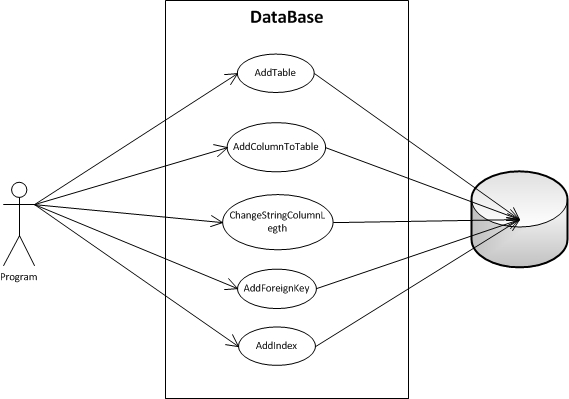
**DATABASE SCHEMA MANAGEMENT PROBLEM**

Database schema management is one of the most important problem to analize and manage before starting to think about any our software projects.

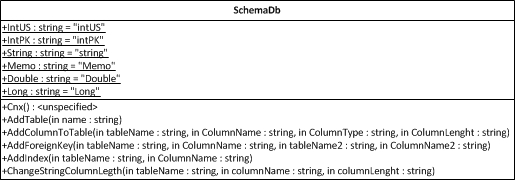
Target: write application indipendently from Database.

The idea is including in the database a simple interface to manage Table, Column, Foreign Key and so other stuff about database schema. In this way we will be able to modify schema directly and automatically from application and the application will be indipendently from database.

**USE CASE**

****

**DESIGN**



Class SchemaDb provide a simple interface to comunicate with database. Each metoth calls stored procedure in the database.

**MYSQL STORED PROCEDURE**

-- --------------------------------------------------------

-- Host: 127.0.0.1

-- Server version: 5.5.21 - MySQL Community Server (GPL)

-- Server OS: Win32

-- HeidiSQL version: 7.0.0.4053

-- Date/time: 2012-03-03 23:42:43

-- --------------------------------------------------------

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET NAMES utf8 \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=0 \*/;

-- Dumping structure for procedure db.AddColumnToTable

DELIMITER //

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddColumnToTable`(IN `tableName` TEXT, IN `columnName` TEXT, IN `typeName` TEXT, IN `typeDimension` INT)

BEGIN

SET @typeMysql = 'INT';

IF typeName = 'intPK' THEN

SET @typeMysql = CONCAT('INT(10) UNSIGNED NOT NULL AUTO\_INCREMENT, ADD PRIMARY KEY (',columnName,') ');

END IF;

IF typeName = 'intUS' THEN

SET @typeMysql = CONCAT('INT(10) UNSIGNED NOT NULL ');

END IF;

IF typeName = 'int' THEN

SET @typeMysql = 'INT';

END IF;

IF typeName = 'string' THEN

SET @typeMysql = CONCAT('CHAR(',typeDimension,') NULL ');

END IF;

IF typeName = 'memo' THEN

SET @typeMysql = 'TEXT NULL ';

END IF;

IF typeName = 'double' THEN

SET @typeMysql = CONCAT('DOUBLE ');

END IF;

SET @sql = CONCAT('ALTER TABLE ', tableName , ' ADD ', columnName ,' ', @typeMysql , ';');

PREPARE s1 from @sql;

EXECUTE s1;

END//

DELIMITER ;

-- Dumping structure for procedure db.AddForeignKey

DELIMITER //

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddForeignKey`(IN `tableName` TEXT, IN `columnName` TEXT, IN `tableName2` TEXT, IN `columnName2` TEXT)

BEGIN

SET @sql = CONCAT('ALTER TABLE ',

tableName ,

' ADD CONSTRAINT FK\_',tableName, '\_', columnName,' FOREIGN KEY (',

columnName,

') REFERENCES ',

tableName2,

' (',columnName2,

');');

PREPARE s1 from @sql;

EXECUTE s1;

END//

DELIMITER ;

-- Dumping structure for procedure db.AddIndex

DELIMITER //

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddIndex`(IN `TableName` TEXT, IN `ColumnName` TEXT)

BEGIN

SET @sql = CONCAT('ALTER TABLE ', tableName , ' ADD INDEX ' , columnName , '(', columnName,');');

PREPARE s1 from @sql;

EXECUTE s1;

END//

DELIMITER ;

-- Dumping structure for procedure db.AddTable

DELIMITER //

CREATE DEFINER=`root`@`localhost` PROCEDURE `AddTable`(IN `name` TEXT)

BEGIN

SET @sql = CONCAT('CREATE TABLE IF NOT EXISTS ', name , ' (`TimeStampTable` TIMESTAMP) ENGINE=InnoDB DEFAULT CHARSET=latin1;');

PREPARE s1 from @sql;

EXECUTE s1;

END//

DELIMITER ;

-- Dumping structure for procedure db.ChangeStringColumnLegth

DELIMITER //

CREATE DEFINER=`root`@`localhost` PROCEDURE `ChangeStringColumnLegth`(IN `tableName` TEXT, IN `columnName` TEXT, IN `columnLenght` INT)

BEGIN

SET @sql = CONCAT('ALTER TABLE ', tableName , ' CHANGE COLUMN ', columnName ,' ', columnName, ' CHAR(', columnLenght, ');');

PREPARE s1 from @sql;

EXECUTE s1;

END//

DELIMITER ;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=1 \*/;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

**TESTING**

Testing has started on the single stored procedure with black box testing.

No error will be handled in stored procedure.

SchemaDb based on Entity Framework has tested with black box testing too for each procedure. No error will be handled in methods but no crash will occur.

A kind of System testing has performed with a simple application driver but this kind of test must be performed each time library will be used in application.

**DEPLOYMENT**