ZIM 9.10

Data Manipulation Commands

SMARTCONE
LEADING THE EDGE

What is Zim?

Zim is

a complete framework to develop and run professional and mission critical applications by tightly integrating a lean relational database, a powerful Fourth Generation Language, an integrated development tool, the integration with outside world and client user interfaces.

SMARTCONE
LEADING THE EDGE

Zim Basic Concepts

- Entity-Relationship model, an upper-set of the Relational Model
- Set processing
- A simple yet powerful object-based 4GL
- Event driven user interfaces
- A built-in, high performance, open, high capacity,multi-user database management system (DBMS)

SMARTCONE
LEADING THE EDGE

The Relational Model Basics

The Relational Model was presented by Codd and Date and is based on the customer point of view:

"How the user sees the data".

- Tables or Entity Sets -

Customers Table

Code	Name	Address	Company	City	Credit
0101	John Voight	1280 Riverside Dr	ACME Inc	New York	1000
0102	Mark Stuart	320 Colonnade Av	B & D Ltd	Boston	1200
0110	Phyllis Morris	25 Fifth Avenue	Mackormik	Ottawa	1000
0105	Mark Knut	111 Main Street	CocaCo Inc	Chicago	850

Table Characteristics

- Has a name (Customers);
- One or more columns or fields (Code, Name, etc;
- Zero or more rows (Records);
- Primary (Unique) key (Code);
- The position of a row in a table is not relevant.

Code	Name	Address	Company	City	Credit
0101	John Voight	1280 Riverside Dr	ACME Inc	New York	1000
0102	Mark Stuart	320 Colonnade Av	B & D Ltd	Boston	1200
0110	Phyllis Morris	25 Fifth Avenue	Mackormik	Ottawa	1000
0105	Mark Knut	111 Main Street	CocaCo Inc	Chicago	850

Customers

SMARTCONE
LEADING THE EDGE

Table Operations

Add	one or more records to a table
Change	one or more records of tables
Delete	one or more records from a table
List	records
Find	records
Compute	values based on data records
Report	from record tables

SMARTCONE
LEADING THE EDGE

Listing Tables

List all Employees

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
1	Joseph Travolta	100000.00	MKT	NYC	19981012	8333.33
2	Mark Hirudsa	62000.00	R&D	OTT	19981212	5166.67
3	Frank Copolla	38000.00	R&D	OTT	19971208	3166.67
4	Mary Stuart	41500.00	MKT	TOR	19990204	3458.33
5	Nicholas Nick	26800.00	R&D	TOR	19970304	2233.33
10	Carl Santana	58900.00	R&D	TOR	19981104	4908.33
7	Mark Chapman	102000.00	SAL	OTT	19980421	8500.00
11	Marshal Winduck	91800.00	MKT	NWJ	19980717	7650.00
12	Larry King	53800.00	R&D	NYC	19980916	4483.33
13	Alex Sander	29800.00	R&D	OTT	19970912	2483.33
14	Lucy MArkham	33300.00	MKT	OTT	19981109	2775.00
15	Karin Lalonde	18600.00	R&D	OTT	19961207	1550.00

List 3 Employees

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
1	Joseph Travolta	100000.00	MKT	NYC	19981012	8333.33
2	Mark Hirudsa	62000.00	R&D	OTT	19981212	5166.67
3	Frank Copolla	38000.00	R&D	OTT	19971208	3166.67

Listing Tables - Formats

```
Set Output FORMAT COMMADELIMITED
List 1 Employees
```

```
1,"Joseph Travolta"," 80,000.00","MKT","NYC","1998/10/12","  6,666.67"
```

```
Set Output FORMAT XMLSIMPLE
List 1 Employees
```

```
<row>
   <EmpNum>1</EmpNum>
   <Name>Joseph Travolta</Name>
   <Salary>80,000.00</Salary>
   <DeptCode>MKT</DeptCode>
   <CityCode>NYC</CityCode>
   <HireDate>1998/10/12</HireDate>
   <MonthlySalary>6,666.67</MonthlySalary>
</row>
```

SMARTCONE
LEADING THE EDGE

Applying Conditional Selections

Conditional Selections

List All Employees where Salary > 50000

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
1	Joseph Travolta	100,000.00	MKT	NYC	1998/10/12	8,333.33
2	Mark Hirudsa	62,000.00	R&D	OTT	1998/12/12	5,166.67
7	Mark Chapman	102,000.00	SAL	OTT	1998/04/21	8,500.00
11	Marshal Winduck	91,800.00	MKT	NWJ	1998/07/17	7,650.00
12	Larry King	53,800.00	R&D	NYC	1998/09/16	4,483.33

List All Employees where EmpNum = 11

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
11	Marshal Winduck	91800.00	MKT	NWJ	19980717	7650.00

List All Employees where Name = ‘M’?

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
2	Mark Hirudsa	62000.00	R&D	OTT	19981212	5166.67
4	Mary Stuart	41500.00	MKT	TOR	19990204	3458.33
7	Mark Chapman	102000.00	SAL	OTT	19980421	8500.00
11	Marshal Winduck	91800.00	MKT	NWJ	19980717	7650.00

The Format Clause

**List All Employees Format **

**Name Salary $Total(Salary) **

$Average(Salary)

Name	Salary		
Joseph Travolta	100,000.00	100000.00	100000.00
Hirudsa	62,000.00	162000.00	81000.00
opolla	38,000.00	200000.00	66666.67
art	41,500.00	241500.00	60375.00
as Nick	26,800.00	268300.00	53660.00
arl Santana		268300.00	53660.00
Mark Chapman	102,000.00	370300.00	61716.67
Marshal Winduck	91,800.00	462100.00	66014.29
Larry King	53,800.00	515900.00	64487.50
Alex Sander		515900.00	64487.50
Lucy MArkham	33,300.00	549200.00	61022.22
Karin Lalonde	18,600.00	567800.00	56780.00

Unknown Values or "NULL"

SMARTCONE
LEADING THE EDGE

Aggregate Functions

Function Syntax	Value Returned
$Total(<Expression>)	Sum of all non null values of <**Expression>**
$Average(<Expression>)	Sum of all non null values of <**Expression>**
$Count(<Expression>)	Number of records where **<Expression>** is not null
$Max(<Expression>)	Maximum value of **<Expression>**.
$Min(<Expression>)	Minimum value of **<Expression>**

Combining Selections, Sort and Format

Set nullvalue "I AM NULL"
**List All Employees **
**where CityCode = "OTT" **
**Sorted by DeptCode Salary Desc **
**format Empnum Name DeptCode Salary **
CityCode $Total(Salary)

ENum	Name	DeptCode	Salary	CityCode	
14	Lucy MArkham	MKT	33,300.00	OTT	33300.00
2	Mark Hirudsa	R&D	62,000.00	OTT	95300.00
3	Frank Copolla	R&D	38,000.00	OTT	133300.00
15	Karin Lalonde	R&D	18,600.00	OTT	151900.00
13	Alex Sander	R&D	I AM NULL	OTT	151900.00
7	Mark Chapman	SAL	102,000.00	OTT	253900.00

SMARTCONE
LEADING THE EDGE

LIST Command Syntax Overview

```
LIST [num] [setspec]                \
     [FORMAT «expression»]          \
     [-> setname(*)]
```

[num] - Number of records to list (Default is 1)

[setspec] - table names, conditions, sort, etc.

«expression» - field name, constant, variable or any valid expression

(*) - See the presentation Result Sets

Changing Records

```
Change 1 Employees where Name = 'Mary Stuart'   \
        let Name = 'Mary Stuart II'             \
        -> sChanged
```

List all sChanged

```
ENum Name                 Salary DeptCode CityCode HireDate Monthly Salary

   4 Mary Stuart II     41500.00 MKT      TOR      19990204        3458.33
```

Output $Membercount

1

The system variable $membercount provides the number of records changed by the last CHANGE operation.

SMARTCONE
LEADING THE EDGE

Changing Records

Changes to the records in the original table are reflected in the result set and vice-versa.

```
> list all Departments wh MgrCode > 0 -> sDept1

DeptCode Department Name            MgrCode            Budget Expenses

SE       Sales Engineers                  2           6000.00      88,000.00
R&D      Research and Development        10           4000.00     284,500.00
MKT      Marketing                       11           2000.00     107,000.00


> Change Departments where MgrCode = 11 \
>      Let Expenses = Expenses + 50000
> out $MemberCount
           1


> list all sDept1

DeptCode Department Name            MgrCode            Budget Expenses

SE       Sales Engineers                  2           6000.00      88,000.00
R&D      Research and Development        10           4000.00     284,500.00
MKT      Marketing                       11           2000.00     157,000.00
```

SMARTCONE
LEADING THE EDGE

CHANGE Command Syntax Overview

```
Change [num] [setspec]                    \
       [Let field=«expression» ...]  \
       [-> setname(*)]
```

[num]	- Number of records to change. Default is 1.
[setspec]	- Table names, conditions, sort…
«expression»	- Field name, constant, variable or any valid expression
(*)	- See the presentation Result Sets

For complete command syntax refer to Language Reference Manual.

SMARTCONE
LEADING THE EDGE

Adding Records to Tables

```
Add 1 Employees                                    \
        Let EmpNum   = 100                         \
            Name     = "Elton Joseph"              \
            Salary   = 130000                      \
            DeptCode = "SAL"                       \
            CityCode = "LND"                       \
            HireDate = $Date                       \
        -> sAdded
```

List sAdded

ENum	Name	Salary	DeptCode	CityCode	HireDate	Monthly Salary
100	Elton Joseph	130000.00	SAL	LDN	20000224	10833.33

SMARTCONE
LEADING THE EDGE

Adding Data From Terminal

ADD Employees FROM TERMINAL PROMPT

```
ENum      Name              Salary       DeptCode CityCode HireDate
:101  Mary  50000 SAL OTT  19991202
```

When entering text with blanks, change the default field delimiter.

SET DELIMITER “;”

AFT Employees

```
ENum      Name              Salary       DeptCode CityCode HireDate
:102;David Letterkid;150000;MKT;NYC;20000315
```

SMARTCONE
LEADING THE EDGE

Adding Data From a Text File

Add Employees from MyData

MyData is a Zim Document name pointing to the file "mydata.txt".

Containing comma delimited data

```
100,"Mrk King"," 34,000.00","R&D","NYC","1998/09/16","  2,833.33"
101,"Alex Sander","","R&D","OTT","1997/09/12",""
102,"Lucy MArkham"," 33,300.00","MKT","OTT","1998/11/09","  2,775.00"
103,"Karin Lalonde"," 18,600.00","R&D","OTT","1996/12/07","  1,550.00"
```

Or fixed record length data

```
100 Larry King       34,000.00 R&D   NYC  1998/09/16   2,833.33
101 Alex Sander                R&D   OTT  1997/09/12
102 Lucy MArkham     33,300.00 MKT   OTT  1998/11/09   2,775.00
103 Karin Lalonde    18,600.00 R&D   OTT  1996/12/07   1,550.00
```

SMARTCONE
LEADING THE EDGE

ADD Command Syntax Overview

```
ADD  [num] [setspec]                                  \
           [Let field=«expression»...]                \
           [-> setname(*)]
```

or

```
ADD  [num] [setspec]                                  \
           From <source>                              \
           [Format <fields>]                          \
           [-> setname(*)]
```

(*) See the presentation Result Sets

SMARTCONE
LEADING THE EDGE

Deleting Records from Tables

Delete ALL Employees where EmpNum > 100
Output $membercount
List ALL Employees

1	Joseph Travolta	168000.00	MKT	NYC	19981012	14000.00
2	Mark Hirudsa	62001.00	R&D	OTT	19981212	5166.75
3	Frank Copolla	38001.00	R&D	OTT	19971208	3166.75
4	Mary Stuart II	41501.00	MKT	TOR	19990204	3458.42
5	Nicholas Nick	26802.01	R&D	TOR	19970304	2233.50
10	Carl Santana		R&D	TOR	19981104	
7	Mark Chapman	102001.00	SAL	OTT	19980421	8500.08
11	Marshal Winduck	91802.01	MKT	NWJ	19980717	7650.17
12	Larry King	53802.01	R&D	NYC	19980916	4483.50
13	Alex Sander		R&D	OTT	19970912	
14	Lucy MArkham	33301.00	MKT	OTT	19981109	2775.08
15	Karin Lalonde	18602.01	R&D	OTT	19961207	1550.17
100	Elton Joseph	130000.00	SAL	LDN	20000224	10833.33

Deleted Records and Result Sets

List ALL Departments where Name = "A"? -> sA

```
DeptCode Department Name              MgrCode            Budget          Expenses

ACC      Accounting                         0           1000.00          88000.00
A&D      Accessibility and Control ????????         ?????????          35000.00
```

Bottom sA

Delete ALL sA

Output $Membercount

1

Top sA

List All sA

```
DeptCode Department Name              MgrCode            Budget          Expenses

ACC      Accounting                         0           1000.00          88000.00
*** Warning *** A member of this set has been deleted
                 since it was first found.
???????? ?????????                  ????????         ?????????          ?????????
```

SMARTCONE
LEADING THE EDGE

DELETE Command Syntax Overview

Delete [num] [setspec] [-> setname(*)]

[num] is 1 by default.

If the target of DELETE is a Result Set, deletion always starts at the current member of the set.
If no selection condition (Where) is present, deletion starts at the first member of the object.

(*) See the presentation Result Sets

SMARTCONE
LEADING THE EDGE

Finding or Listing Records

- **The LIST command lists the selected records and *optionally* creates a result set.**
- **The FIND command finds the selected records and *always* creates a result set, even if a set name was not specified.**

SMARTCONE
LEADING THE EDGE

Finding Records

Find All Employees -> $CurrentSet

"ALL" is the default number of records to be selected

$CurrentSet is the Result Set default name

Find Employees

The above commands are equivalent

SMARTCONE
LEADING THE EDGE

Finding Records

Find commands ALWAYS generate a $CURRENTSET

```
> Find all Employees where Salary > 50000 and NOT DeptCode IN ("MKT","R&D")
1 selected.
> List all CurrentSet

   ENum Name            Salary     DeptCode CityCode HireDate    Monthly Salary

      7 Mark Chapman    102,000.00 SAL      OTT      1998/04/21   8,500.00
```

If no set name is specified in a command, $CURRENTSET is assumed.

```
> List all

   ENum Name            Salary     DeptCode CityCode HireDate    Monthly Salary

      7 Mark Chapman    102,000.00 SAL      OTT      1998/04/21   8,500.00
```

SMARTCONE
LEADING THE EDGE

Finding Records

The $CurrentSet is always the result set of the last FIND command.

But, by using a set name, this set stays valid until the end of the ZimQTC session.

```
> Find all Employees where Salary > 50000 and NOT DeptCode IN ("MKT","R&D") -> sEmp
1 selected.
> List all Currentset

    ENum Name              Salary     DeptCode CityCode HireDate   Monthly Salary

       7 Mark Chapman      102,000.00 SAL      OTT      1998/04/21   8,500.00
>
> Find all Departments where Name = 'A'? -> sDep
1 selected.
> List all Currentset

DeptCode Department Name              MgrCode             Budget Expenses

ACC      Accounting                         0            1000.00      88,000.00
```

SMARTCONE
LEADING THE EDGE

System Variables
$SetCount and $MemberCount

```
> Find Departments where MgrCode > 0 -> sDept1
3 selected.

> Out $SetCount
            3
> Out $MemberCount
            3


> List 2

DeptCode Department Name              MgrCode              Budget Expenses

SE       Sales Engineers                    2            6000.00      88,000.00
R&D      Research and Development          10            4000.00     284,500.00
> Out $MemberCount
            2
> out $SetCount
            3
```

SMARTCONE
LEADING THE EDGE

FIND Command Syntax Overview

```
Find [num] [setspec]          \
     [Evaluate <expr>]        \
     [-> setname(*)]
```

- **[num]** is ALL by default.
- The CurrentSet is always generated even when **setname** is not explicitly specified.
- If **[setspec]** is not specified, CurrentSet is assumed.
- The Evaluate clause is used to assign values to variables using the records processed
- (*) See the presentation Result Sets

SMARTCONE
LEADING THE EDGE

ZIM 9.10

Data Manipulation Commands

SMARTCONE
LEADING THE EDGE