

IRIS New Features

Mobigen, November 20, 2015

목차

1	지원 정보	4
1.1	문서 버전	4
1.2	IRIS 버전	4
1.3	사용 문의 및 기술 정보	4
2	1.5 버전 신규 기능	5
2.1	FTS (Full-Text-Search)	5
2.2	암호화	6
2.3	Hadoop EcoSystem 지원	8
2.4	시스템 정보 제공기능	12
2.5	Session 모니터링 기능	12
2.6	IRIS Client 제공	13
2.7	IRIS 고도화	14
	2.7.1 DTD	14
	2.7.2 IRIS FileSystem	14
	2.7.3 PR / PL	14
2.8	CLI 명령어	15
	2.8.1 IPLUS CMD 명령어	15
	2.8.2 IPLUS CLI 명령어	17
	2.8.3 IRIS 정보조회/수정 관련 명령어	17

2.8.4 System 정보조회 명령어 20

2.8.5 Error 로그 조회 명령어 20

2.8.6 노드 명령어 21

2.8.7 Daemon 프로세스 명령어 21

2.8.8 부가기능 명령어 21

1 지원 정보

1.1 문서 버전

0.1

1.2 IRIS 버전

1.5.1

1.3 사용 문의 및 기술 정보

(주) 모비젠

본사

- 135-280 서울 강남구 대치동 967-3번지 KM빌딩 2층, (주) 모비젠
- T : 02 - 538 - 9360
- F : 02 - 538 - 9369

기술 연구소

- 135-280 서울 강남구 대치동 967-3번지 KM빌딩 5층, (주) 모비젠
- T : 02 - 538 - 9364
- F : 02 - 538 - 9368

모비젠 IRIS 기술 지원

- T : 02 - 538 - 9364
- M : iris@mobigen.com

2 1.5 버전 신규 기능

2.1 FTS (Full-Text-Search)

FTS 기능을 사용하면 문자열 기반 고속 검색을 위해서 단어단위로 인덱스를 생성합니다. FTS 기능을 사용하려면 FTS 전용 SQL 문을 사용해야 합니다.

FTS 기능 사용 테이블 생성

```
CREATE virtual TABLE TEST_TABLE USING FTS4 (  
  k      TEXT,  
  p      TEXT,  
  a      TEXT  
)  
datascope LOCAL  
ramexpire 30  
diskexpire 1440  
partitionkey k  
partitiondate p  
partitionrange 10  
;
```

FTS 기능을 사용해서 데이터를 검색하는 쿼리

```
iplus> SELECT * FROM TEST_TABLE WHERE a MATCH '1';  
Ret : +OK Success  
  
K          P          A  
-----  
k2          20110616000000 1  
k3          20110616000000 1.2  
k5          20110616000000 0.1  
-----  
3 row in set  
0.1774 sec
```

```

iplus> SELECT * FROM TEST_TABLE WHERE a MATCH '^1';
Ret : +OK Success

  K          P          A
=====
k2          20110616000000 1
k3          20110616000000 1.2
=====
2 row in set
0.0457 sec

```

2.2 암호화

- 암호화 스펙
 - 암호화 알고리즘 : AES-256
 - 블록사이즈 : 32 bit
 - HASH 알고리즘 : SHA256
- DB파일 암호화 제공

IRIS 를 DB암호화버전으로 설치시 모든 데이터가 자동으로 암호화되어 저장됩니다. IRIS 를 통해서 조회하면 자동으로 복호화되어 조회됩니다.

- Column 암호화 제공

일부 데이터를 IRIS 를 통해서 조회하더라도 평문이 보이지 않도록 암호화합니다. 비 DB암호화버전 IRIS 도 사용가능합니다. 데이터 로딩시 암호화할 Column을 선택하면 IRIS 내부에서 암호화해서 저장합니다.

ENCRYPT 암호화 방법은 SELECT QUERY 에서 DECRYPT 함수를 사용해서 원문을 볼 수 있습니다. 로딩시 컬럼이름 앞에 @ 을 붙여서 표시합니다.

```

[iris@]$ cat LOCAL_TEST_TABLE.ct1
k
P
@a

[iris@]$ cat LOCAL_TEST_TABLE.dat
k,20110616000000,b
k,20110616000000,c

[iris@]$ python 06.load.py
+OK SUCCESS. success count : 2

iplus> select k,p,a from LOCAL_TEST_TABLE;
Ret : +OK Success

  K          P          A
=====
k          20110616000000 xjYSHpzXiQm6T/fUP5X4NPnn7...
k          20110616000000 gJjVSp9y6Qfnf16t09XaHydq9...
=====
2 row in set
0.1116 sec

iplus> select k,p,decrypt(a) from LOCAL_TEST_TABLE;
Ret : +OK Success

  K          P          DECRYPT(A)
=====
k          20110616000000 b
k          20110616000000 c
=====
2 row in set
0.0505 sec

iplus>

```

HASH 암호화 방법은 복호화가 불가능한 HASH 암호화를 적용합니다. 로딩시 컬럼이름 앞에 # 을 붙여서 표시합니다.

```

[iris@]$ cat LOCAL_TEST_TABLE.ct1
k
p
#a

[iris@]$ cat LOCAL_TEST_TABLE.dat
k,20110616000000,b
k,20110616000000,c

[iris@]$ python 06.load.py
+OK SUCCESS. success count : 2

iplus> select k,p,a from LOCAL_TEST_TABLE;
Ret : +OK Success

  K           P           A
=====
k           20110616000000  3e23e8160039594a33894f656...
k           20110616000000  2e7d2c03a9507ae265ecf5b53...
=====
2 row in set
0.1116 sec

iplus>

```

2.3 Hadoop EcoSystem 지원

IRIS 에 저장된 데이터를 Hadoop EcoSystem 을 통해서 검색할 수 있습니다. IRIS 에서 제공하는 EcoSystem 용 드라이버를 사용해서 IRIS 를 HDFS 인터페이스로 추상화해서 연결합니다. IRIS 로부터는 데이터를 읽을수만 있기 때문에 데이터 분석을 수행하기 위해 HDFS 를 사용하는 경우 (Hadoop 등) HDFS 가 필요합니다.

- 지원 EcoSystem

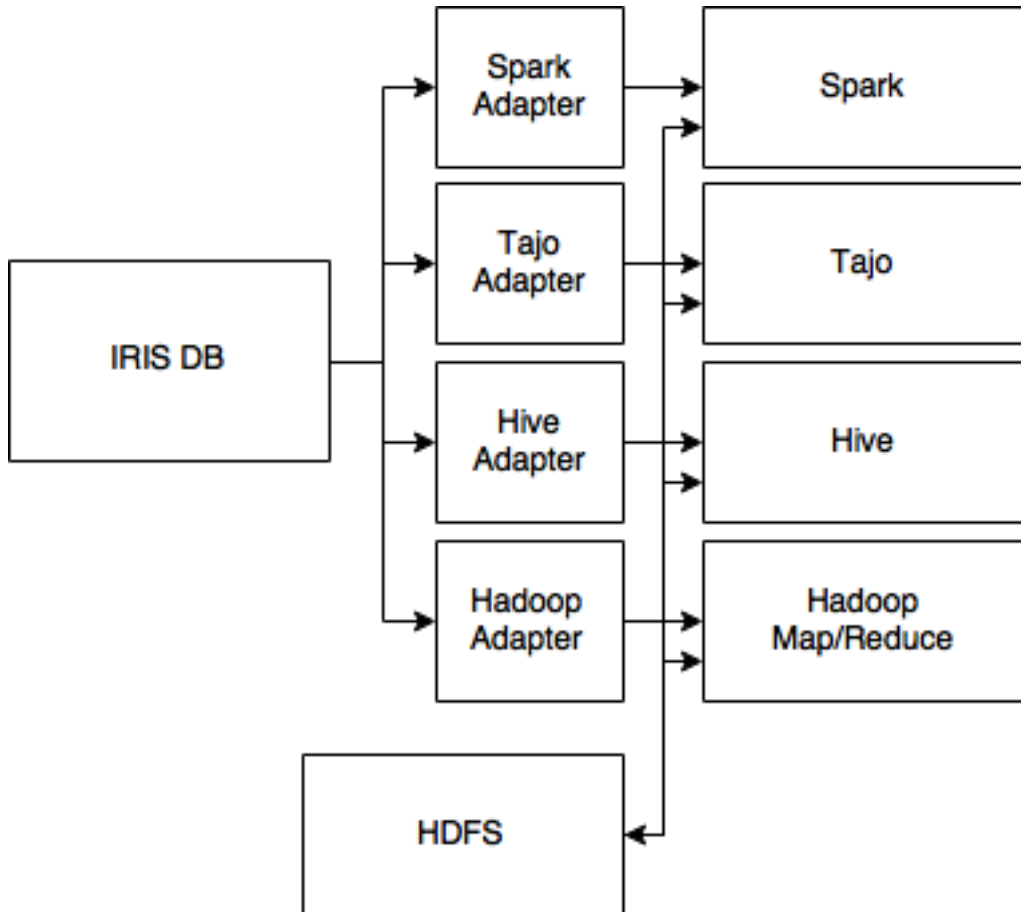


그림 1: Hadoop EcoSystem 구성도

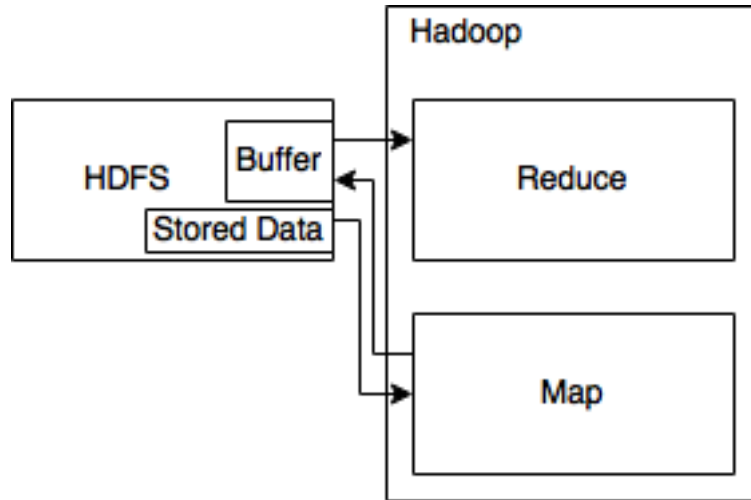


그림 2: 일반적인 HDFS 기반 Map/Reduce 구성도

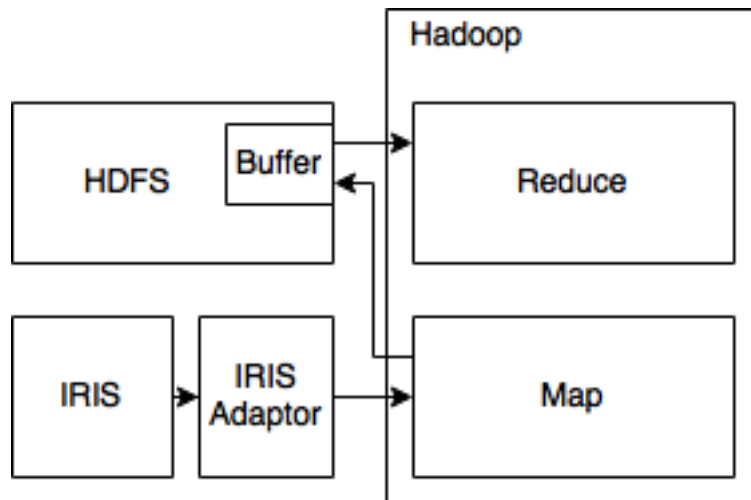


그림 3: IRIS 를 사용한 Map/Reduce 구성도

Spark

```
사용예시)
from pyspark import SparkContext
sc = SparkContext(appName="test")
from sparkiris import sparkiris
sc = sparkiris(sc)
sc.set_iris_info('192.168.0.170', 'test','test', 35210, 35110, \
    35000, 35678, 35800)
print sc.irisDB('select * from benchtest_100m;').count()
```

Tajo

```
사용예시)
default> SELECT ID, INT_KEY from BENCHTEST_100M WHERE key = 'k0'
    AND partition = '20141020092000' LIMIT 3;
Progress: 0%, response time: 0.432 sec
Progress: 0%, response time: 0.433 sec
Progress: 0%, response time: 0.835 sec
Progress: 0%, response time: 1.637 sec
Progress: 0%, response time: 2.639 sec
Progress: 100%, response time: 2.857 sec
id, int_key
-----
5, 50
6, 10
10, 50
(3 rows, 2.857 sec, 16 B selected)
default>
```

Hive

Hadoop Map/Reduce

2.4 시스템 정보 제공기능

SQL 명령어를 통해 시스템정보를 제공합니다. 옵션을 설정해서 기간별 누적정보를 제공할 수 있습니다. 자세한 내용 및 사용법은 아래의 CLI 목록에 있습니다.

```

iplus> .statistics system
Ret : +OK Success

  UPDATE_TIME  NODE_ID    NODE_IP    SYS_STATUS  ADM_STATUS  HOST_NAME
OS_NAME        OS_VERSION OS_TYPE    NET_NAME    NET_TYPE    NET_MAC
NET_IN_PACKET NET_OUT_PACKET NET_IN_BYTE NET_OUT_BYTE CPU_CLOCK
CPU_CORE       CPU_USAGE  CPU_L_AVG  CPU_IOWAIT  RAM_TOTAL
RAM_USAGE_FILE RAM_USAGE_PROCESS RAM_SWAP_TOTAL RAM_SWAP_USAGE
HDD_TOTAL      HDD_USAGE
=====
201506040845  0          192.168.131.13 VALID        ENABLE      ...
201506040845  1          192.168.131.131 VALID        ENABLE      ...
201506040845  2          192.168.131.132 VALID        ENABLE      ...
=====
4 row in set
0.0206 sec

iplus>

```

2.5 Session 모니터링 기능

현재 실행중인 쿼리의 상태정보를 출력합니다. 조건에 맞는 세션을 조회하거나, 임의의 세션을 강제종료할 수 있습니다. 기본실행시 현재 실행중인 최신 10건의 세션 정보를 출력합니다. 자세한 내용 및 사용법은 아래의 CLI 목록에 있습니다.

```

iplus> .session list
Ret : +OK Success

```

SID	QUERY_STRING	START_TIME	END_TIME	TYPE	NODE
PID	RESULT				
=====					
20150604140235_2_19577_32	select		20150604140235	20150604140235	END ...
20150604140235_2_19577_33	node		20150604140235	20150604140235	END ...
20150604140235_2_19577_34	select		20150604140235	20150604140235	END ...
20150604140235_2_19577_35	system		20150604140235	20150604140235	END ...
20150604140235_2_19577_36	select		20150604140235	20150604140236	END ...
20150604140236_2_19577_37	system		20150604140236	20150604140236	END ...
20150604140236_2_19577_38	select		20150604140236	20150604140236	END ...
20150604140236_2_19577_39	system		20150604140236	20150604140236	END ...
20150604140236_2_19577_40	select		20150604140236	20150604140236	END ...
20150604140236_2_19577_41	system		20150604140236	20150604140236	END ...
=====					
10 row in set					
0.2966 sec					

작업 종료 기능 제공

```

iplus> .session term 20150604140235_2_19577_34
Ret : +OK kill 20150604140235_2_19577_34 session.

6.2503 sec

iplus>

```

2.6 IRIS Client 제공

IRIS API 파일과 IPLUS (IRIS CLI Interface) 를 하나로 묶은 통합 패키지를 제공합니다. IPLUS 를 임의의 서버에 설치해서 원격지의 IRIS에 접속할 수 있습니다. IRIS 는 대부분의 작업을 CLI 명령어로 제공하므로 IRIS Client 만으로도 원격지의 IRIS 를 제어, 관리 할 수 있습니다. 자세한 내용 및 사용법은 아래의 CLI 목록에 있습니다.

2.7 IRIS 고도화

2.7.1 DTD

DTD는 기존 IRIS에서 데이터를 동기화, 삭제를 진행하기 위해서 사용하는 BTD, BSD 데몬은 대체하게 위해서 만들어진 데몬입니다. 기존의 BTD, BSD 에서 하는 모든 작업을 DTD에서 수행하며, 코드 최적화 및 범용성을 가지도록 하였습니다.

2.7.2 IRIS FileSystem

기존 IRIS의 슬레이브 노드에서는 각각의 데이터 위치를 Ram 파일 시스템의 링크를 참조하는 방식으로 진행되었습니다. 이는 불필요한 메모리 소모 및 관리 어려움을 야기하였습니다. 이를 해결하고자 IRIS FileSystem 을 도입하였습니다. IRIS FileSystem은 기존의 위치 정보를 따로 관리 함으로써 메모리 소모 및 관리의 편리성을 확보하였습니다.

2.7.3 PR / PL

기존 IRIS PM 에서 하는 일이 대부분 IRIS FileSystem 내부로 들어감에 따라 PM은 제거되고 데이터 expire 관리를 위해 PR / PL 이 개발되었습니다.

2.7.3.1 PL [Partition Locator] PL은 Partition Locator로 파티션 위치 정보를 관리하며, RAM -> SSD or DISK, SSD -> DISK 의 이동을 관리합니다.

2.7.3.2 PR [Partition Remover] PR은 Partition Remover로 파티션 및 기타 파일 삭제를 관리합니다. 삭제를 관리하는 항목은 아래와 같습니다.

- 디스크에서 삭제 대기중인 파티션
- 사용하지 않는 락 파일
- 임시 파일

2.8 CLI 명령어

IRIS 에서는 시스템 정보 조회, 또는 IRIS 에서 지원하는 기능을 사용할 수 있는 CLI 명령어를 제공합니다. API 또는 IPLUS 를 사용해서 쿼리와 동일하게 CLI 명령어를 실행, 결과를 리턴받습니다. IRIS 가 설치되지 않은 원격 클라이언트에서 사용 가능하며, 일부 기능은 계정권한에 영향을 받습니다.

2.8.1 IPLUS CMD 명령어

IRIS 에서는 IRIS 에 편하게 접속이 가능한 IPLUS CLI 를 제공합니다.

접속방법 : `iplus 계정명[@호스트[:포트번호]]`

계정명만 사용하는 경우, 해당 노드에 설치된 IRIS 에 접속합니다.

```
[iris@ ~]$ iplus test
Password:
Connecting to IRIS(test@192.168.131.13:5050).
Connected to IRIS.
Enter ".help" for instructions
iplus>
```

마스터노드를 호스트로 설정한 경우 포트번호 입력없이 접속가능합니다.

포트번호는 5050 입니다. (기본값이므로 입력안해도 무방합니다.)

```
$ iplus test@192.168.131.13
Password:
```

```
Connecting to IRIS(test@192.168.131.13:5050).
Connected to IRIS.
Enter ".help" for instructions
iplus>
```

특정 슬레이브노드를 호스트로 설정한 경우 포트번호는 5100 입니다.

```
$ iplus test@192.168.131.131:5100
Password:
Connecting to IRIS(test@192.168.131.131:5100).
Connected to IRIS.
Enter ".help" for instructions
iplus> .q
```

시퀀스파일 실행방법 : IPlus < {시퀀스파일경로}

STDIN 으로 시퀀스파일경로를 넣으면 해당 파일속의 명령어/쿼리를 순서대로 실행합니다.

```
[iris@localhost ~]$ cat seq_test
--#test/test#--
.recordSep \n
.fieldSep \t
SELECT * FROM LOCAL_TEST_TABLE;
[iris@localhost ~]$

[iris@localhost ~]$ iplus < seq_test
Connecting to IRIS(test@192.168.131.13:5050).
Connected to IRIS.
Enter ".help" for instructions
SQL syntax error.
After the error occured, session recreated.
SQL syntax error.
After the error occured, session recreated.
Ret : +OK Success

K          P          A
```



```

=====
k6          20150616000000 None
k7          20150616000000
k4          20150616000000 0
k5          20150616000000 0.1
k3          20150616000000 1.2
k2          20150616000000 1
=====

6 row in set
0.0982 sec

Goodbye.
[iris@localhost ~]$

```

2.8.2 IPLUS CLI 명령어

IPLUS CLI 에서 사용가능한 명령어입니다.

명령어	설명
.history	명령/쿼리 실행 기록을 출력합니다.
.sep {Field구분자} {Record구분자}	IRIS 에 저장되지 않고 서버 계정별로 로컬에 저장됩니다.
.field_sep {Field구분자}	현재 IPLUS 접속에서 사용할 구분자를 설정합니다.
.record_sep {Record구분자}	현재 IPLUS 접속에서 사용할 Field 구분자만 설정합니다.
.width {길이} {길이} ...	현재 IPLUS 접속에서 사용할 Record 구분자만 설정합니다. 각 컬럼별 길이를 순서대로 설정합니다. 설정되지 않은 컬럼은 13 으로 설정됩니다.
.load {테이블명} {키} {파티션}	로컬에 있는 파일을 IRIS 로 로딩합니다.
{컨트롤파일경로} {데이터파일경로}	파일경로는 절대경로를 사용하세요.
.spool {파일경로}	결과데이터를 파일로 출력합니다. 결과메세지는 출력하지 않습니다.
.spool	파일경로를 넣지 않으면 현재 spool 기능을 끄고 stdout 으로 출력합니다.
!{Shell명령어}	Shell 명령어를 실행합니다.
@{시퀀스파일}	시퀀스파일 속의 명령어/쿼리를 순서대로 실행합니다.

2.8.3 IRIS 정보조회/수정 관련 명령어

2.8.3.1 테이블 명령어

테이블목록을 조회하고, 테이블 설정값을 수정합니다. 테이블 인덱스도 조회/수정합니다.

명령어	설명
<code>.table list</code>	현재 접속한 계정 소유의 테이블 목록을 출력합니다. -a 옵션을 뒤에 붙이면 전체테이블이 조회됩니다. -s 옵션을 뒤에 붙이면 시스템테이블이 조회됩니다.
<code>.table info {테이블이름}</code>	인자로 입력한 테이블의 정보를 출력합니다.
<code>.table schema {테이블이름}</code>	인자로 입력한 테이블의 스키마를 출력합니다. IRIS 옵션은 출력하지 않습니다.
<code>.table index</code>	현재 유저가 사용가능한 테이블들의 인덱스 목록을 출력합니다.
<code>.table index -all</code>	전체 인덱스 목록을 출력합니다.
<code>.table index -add {인덱스명} {테이블명} (컬럼, ...)</code>	LOCAL 테이블의 인덱스를 추가합니다.
<code>.table index -del {인덱스명}</code>	해당 LOCAL 테이블의 인덱스를 제거합니다.
<code>.table expire {테이블명} -ram {램보관주기(분)}</code>	해당 LOCAL 테이블의 램디스크 보관주기를 변경합니다.
<code>.table expire {테이블명} -disk {디스크보관주기(분)}</code>	해당 LOCAL 테이블의 디스크 보관주기를 변경합니다.

2.8.3.2 세션 관리 명령어

세션목록을 조회하거나 특정 세션을 종료합니다. 세션 종료는 사용자가 직접 1개씩 SID 를 입력해야 합니다.

명령어	설명
<code>.session list</code>	세션 목록을 출력합니다. 옵션을 사용해서 여러 조건에 맞는 세션목록만 출력할 수 있습니다. 기본값 현재시간부터 1시간 전까지 모든 타입에 대해서 가장 최근에 끝난 세션부터 10개를 출력합니다.
<code>.session info {SID}</code>	입력한 SID 에 해당하는 세션의 정보를 출력합니다. 전체 쿼리 문자열을 확인할 수 있습니다.
<code>.session term {SID}</code>	입력한 SID 에 해당하는 세션이 살아있는경우, 해당 세션을 강제종료합니다. 이미 종료된 상태이면 오류메세지를 출력합니다.

2.8.3.3 휴지통기능 명령어

휴지통기능을 사용하는 테이블 목록을 조회/관리 합니다. 비우기, 복원하기 등의 휴지통기능은 QUERY 로 지원합니다.

DDL, DML 항목을 참조하세요.

명령어	설명
.recyclebin list	현재 휴지통기능을 사용하는 테이블 목록을 출력합니다.
.recyclebin add {테이블이름}	인자로 입력한 테이블을 휴지통기능을 사용하도록 설정합니다.
.recyclebin del {테이블이름}	인자로 입력한 테이블을 휴지통기능을 사용하지 않도록 설정합니다.

2.8.3.4 파일 관리 명령어 테이블의 보관기관과 관련된 옵션을 조회/수정합니다. 위의 테이블명령어에서 설명한 expire 옵션을 반영하는 기준을 설정하는 옵션입니다.

명령어	설명
.pm list	현재 각 테이블별 램옵션, 디스크옵션을 출력합니다. 둘 다 \$DEFAULT 인 경우는 출력되지 않습니다.
.pm {테이블이름} {램옵션} {디스크옵션}	이 명령어는 테이블의 램옵션, 디스크옵션을 설정하는 명령어입니다. 옵션은 테이블 생성시에 설정한 expire 값과 관련있는 옵션입니다. OFF : 데이터의 저장소변경/자동삭제를 하지 않습니다. LOCAL_TIME_BASE : 서버시간 기준으로 expire 시간을 계산합니다. PARTITION_BASE : 현재 저장된 데이터 기준으로 expire 시간을 계산합니다. \$DEFAULT : 기본옵션이며, .pm list 로 확인할 수 있습니다. 램 옵션 : 램디스크에 저장된 데이터가 하드디스크로 옮기는 시간의 옵션입니다. 디스크 옵션 : 디스크에 저장된 데이터가 자동삭제되는 시간의 옵션입니다.
.pm default {램옵션} {디스크옵션}	\$DEFAULT 의 값을 변경합니다. 옵션설명은 위와 같습니다.
.pm ram {테이블이름} {램옵션}	테이블의 램옵션만을 수정합니다. 옵션설명은 위와 같습니다.
.pm disk {테이블이름} {디스크옵션}	테이블의 디스크옵션만을 수정합니다. 옵션설명은 위와 같습니다.
.pm del {테이블이름}	테이블의 옵션을 삭제합니다. 삭제되면 램옵션, 디스크옵션 둘 다 \$DEFAULT 로 적용됩니다.

2.8.3.5 계정 명령어 계정 명령어는 password 를 제외하고 전부 root 계정으로 접속해야 사용가능합니다. root 계정의 비밀번호는 m6administrator 입니다.

명령어	설명
.user list	사용자목록을 출력합니다.
.user add {계정명:암호@호스트규칙}	계정을 추가합니다. 계정명, 암호, 접속가능한 호스트규칙을 함께 넣습니다. 모든 호스트에서 접속하게 하려면 * 을 사용합니다.
.user password {계정명:암호@호스트규칙}	계정의 암호를 수정합니다. 모든 계정은 자신의 암호를 수정할 수 있습니다. root 계정은 모든 계정의 암호를 수정할 수 있습니다.
.user del {계정명@호스트규칙}	계정을 삭제합니다. 소유 테이블은 삭제하지 않습니다.

2.8.4 System 정보조회 명령어

시스템정보를 조회합니다. 옵션을 사용해서 최신정보, 또는 기간별 정보를 출력합니다.

명령어	설명
.node list	현재 전체 노드정보를 출력합니다. IRIS 명령어의 ntop 과 동일합니다.
.statistics system	시스템정보 통계를 출력합니다. 다양한 옵션을 사용할 수 있으며, 기본으로 각 노드별 현재 시스템정보를 출력합니다.
.statistics table	테이블정보 통계를 출력합니다. 다양한 옵션을 사용할 수 있으며, 기본으로 전체 LOCAL 테이블의 최신 크기/파일갯수를 출력합니다.

2.8.5 Error 로그 조회 명령어

오류로그를 조회합니다.

명령어	설명
.error list	오류목록을 시간순서대로 출력합니다. 옵션을 사용해서 여러 조건에 맞는 오류목록만 출력할 수 있습니다. 기본값 : 현재시간부터 1시간 전까지 출력합니다.
.error count	오류 갯수를 출력합니다. 기본값은 list 명령과 동일합니다.

명령어	설명
-----	----

2.8.6 노드 명령어

IRIS 노드의 정보를 조회하거나 상태를 변경합니다.

명령어	설명
<code>.node list</code>	현재 전체 노드의 상태정보를 출력합니다.
<code>.node enable {노드ID}</code>	해당 노드를 활성화 합니다. 활성화하기전에 해당 노드 복구작업이 성공적으로 종료되었는지 확인하세요.
<code>.node disable {노드ID}</code>	해당 노드를 비활성화 합니다. 모든 노드가 비활성화되면 기존 접속은 유지되나 신규접속은 불가능합니다.

2.8.7 Daemon 프로세스 명령어

IRIS Daemon 프로세스의 상태를 조회하거나 시작/종료합니다.

명령어	설명
<code>.daemon list</code>	Daemon 프로세스 목록을 출력합니다. 옵션을 사용해서 해당 조건에 맞는 항목만 확인할 수 있습니다. 지원 옵션 : <code>-node {노드ID}</code> , <code>-abn</code> , <code>-name {프로세스명}</code> , <code>-cmd</code>
<code>.daemon act {노드ID} {MID}</code>	해당 노드의 MID 에 해당하는 Daemon 프로세스를 시작합니다.
<code>.daemon term {노드ID} {MID}</code>	해당 노드의 MID 에 해당하는 Daemon 프로세스를 종료합니다.

2.8.8 부가기능 명령어

기타 유용한 부가기능 명령어 목록입니다.

명령어	설명
<code>.whoami</code>	현재 접속중인 IRIS 계정이름을 출력합니다.