한 데이터 소스를 정의할 수 있지만 프로젝트를 수정할 수는 없습니다.

- 작성자를 제거하려면 작성자 목록에서 사용자 또는 그룹을 선택하고 **멤버 제거**를 클릭하십시오.

참고: 관리자는 명확하게 멤버로 나열되는지 여부에 관계없이 모든 프로젝트에 대한 읽기 및 쓰기 액세스 권한을 가집니다.

참고: 공유에 대한 변경사항은 즉시 자동으로 적용됩니다.

파일

프로젝트 구조 분할창

오른쪽 분할창은 현재 선택된 프로젝트에 대한 프로젝트/폴더 구조를 표시합니다. 폴더 구조는 찾아볼 수 있지만 단추를 통하는 경우를 제외하고 편집 불가능합니다.

- 선택한 파일을 로컬 파일 시스템으로 다운로드하려면 **로컬 파일 시스템에 파일 다운로드**를 클릭하십시오.
- 선택한 파일/폴더를 제거하려면 **선택한 파일 삭제**를 클릭하십시오.

파일 뷰어

현재 프로젝트의 폴더 구조를 보여줍니다. 폴더 구조는 정의된 프로젝트 내에서만 편집할 수 있습니다. 즉, **프로젝트** 모드의 루트 수준에서 파일 추가, 폴더 작성 또는 항목 삭제를 수행할 수 없습니다. 프로젝트 삭제를 작성하려면 프로젝트 목록으로 돌아가십시오.

- **HDFS로 파일 업로드**를 클릭하여 파일을 현재 프로젝트/하위 폴더로 업로드하십시오.
- **새 폴더 작성**을 클릭하면 **새 폴더 이름** 대화 상자에서 지정한 이름의 새 폴더가 현재 폴더 아래에 작성됩니다.
- 선택한 파일을 로컬 파일 시스템으로 다운로드하려면 **로컬 파일 시스템에 파일 다운로드**를 클릭하십시오.
- 선택한 파일/폴더를 제거하려면 **선택한 파일 삭제**를 클릭하십시오.

버전

프로젝트는 파일 및 폴더 컨텐츠의 변경사항에 따라 버전화됩니다. 프로젝트 속성(예: 설명, 설명의 공용 여부 및 설명의 공유자)의 변경에는 새 버전이 필요하지 않습니다. 파일이나 폴더를 추가, 수정 또는 삭제할 경우 새 버전이 필요합니다.

프로젝트 버전화 테이블

테이블은 기존 프로젝트 버전, 해당 작성 및 커미트 날짜, 각 버전의 담당 사용자 및 상위 버전을 표시합니다. 상위 버전은 선택된 버전의 기반이 되는 버전입니다.

- 선택한 프로젝트 버전 컨텐츠의 변경사항을 작성하려면 **잠금**을 클릭하십시오.
- 프로젝트에 작성된 모든 변경사항을 저장하고 이 버전을 프로젝트의 현재 표시 가능 상태로 만들려면 **커미트**를 클릭하십시오.
- 잠긴 프로젝트에 작성된 모든 변경사항을 버리고 프로젝트의 표시 가능 상태를 가장 최근에 커미트된 버전으로 되돌리려면 **버리기**를 클릭하십시오.
- 선택한 버전을 제거하려면 **제거**를 클릭하십시오.

사용자 관리

관리자는 사용자 페이지에서 사용자와 그룹의 역할을 관리할 수 있습니다.

컨텐츠 영역은 접을 수 있는 **세부사항**과 **프린시펄** 섹션으로 나누어져 있습니다.

세부사항

이름 편집할 수 없는 텍스트 필드이며 테넌트 이름이 표시됩니다.

설명 편집할 수 있는 텍스트 필드이며 테넌트에 대한 설명 텍스트를 입력할 수 있습니다.

URL Analytic Server 콘솔을 통해 테넌트에 로그인할 수 있도록 사용자에게 제공되는 URL입니다.

프린시펄

프린시펄은 구성 중에 설정한 보안 제공자에게서 얻은 사용자와 그룹입니다. 프린시펄의 역할을 관리자나 사용자로 변경할 수 있습니다.

메트릭 테넌트의 자원 한계를 구성할 수 있습니다. 현재 테넌트에서 사용하는 디스크 공간을 보고합니다.

- 테넌트의 최대 디스크 공간 할당량을 설정할 수 있습니다. 이 한계에 도달하면 디스크 공간을 비워서 테넌트 디스크 공간 사용량이 할당량 미만으로 내려가야 이 테넌트의 디스크에 데이터를 작성할 수 있습니다.
- 테넌트의 디스크 공간 경고 레벨을 설정할 수 있습니다. 할당량을 초과하면 디스크 공간을 비워서 테넌트 디스크 공간 사용량이 할당량 미만으로 내려가야만 이 테넌트의 프린시펄에서 분석 작업을 제출할 수 있습니다.
- 이 테넌트에서 한 번에 실행할 수 있는 병렬 작업의 최대 수를 설정할 수 있습니다. 할당량을 초과하면 현재 실행 중인 작업이 완료되어야만 이 테넌트의 프린시펄에서 분석 작업을 제출할 수 있습니다.
- 데이터 소스에서 포함할 수 있는 최대 필드 수를 설정할 수 있습니다. 데이터 소스를 작성하거나 업데이트할 때마다 한계가 선택됩니다.
- 데이터 소스에서 포함할 수 있는 최대 레코드 수를 설정할 수 있습니다. 데이터 소스를 작성하거나 업데이트할 때(예를 들어, 새 파일을 추가하거나 파일의 설정을 변경할 때)마다 한계가 선택됩니다.
- 최대 파일 *크기*(MB)를 설정할 수 있습니다. 파일 업로드 시 한계가 선택됩니다.

이름 지정 규칙

데이터 소스나 프로젝트와 같이 Analytic Server에서 고유한 이름을 지정할 수 있는 모든 이름에는 다음 규칙이 적용됩니다.

- 이름은 같은 유형의 오브젝트 안에서 고유해야 합니다. 예를 들어, 두 개의 데이터 소스에 모두 insuranceClaims라는 이름을 지정할 수 없지만 데이터 소스와 프로젝트에 각각 insuranceClaims라는 이름을 지정할 수는 있습니다.

- 이름은 대소문자를 구분합니다. 예를 들어, insuranceClaims와 InsuranceClaims는 고유한 이름으로 취급됩니다.
- 이름의 선행 및 후행 공백은 무시합니다.
- 다음 문자는 이름에 사용할 수 없습니다.

  ~, #, %, &, *, {, }, \\, :, <, >, ?, /, |, ", \t, \r, \n

제 3 장 SPSS Modeler 통합

SPSS Modeler는 분석에 시각적으로 접근하는 데이터 마이닝 워크벤치입니다. 데이터 소스 액세스에서 레코드 병합을 거쳐 새 파일 작성 또는 모델 빌드에 이르기까지 한 작업에 포함된 각 별개의 조치가 캔버스에 노드로 표시됩니다. 이러한 조치를 함께 연결하여 분석 스트림을 형성합니다.

Analytic Server 데이터 소스에 대해 실행할 수 있는 SPSS Modeler 스트림을 구성하려면 Analytic Server 소스 노드로 시작하십시오. SPSS Modeler는 가능한 많은 스트림을 Analytic Server에 되돌리고 필요하면 레코드의 서브세트를 가져와서 SPSS Modeler 서버에서 "로컬로" 스트림 실행을 완료합니다. SPSS Modeler가 Analytic Server 스트림 특성에서 다운로드하는 최대 레코드 수를 설정할 수 있습니다.

분석이 종료되어 레코드가 다시 HDFS에 기록된 경우 Analytic Server 내보내기 노드로 스트림을 완료하십시오.

이러한 노드에 대한 자세한 정보는 SPSS Modeler 문서를 참조하십시오.

지원되는 노드

많은 SPSS Modeler 노드가 HDFS에서 실행되도록 지원되지만 특정 노드의 실행에는 몇 가지 차이가 있을 수 있으며 일부는 현재 지원되지 않습니다. 이 토픽에서는 현재 수준의 지원에 대해 자세히 설명합니다.

일반

- 일반적으로 따옴표로 묶인 Modeler 필드 이름 내에서 허용 가능한 일부 문자가 Analytic Server에서는 허용되지 않습니다.
- Modeler 스트림을 Analytic Server에서 실행하려면 이 스트림이 한 개 이상의 Analytic Server 소스 노드로 시작하여 단일 모델링 노드나 Analytic Server 내보내기 노드로 끝나야 합니다.
- 정수보다는 실수로 연속형 대상의 저장 공간을 설정하도록 권장합니다. 스코어링 모델은 항상 연속형 대상에 대한 결과 데이터 파일에 실수 값을 쓰지만 스코어의 결과 데이터 모델은 대상의 저장 공간을 따릅니다. 따라서 연속형 대상에 정수 저장 공간이 있는 경우 쓰여진 값과 스코어의 데이터 모델 간에 불일치가 발생하고 이 불일치로 인해 스코어가 지정된 데이터를 읽으려고 시도할 때 오류가 발생합니다.

소스

- Analytic Server 소스 노드가 아닌 노드로 시작하는 스트림은 로컬에서 실행됩니다.

레코드 조작

스트리밍 TS 및 공간 시간 상자 노드를 제외한 모든 레코드 조작이 지원됩니다. 지원되는 노드 기능에 대한 추가 정보가 뒤에 나옵니다.

선택

- 도출 노드에서 지원하는 같은 기능 세트를 지원합니다.

표본

- 블록 레벨 표본추출은 지원되지 않습니다.
- 복합 표본추출 방법은 지원되지 않습니다.

집계

- 연속 키는 지원되지 않습니다. 데이터를 정렬하도록 설정한 기존 스트림을 다시 사용하고 집계 노드에서 이 설정을 사용하는 경우 정렬 노드를 제거하도록 스트림을 변경하십시오.
- 순서 통계(중앙값, 첫 번째 사분위수, 세 번째 사분위수)는 근사한 값으로 계산되며 최적화 탭에서 지원됩니다.

정렬

- 최적화 탭은 지원되지 않습니다.

분산 환경에서는 정렬 노드에서 설정한 레코드 순서를 보존하는 조작의 수가 제한됩니다.

- 정렬 다음에 내보내기 노드가 있으면 정렬된 데이터 소스가 생성됩니다.
- 정렬 다음에 **첫 번째** 레코드 표본추출이 있는 샘플 노드가 있으면 처음 *N*개 레코드가 리턴됩니다.
- 정렬 다음에 **매우 큰 데이터 세트를 위한 최적화** 목표(신경망, 선형, C&R 트리, Quest 또는 CHAID)가 있는 모델링 노드가 있으면, 원래 레코드가 정렬되어 있는 경우 모델 작성 알고리즘으로 도입되는 편향을 방지하기 위해 도출된 임의 번호 키를 정렬하여 레코드를 임의로 다시 섞는 데 유용한 패턴이 됩니다.

일반적으로 정렬 노드는 정렬된 레코드가 필요한 조작에 가능한 가깝게 배치해야 합니다.

병합

- 순서별 병합은 지원되지 않습니다.
- 최적화 탭은 지원되지 않습니다.
- Analytic Server 소스 노드와 병합 노드 사이에 표본 노드 또는 모델 너깃을 배치하는 것은 현재 지원되지 않습니다. 일반적으로 표본 노드의 기능을 대체할 선택 노드를 지정할 수 있습니다.
- Analytic Server는 빈 문자열 키를 결합하지 않습니다. 즉, 병합에 사용 중인 키 중 하나에 빈 문자열이 포함되어 있는 경우 빈 문자열이 포함된 모든 레코드가 병합된 결과에서 삭제됩니다.
- 병합 조작은 상대적으로 느립니다. HDFS에 사용 가능한 공간이 있는 경우 각 스트림에서 데이터 소스를 병합하는 것보다는 한 번 데이터 소스를 병합하고 다음 스트림에서 병합된 소스를 사용하는 것이 훨씬 빠를 수 있습니다.

R 변환

노드의 R 구문은 한 번에 한 개 레코드 조작으로 구성해야 합니다.

필드 조작

전치, 시간 구간 및 히스토리 노드를 제외한 모든 필드 조작이 지원됩니다. 지원되는 노드 기능에 대한 추가 정보가 뒤에 나옵니다.

자동 데이터 준비

- 노드 훈련이 지원되지 않습니다. 훈련된 자동 데이터 준비 노드의 변환을 새 데이터에 적용하는 작업은 지원됩니다.

유형

- 검사 열은 지원되지 않습니다.
- 형식 탭은 지원되지 않습니다.

도출

- 시퀀스 함수를 제외한 모든 도출 함수가 지원됩니다.
- 분할 필드는 이를 분할로 사용하는 동일 스트림에서는 도출할 수 없으므로, 두 개의 스트림(분할 필드를 도출하는 스트림 및 이 필드를 분할로 사용하는 스트림)을 작성해야 합니다.
- 플래그 필드 자체를 비교에 사용할 수 없습니다. 즉, if (flagField) then ... endif를 사용하면 오류가 발생합니다. 임시 해결책으로 if (flagField=trueValue) then ... endif를 사용할 수 있습니다.
- Modeler에서 결과를 일치시키기 위해 ** 연산자를 사용하여 지수를 x**2 대신에 x**2.0과 같은 실수로 지정할 때 권장됩니다.

필러

- 도출 노드에서 지원하는 같은 기능 세트를 지원합니다.

구간화 다음 기능은 지원되지 않습니다.

- 최적 구간화
- 순위
- 바둑판식 -> 바둑판식 배열: 합계
- 바둑판식 -> 등순위: 현재로 유지 및 임의 지정
- 바둑판식 -> 사용자 정의 N: 100을 초과하는 값 및 100 % N이 0이 아닌 임의의 N값입니다.

RFM 분석

- 등순위를 처리하기 위한 "현재로 유지" 옵션이 지원되지 않습니다. 항상 RFM(최근구매일, 구매빈도, 구매금액) 스코어가 Modeler에서 동일한 데이터로 계산된 스코어와 일치하지는 않습니다. 스코어 범위는 동일하지만 스코어 할당(구간 수)은 다를 수 있습니다.

그래프 모든 그래프 노드가 지원됩니다.

모델링 선형, 신경망, C&RT, Chaid, Quest, TCM, TwoStep-AS, STP 및 연관 규칙 모델링 노드가 지원됩니다. 이러한 노드의 기능에 대한 추가적인 참고사항이 뒤따릅니다.

선형 빅 데이터에 대한 모델을 작성할 때는 일반적으로 목표를 매우 큰 데이터 세트로 변경하거나 분할을 지정합니다.

- 기존 PSM 모델의 지속적인 훈련은 지원되지 않습니다.
- 각 분할에 있는 레코드 수가 너무 크지 않도록 분할 필드를 정의한 경우에만 표준 모델 작성 오브젝트를 권장합니다. 여기서 "너무 큰"의 정의는 Hadoop 클러스터에 있는 개별 노드의 기능에 따라 다릅니다. 그에 반해, 분할이 매우 세부적으로 정의되어 모델을 작성하기 위한 레코드가 너무 적지 않은지도 주의 깊게 확인해야 합니다.
- 부스팅 목표가 지원되지 않습니다.
- 배깅 목표가 지원되지 않습니다.
- 레코드 수가 적은 경우 매우 큰 데이터 세트 목표는 권장되지 않습니다. 이는 흔히 모델을 작성하지 않거나 하급 모델을 작성합니다.
- 자동 데이터 준비는 지원되지 않습니다. 이로 인해 누락된 값이 많은 데이터에 대한 모델을 작성하려고 시도할 때 문제점이 발생할 수 있습니다. 일반적으로 이러한 문제점은 자동 데이터 준비의 일부로 전가될 수 있습니다. 임시 해결책은 선택된 결측값을 대치하기 위한 고급 설정이 있는 나무 모형 또는 신경망을 사용하는 것입니다.
- 분할 모델에 대한 정확도 통계는 계산되지 않습니다.

신경망 빅 데이터에 대한 모델을 작성할 때는 일반적으로 목표를 매우 큰 데이터 세트로 변경하거나 분할을 지정합니다.

- 기존 표준 또는 PSM 모델의 지속적인 훈련은 지원되지 않습니다.
- 각 분할에 있는 레코드 수가 너무 크지 않도록 분할 필드를 정의한 경우에만 표준 모델 작성 오브젝트를 권장합니다. 여기서 "너무 큰"의 정의는 Hadoop 클러스터에 있는 개별 노드의 기능에 따라 다릅니다. 그에 반해, 분할이 매우 세부적으로 정의되어 모델을 작성하기 위한 레코드가 너무 적지 않은지도 주의 깊게 확인해야 합니다.
- 부스팅 목표가 지원되지 않습니다.
- 배깅 목표가 지원되지 않습니다.
- 레코드 수가 적은 경우 매우 큰 데이터 세트 목표는 권장되지 않습니다. 이는 흔히 모델을 작성하지 않거나 하급 모델을 작성합니다.
- 데이터에 결측값이 많은 경우 고급 설정을 사용하여 결측값을 대치하십시오.
- 분할 모델에 대한 정확도 통계는 계산되지 않습니다.

C&R 트리, CHAID 및 Quest

빅 데이터에 대한 모델을 작성할 때는 일반적으로 목표를 매우 큰 데이터 세트로 변경하거나 분할을 지정합니다.

- 기존 PSM 모델의 지속적인 훈련은 지원되지 않습니다.
- 각 분할에 있는 레코드 수가 너무 크지 않도록 분할 필드를 정의한 경우에만 표준 모델 작성 오브젝트를 권장합니다. 여기서 "너무 큰"의 정의는 Hadoop 클러스터에 있는 개별 노

드의 기능에 따라 다릅니다. 그에 반해, 분할이 매우 세부적으로 정의되어 모델을 작성하기 위한 레코드가 너무 적지 않은지도 주의 깊게 확인해야 합니다.

- 부스팅 목표가 지원되지 않습니다.
- 배깅 목표가 지원되지 않습니다.
- 레코드 수가 적은 경우 매우 큰 데이터 세트 목표는 권장되지 않습니다. 이는 흔히 모델을 작성하지 않거나 하급 모델을 작성합니다.
- 대화형 세션은 지원되지 않습니다.
- 분할 모델에 대한 정확도 통계는 계산되지 않습니다.

모델 스코어링

모델링을 지원하는 모든 모델은 스코어링에도 지원됩니다. 또한 다음 노드에 대해 로컬로 작성된 모델 너깃이 스코어링에 지원됩니다. C&RT, Quest, CHAID, 선형, 신경망(모델이 표준인지 Boosted Bagged 인지 매우 큰 데이터 세트용인지 여부에 관계 없음), 회귀분석, C5.0, 로지스틱, Genlin, GLMM, Cox, SVM, Bayes Net, TwoStep, KNN, 의사결정 목록, 판별, 자체 학습, 이상 항목 발견, Apriori, Carma, K-Means, Kohonen, R, Text Mining.

- 원시 또는 조정된 성향은 스코어링되지 않습니다. 임시 해결책으로 다음 표현식으로 도출 노드를 사용하여 수동으로 원시 성향을 계산하면 동일한 효과를 얻을 수 있습니다. if 'predicted-value' == 'value-of-interest' then 'prob-of-that-value' else 1-'prob-of-that-value' endif
- 모델을 스코어링할 때 Analytic Server는 모델에서 사용된 모든 필드가 데이터 세트에 있는지 검사하지 않으므로 Analytic Server에서 실행하기 전에 이것이 맞는지 확인하십시오.

R 너깃의 R 구문은 한 번에 한 개 레코드 조작으로 구성해야 합니다.

결과 행렬, 분석, 데이터 감사, 변환, 통계 및 평균 노드가 지원됩니다.

테이블 노드는 업스트림 조작의 결과가 포함된 임시 Analytic Server 데이터 소스를 작성하여 지원됩니다. 그러면 테이블 노드에서 해당 데이터 소스의 컨텐츠를 확인합니다.

내보내기

스트림은 Analytic Server 소스 노드로 시작하고 Analytic Server 내보내기 노드가 아닌 다른 내보내기 노드로 종료될 수 있지만 HDFS에서 SPSS Modeler Server로, 최종적으로 내보내기 위치로 데이터가 이동합니다.

주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이센스까지 부여하는 것은 아닙니다. 라이센스에 대한 의문사항은 다음으로 문의하십시오.

135-700
서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
고객만족센터
전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 라이센스 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증 없이 이 책을 "현상태대로" 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 언급되는 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(1) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함)간의 정보 교환 및 (2) 교환된 정보의 상호 이용을 목적으로 본 프로그램에 관한 정보를 얻고자 하는 라이센스 사용자는 다음 주소로 문의하십시오.

135-700
서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩
한국 아이.비.엠 주식회사
고객만족센터

이러한 정보는 해당 조건(예를 들면, 사용료 지불 등)하에서 사용될 수 있습니다.

이 정보에 기술된 라이센스가 부여된 프로그램 및 프로그램에 대해 사용 가능한 모든 라이센스가 부여된 자료는 IBM이 IBM 기본 계약, IBM 국제 프로그램 라이센스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 단계의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고는 보증할 수 없습니다. 또한 일부 성능은 추정을 통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 해당 데이터를 본인의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 반드시 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확인할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지 없이 변경될 수 있습니다.

여기에 나오는 모든 IBM의 가격은 IBM이 제시하는 현 소매가이며 통지 없이 변경될 수 있습니다. 실제 판매가는 다를 수 있습니다.

이 정보는 계획 수립 목적으로만 사용됩니다. 이 정보는 기술된 제품이 GA(General Availability)되기 전에 변경될 수 있습니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

이러한 샘플 프로그램 또는 파생 제품의 각 사본이나 그 일부에는 반드시 다음과 같은 저작권 표시가 포함되어야 합니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

이러한 샘플 프로그램 또는 파생 제품의 각 사본이나 그 일부에는 반드시 다음과 같은 저작권 표시가 포함되어야 합니다.

© (귀하의 회사명) (연도). 이 코드의 일부는 IBM Corp.의 샘플 프로그램에서 파생됩니다.

© Copyright IBM Corp. _연도 또는 복수 연도_. All rights reserved.

이 정보를 소프트카피로 확인하는 경우에는 사진과 컬러 삽화가 제대로 나타나지 않을 수도 있습니다.

상표

IBM, IBM 로고 및 ibm.com은 전세계 여러 국가에 등록된 International Business Machines Corp.의 상표 또는 등록상표입니다. 기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다. 현재 IBM 상표 목록은 웹(www.ibm.com/legal/copytrade.shtml)의 "저작권 및 상표 정보"에 있습니다.

Adobe, Adobe 로고, PostScript 및 PostScript 로고는 미국 또는 기타 국가에서 사용되는 Adobe Systems Incorporated의 등록상표 또는 상표입니다.

IT Infrastructure Library는 현재 Office of Government Commerce의 일부인 Central Computer and Telecommunications Agency의 등록상표입니다.

Intel, Intel 로고, Intel Inside, Intel Inside 로고, Intel Centrino, Intel Centrino 로고, Celeron, Intel Xeon, Intel SpeedStep, Itanium 및 Pentium은 미국 또는 기타 국가에서 사용되는 Intel Corporation 또는 그 계열사의 상표 또는 등록상표입니다.

Linux는 미국 또는 기타 국가에서 사용되는 Linus Torvalds의 등록상표입니다.

Microsoft, Windows, Windows NT 및 Windows 로고는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표입니다.

ITIL은 미국 특허청(U.S. Patent and Trademark Office)에 등록된 The Minister for the Cabinet Office의 등록상표 및 등록 공동체 상표입니다.

UNIX는 미국 및 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

Java 및 모든 Java 기반 상표와 로고는 Oracle 및/또는 그 계열사의 상표 또는 등록상표입니다.

Cell Broadband Engine은 미국 또는 기타 국가에서 사용되는 Sony Computer Entertainment, Inc.의 상표이며 이에 따른 라이센스가 적용됩니다.

Linear Tape-Open, LTO 및 LTO 로고, Ultrium 및 Ultrium 로고는 미국 또는 기타 국가에서 사용되는 HP, IBM Corp. 및 Quantum의 상표입니다.