

## **Program Summary**

Program is a database for a cooperative with about 50,000 members. Current application is using "dbf" files and written in Clipper and Delphi 5 using Apollo drivers. Source code for prototype application written in Delphi 5 will be provided.

The current project is for the membership part of the program. Additional modules will need to be written later.

## **Program Summary**

Tools :

1. Delphi 2007
2. FibPlus 6.85
3. Firebird 2
4. Database Grid (recommendations are welcome)
5. Fastreport

## **Requirements**

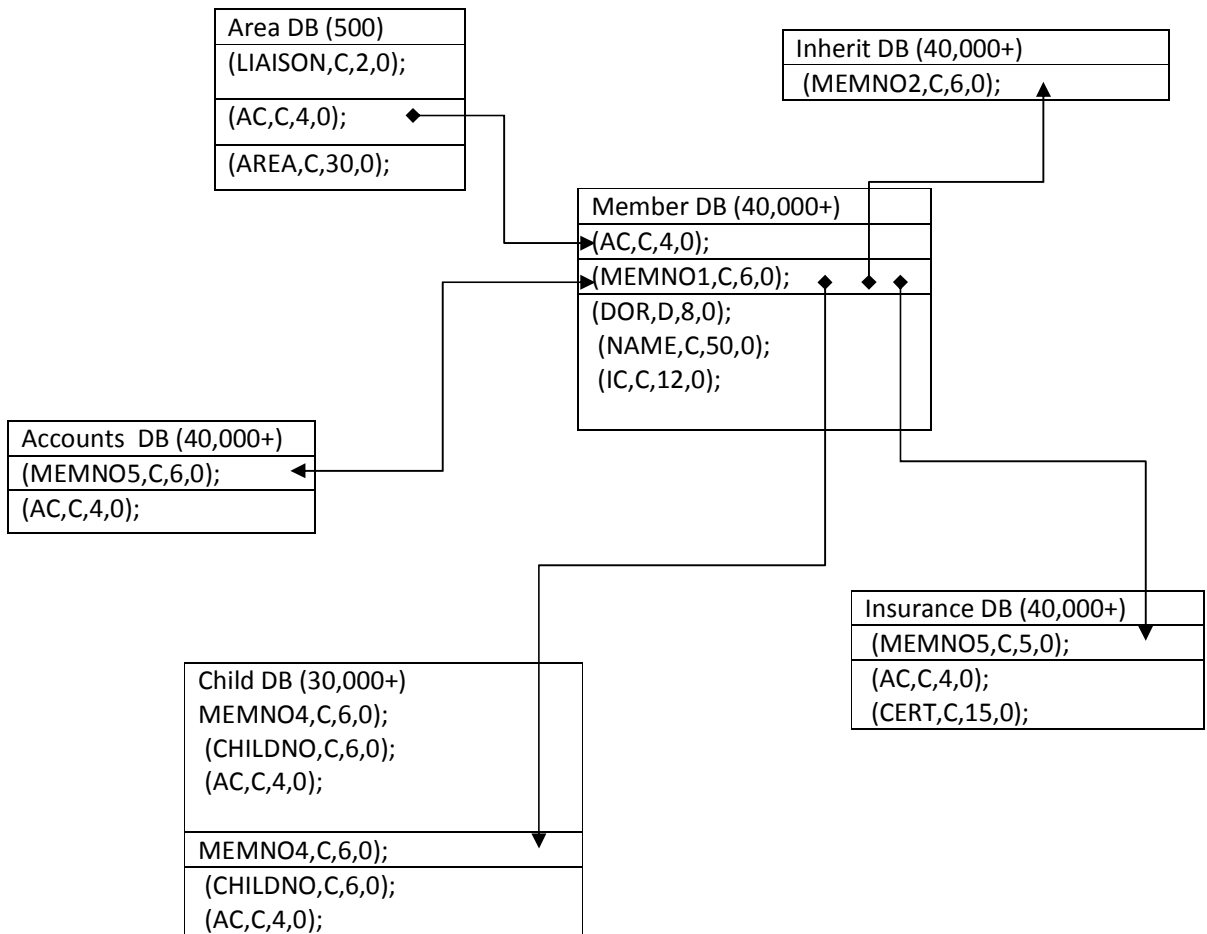
1. Data converted to Firebird. Provide code used to do conversion
2. Database to reflect file dependencies.
3. Data Entry screens as provided.
4. Some existing fields will be removed and other new fields will be added.
5. Database validation for duplicate records, one to many, one to one relationships.
6. Complete info on server installation for Windows 2003 server.
7. Ini files for directory / server information.
8. Require password security system. Supervisor password protection for editing certain functions, databases and editing fields.
9. Basic Reports.

## **Other considerations**

The written program will be used as a base for further modifications and additions. These modifications will be done in-house. Some help and support on using Fibplus / Firebird is required. At least another 5 modules are required for completion of the full project. These modules will be offered separately and will commence once this module is completed.

## **Copyright**

Full source code and copyright of all programs and databases will be transferred.



Liaison DB (45)  
Liaison Information  
Indexes :  
CreateField(LIAISON,C,2,0);

Function  
Areas are divided into liaisons.  
The sole purpose of this Db is to divide areas by liaisons

Area DB (500)  
Area Information  
Indexes :  
CreateField(AC,C,4,0);  
CreateField(AREA,C,30,0);  
CreateField(LIAISON,C,2,0);

Function  
All databases have an area field. All data can be filtered by area.

Member DB (40,000+)  
Membership Information  
Indexes :  
CreateField(MEMNO1,C,6,0);  
CreateField(AC,C,4,0);  
CreateField(DOR,D,8,0);  
CreateField(NAME,C,50,0);  
CreateField(IC,C,12,0);

Function  
Basic members information including address, date of birth etc.

Accounts DB (40,000+)  
Accounts Information  
Indexes :  
CreateField(MEMNO5,C,6,0);  
CreateField(AC,C,4,0);

Function  
Stores all financial details of member

Inherit DB (40,000+)  
Inheritance Information  
Indexes :  
CreateField(MEMNO2,C,6,0);

Function  
Details of Nominees

Child DB (30,000+)  
Children Information  
Indexes :  
CreateField(MEMNO4,C,6,0);  
CreateField(CHILDNO,C,6,0);  
CreateField(AC,C,4,0);

Function  
Members children details. This includes some accounting information

Share DB (20,000+)  
Share Information  
Indexes :  
CreateField(MEMNO10,C,6,0);  
CreateField(AREA,C,4,0);  
CreateField(CERT\_NO,C,7,0);

Function  
Details of shares purchased by member.

Insurance DB (40,000+)  
Insurance Information  
Indexes :  
CreateField(MEMNO5,C,5,0);  
CreateField(AC,C,4,0);  
ApolloTable1.CreateField(CERT,C,15,0);

Function  
Details of Insurance scheme. Includes accounting information

The primary indexes are Memno and AC. Most data is retrieved and calculated using these indexes.

These databases are just the part of the full system that contain summary data. There are other databases that do monthly calculations.

The current application is written in Apollo using Foxpro DBF/CDX databases. Program written in Delphi 5 is available.

Program needs to be rewritten in Firebird using Fibplus in Delphi 2007.

KKP Membership System

Utilities Index Databases

Navigation icons: Home, Back, Forward, Stop, Refresh, Print, Filter, etc. MEMNO1 statuslabel Close

00103 0002 JINJANG. ANNAMALAI A/L MUNIANDY Label1 Button2

Membership Information Browse Members Summary Area Reports Accounts

Membership No.	00103	ANNAMALAI A/L MUNIANDY					
Area	0002	JINJANG.	Status	Share Issued	0		
Age	50	Date of Birth	12/12/1954	Sex	Male	Race	Indian
Register Date	23/07/1978	IC No	541123105209	Old IC	1233394	Expiry Date	

Address	11 JALAN 135/42, TAMAN SEJAHTERA,SEGAMBUT, KUALA LUMPUR.	State	Wilayah Persekutuan
Place of Birth		Postcode	51200
Employers Name	K.K.P.]	Citizenship	Malaysian
Employers Address	ND 139-1 JALAN 9/42, TMN SEJAHTERA OFF, JLN KUCHING K.L.	Type of Job	P
		Monthly Salary	0.00
		Monthly Income	0.00
		Occupation	MANAGER FINANCE/ACI
		Proposer No.	
		Seconder No.	

MEMN01 statuslabel Close

00103 0002 JINJANG. ANNAMALAI A/L MUNIANDY Label1 Button2

Membership Information | Browse Members | Summary | Area | Reports | Accounts

Membership No. 00103 ANNAMALAI A/L MUNIANDY

Area 0002 JINJANG. Status Share Issued 0

Age 50 Date of Birth 12/12/1954 Sex Male Race Indian

Register Date 23/07/1978 IC No 541123105209 Old IC 1233394 Expiry Date

Nominee Name	IC No	Age	Relationship	%	Cert No.	Date	Units	Total	Share
THILLAINAYAKI A/P MUNIAPPILLAI	7360708	34	W	100	0014143	17/04/2002	10	10	1000.00

Child No	Child Name	Date of Birth	Total	Ins	Saving	Interest	Edu
002282	ARAVINTHAN A/L ANNAMA	08/07/1987	0.00	0.00	0.00	0.00	0
004484	AGILAN A/L ANNAMALAI	12/11/1989	-2529.22	0.00	2579.55	23.83	0
009155	ARIVALAGI A/P ANNAMALA	07/02/1993	326.62	0.00	2393.99	21.98	0

```

// Create Table
ApolloTable1.CreateNew('I:\knp2\MEMBER.DBF',2,35);
ApolloTable1.CreateField(MEMNO1,C,6,0);
ApolloTable1.CreateField(AC,C,4,0);
ApolloTable1.CreateField(DOR,D,8,0);
ApolloTable1.CreateField(NAME,C,50,0);
ApolloTable1.CreateField(IC,C,12,0);
ApolloTable1.CreateField(ICOLD,C,8,0);
ApolloTable1.CreateField(AGE,C,2,0);
ApolloTable1.CreateField(SEX,C,1,0);
ApolloTable1.CreateField(ADD1,C,30,0);
ApolloTable1.CreateField(ADD2,C,30,0);
ApolloTable1.CreateField(ADD3,C,30,0);
ApolloTable1.CreateField(STATE,C,2,0);
ApolloTable1.CreateField(PC,C,5,0);
ApolloTable1.CreateField(CITIZEN,C,1,0);
ApolloTable1.CreateField(BIRTH_P,C,15,0);
ApolloTable1.CreateField(BIRTH_D,D,8,0);
ApolloTable1.CreateField(E_NAME,C,20,0);
ApolloTable1.CreateField(E_ADD1,C,20,0);
ApolloTable1.CreateField(E_ADD2,C,20,0);
ApolloTable1.CreateField(E_ADD3,C,20,0);
ApolloTable1.CreateField(OCUP,C,20,0);
ApolloTable1.CreateField(TOJ,C,1,0);
ApolloTable1.CreateField(M_SAL,N,8,2);
ApolloTable1.CreateField(M_INC,N,8,2);
ApolloTable1.CreateField(P_MEMNO,C,6,0);
ApolloTable1.CreateField(S_MEMNO,C,6,0);
ApolloTable1.CreateField(RACE,C,1,0);
ApolloTable1.CreateField(STATUS,C,1,0);
ApolloTable1.CreateField(DOE,D,8,0);
ApolloTable1.CreateField(SHARE_TOT,N,3,0);
ApolloTable1.CreateField(OLDAC,C,4,0);
ApolloTable1.CreateField(MARK,C,1,0);
ApolloTable1.CreateField(TEMP,C,10,0);
ApolloTable1.CreateExec;

// Create Index Tags
// Table must be open to create index tags
ApolloTable1.DatabaseName := 'I:\knp2\';
ApolloTable1.TableName := 'MEMBER.DBF';
ApolloTable1.Open;
ApolloTable1.Close;
ApolloTable1.Exclusive := True;
DeleteFile('I:\knp2\MEMBER.cdx');
ApolloTable1.Open;
ApolloTable1.IndexTag("",MEMNO1,MEMNO1,0,False,);
ApolloTable1.IndexTag("",AC1A,AC+MEMNO1,0,False,);

```

```
ApolloTable1.IndexTag(",IC1,IC,0,False,);  
ApolloTable1.IndexTag(",NAME1,LEFT(NAME,5),0,False,);  
ApolloTable1.IndexTag(",ICOLD,ICOLD,0,False,);  
ApolloTable1.IndexTag(",STATUS,STATUS,0,False,);
```

**// Create Table**

```
ApolloTable1.CreateNew('I:\kkp2\KPSHARE.DBF',2,12);
```

```
ApolloTable1.CreateField(MEMNO10,C,6,0);  
ApolloTable1.CreateField(AREA,C,4,0);  
ApolloTable1.CreateField(CERT_NO,C,7,0);  
ApolloTable1.CreateField(ISSUE_DATE,D,8,0);  
ApolloTable1.CreateField(UNITS,N,3,0);  
ApolloTable1.CreateField(SHARE_TD,N,3,0);  
ApolloTable1.CreateField(SHARE_BAL,N,8,2);  
ApolloTable1.CreateField(PRINTED,L,1,0);  
ApolloTable1.CreateField(CANCELLED,L,1,0);  
ApolloTable1.CreateField(CANCELDATE,D,8,0);  
ApolloTable1.CreateExec;
```

**// Create Index Tags**

```
// Table must be open to create index tags  
ApolloTable1.DatabaseName := 'I:\kkp2\';  
ApolloTable1.TableName := 'KPSHARE.DBF';  
ApolloTable1.Open;  
ApolloTable1.Close;  
ApolloTable1.Exclusive := True;  
DeleteFile('I:\kkp2\KPSHARE.cdx');  
ApolloTable1.Open;  
ApolloTable1.IndexTag(",CERT_NO,CERT_NO,0,False,);  
ApolloTable1.IndexTag(",MEMNO10,MEMNO10,0,False,);  
ApolloTable1.IndexTag(",AREA,AREA,0,False,);
```

**// Create Table**

```
ApolloTable1.CreateNew('I:\kkp2\KPCHILD.DBF',2,21);
```

```
ApolloTable1.CreateField(MEMNO4,C,6,0);  
ApolloTable1.CreateField(CHILDNO,C,6,0);  
ApolloTable1.CreateField(NAME,C,30,0);  
ApolloTable1.CreateField(ICBC,C,8,0);  
ApolloTable1.CreateField(DOB,D,8,0);  
ApolloTable1.CreateField(EDU,N,3,0);  
ApolloTable1.CreateField(SCHOOL,C,30,0);  
ApolloTable1.CreateField(STD,C,2,0);  
ApolloTable1.CreateField(ACC,N,9,2);  
ApolloTable1.CreateField(AC,C,4,0);  
ApolloTable1.CreateField(INS,N,6,2);  
ApolloTable1.CreateField(INT1,N,7,2);  
ApolloTable1.CreateField(SAVBAL,N,8,2);  
ApolloTable1.CreateField(INTBAL,N,7,2);
```

```

ApolloTable1.CreateField(ZACC,N,9,2);
ApolloTable1.CreateField(ZINT1,N,7,2);
ApolloTable1.CreateField(IJUN,N,6,2);
ApolloTable1.CreateField(IDECA,N,6,2);
ApolloTable1.CreateField(STATUS,C,1,0);
ApolloTable1.CreateExec;

// Create Index Tags
// Table must be open to create index tags
ApolloTable1.DatabaseName := 'I:\kcp2\';
ApolloTable1.TableName := 'KPCHILD.DBF';
ApolloTable1.Open;
ApolloTable1.Close;
ApolloTable1.Exclusive := True;
DeleteFile('I:\kcp2\KPCHILD.cdx');
ApolloTable1.Open;
ApolloTable1.IndexTag(",MEMNO4,MEMNO4,0,False,);
ApolloTable1.IndexTag(",CHILDNO,CHILDNO,0,False,);
ApolloTable1.IndexTag(",AC4,AC+CHILDNO,0,False,);

// Create Table
ApolloTable1.CreateNew('I:\kcp2\KLIAISON.DBF',2,8);
ApolloTable1.CreateField(AC,C,4,0);
ApolloTable1.CreateField(AREA,C,30,0);
ApolloTable1.CreateField(ACCMON,C,2,0);
ApolloTable1.CreateField(ACCYEAR,C,4,0);
ApolloTable1.CreateField(LIAISON,C,2,0);
ApolloTable1.CreateField(LNAME,C,20,0);
ApolloTable1.CreateExec;

// Create Index Tags
// Table must be open to create index tags
ApolloTable1.DatabaseName := 'I:\kcp2\';
ApolloTable1.TableName := 'KLIAISON.DBF';
ApolloTable1.Open;
ApolloTable1.Close;
ApolloTable1.Exclusive := True;
DeleteFile('I:\kcp2\KLIAISON.cdx');
ApolloTable1.Open;
// No Index Defined

// Create Table
ApolloTable1.CreateNew('I:\kcp2\INHERIT.DBF',2,8);
ApolloTable1.CreateField(MEMNO2,C,6,0);
ApolloTable1.CreateField(NAME,C,30,0);
ApolloTable1.CreateField(ICNO,C,12,0);
ApolloTable1.CreateField(AGE,C,2,0);

```

```
ApolloTable1.CreateField(RELA,C,1,0);  
ApolloTable1.CreateField(PERCENT,N,3,0);  
ApolloTable1.CreateExec;
```

```
// Create Index Tags  
// Table must be open to create index tags  
ApolloTable1.DatabaseName := 'I:\kcp2\';  
ApolloTable1.TableName := 'INHERIT.DBF';  
ApolloTable1.Open;  
ApolloTable1.Close;  
ApolloTable1.Exclusive := True;  
DeleteFile('I:\kcp2\INHERIT.cdx');  
ApolloTable1.Open;  
// No Index Defined
```

```
// Create Table  
ApolloTable1.CreateNew('I:\knp2\AREA.DBF',2,8);  
ApolloTable1.CreateField(AC,C,4,0);  
ApolloTable1.CreateField(AREA,C,30,0);  
ApolloTable1.CreateField(ACCMON,C,2,0);  
ApolloTable1.CreateField(ACCYEAR,C,4,0);  
ApolloTable1.CreateField(LIAISON,C,2,0);  
ApolloTable1.CreateField(LNAME,C,20,0);  
ApolloTable1.CreateExec;
```

```
// Create Index Tags  
// Table must be open to create index tags  
ApolloTable1.DatabaseName := 'I:\knp2\';  
ApolloTable1.TableName := 'AREA.DBF';  
ApolloTable1.Open;  
ApolloTable1.Close;  
ApolloTable1.Exclusive := True;  
DeleteFile('I:\knp2\AREA.cdx');  
ApolloTable1.Open;  
ApolloTable1.IndexTag(",AC,AC,0,False,);  
ApolloTable1.IndexTag(",AREA,AREA,0,False,);  
ApolloTable1.IndexTag(",LIAISON,LIAISON,0,False,);
```

```
// Create Table  
ApolloTable1.CreateNew('I:\knp2\KPACC55.DBF',2,37);  
ApolloTable1.CreateField(DATE,D,8,0);  
ApolloTable1.CreateField(MEMNOS,C,6,0);  
ApolloTable1.CreateField(AC,C,4,0);  
ApolloTable1.CreateField(ENTFEE,N,5,2);  
ApolloTable1.CreateField(SHARE,N,8,2);  
ApolloTable1.CreateField(SAVING,N,8,2);  
ApolloTable1.CreateField(DIVIDEND,N,7,2);  
ApolloTable1.CreateField(SDEPO,N,8,2);  
ApolloTable1.CreateField(SDINT,N,7,2);  
ApolloTable1.CreateField(FSFEE,N,6,2);  
ApolloTable1.CreateField(INS,N,7,2);  
ApolloTable1.CreateField(CHILDS,N,8,2);  
ApolloTable1.CreateField(WELF,N,7,2);  
ApolloTable1.CreateField(EDU,N,7,2);  
ApolloTable1.CreateField(SLBAL,N,9,2);  
ApolloTable1.CreateField(SLPAID,N,9,2);  
ApolloTable1.CreateField(SLIBAL,N,8,2);  
ApolloTable1.CreateField(SLIPAID,N,8,2);  
ApolloTable1.CreateField(ULBAL,N,8,2);  
ApolloTable1.CreateField(ULPAID,N,8,2);  
ApolloTable1.CreateField(ULIBAL,N,7,2);  
ApolloTable1.CreateField(ULIPAID,N,7,2);
```

```

ApolloTable1.CreateField(SHJUNE,N,8,2);
ApolloTable1.CreateField(SDIJUNE,N,6,2);
ApolloTable1.CreateField(SDIDEC,N,6,2);
ApolloTable1.CreateField(VFUND,N,5,0);
ApolloTable1.CreateField(RFUND,N,5,0);
ApolloTable1.CreateField(SAVJUNE,N,8,2);
ApolloTable1.CreateField(DLQ_CONS,N,3,0);
ApolloTable1.CreateField(DLQ_TOT,N,3,0);
ApolloTable1.CreateField(INS_UNITS,N,1,0);
ApolloTable1.CreateField(INS_DUE,N,6,2);
ApolloTable1.CreateField(ELOAN,N,8,2);
ApolloTable1.CreateField(BUSINESS,C,1,0);
ApolloTable1.CreateField(BDATE,D,8,0);
ApolloTable1.CreateExec;

// Create Index Tags
// Table must be open to create index tags
ApolloTable1.DatabaseName := 'I:\kjp2\';
ApolloTable1.TableName := 'KPACC55.DBF';
ApolloTable1.Open;
ApolloTable1.Close;
ApolloTable1.Exclusive := True;
DeleteFile('I:\kjp2\KPACC55.cdx');
ApolloTable1.Open;
ApolloTable1.IndexTag('',MEMNO55,MEMNO5,0,False,);
ApolloTable1.IndexTag('',AC55,AC+MEMNO5,0,False1,);

// Create Table
ApolloTable1.CreateNew('I:\kjp2\KPINS.DBF',2,31);
ApolloTable1.CreateField(MEMNO5,C,5,0);
ApolloTable1.CreateField(AC,C,4,0);
ApolloTable1.CreateField(STATUS,C,1,0);
ApolloTable1.CreateField(RESIGN_D,D,8,0);
ApolloTable1.CreateField(INS_UNITS,N,1,0);
ApolloTable1.CreateField(INS_DUE,N,7,2);
ApolloTable1.CreateField(PREV_BAL,N,7,2);
ApolloTable1.CreateField(INS_PAID,N,6,2);
ApolloTable1.CreateField(STARTDATE,D,8,0);
ApolloTable1.CreateField(CERT,C,15,0);
ApolloTable1.CreateField(CHANGEUNIT,N,1,0);
ApolloTable1.CreateField(CHANGEDATE,D,8,0);
ApolloTable1.CreateField(CLOSEDATE,D,8,0);
ApolloTable1.CreateField(FINAL_BAL,N,7,2);
ApolloTable1.CreateField(TOTAL_PAID,N,6,2);
ApolloTable1.CreateField(UNITS_DUE,N,6,2);
ApolloTable1.CreateField(PREV_OPEN,N,7,2);
ApolloTable1.CreateField(OPEN_BAL,N,7,2);
ApolloTable1.CreateField(DUE1,N,6,2);
ApolloTable1.CreateField(DUE2,N,6,2);

```

```
ApolloTable1.CreateField(YTD_DUE,N,6,2);
ApolloTable1.CreateField(YTD_PAID,N,6,2);
ApolloTable1.CreateField(BAL_DUE,N,6,2);
ApolloTable1.CreateField(DEDUCTION,N,6,2);
ApolloTable1.CreateField(BIRTH_D,D,8,0);
ApolloTable1.CreateField(AGE,N,2,0);
ApolloTable1.CreateField(PREV_UNIT,N,1,0);
ApolloTable1.CreateField(REMARK,C,20,0);
ApolloTable1.CreateField(REMARK2,C,30,0);
ApolloTable1.CreateExec;

// Create Index Tags
// Table must be open to create index tags
ApolloTable1.DatabaseName := 'I:\kcp2\';
ApolloTable1.TableName := 'KPINS.DBF';
ApolloTable1.Open;
ApolloTable1.Close;
ApolloTable1.Exclusive := True;
DeleteFile('I:\kcp2\KPINS.cdx');
ApolloTable1.Open;
ApolloTable1.IndexTag('MEMNO5,MEMNO5,0,False,);
ApolloTable1.IndexTag('CERT,CERT,0,False,);
ApolloTable1.IndexTag('AC,AC,0,False,);
```