The object model of splines

Mojca Indihar Štemberger and Janez Grad
University of Ljubljana,
Faculty of Economics,
Kardeljeva ploščad 17
SI-1000 Ljubljana,
Slovenia
Tel: +386 61 1892 400, Fax: +386 61 1892 698
E-mail: mojca.stemberger@uni-lj.si, janez.grad@uni-lj.si

Keywords: splines, object-oriented modeling, object-oriented database, UML, ODMG, Java

Received: June 8, 2000

Data analysis like Data Mining is getting more and more important every day because the organizations want to get their competitive advantage. Splines are piecewise polynomial functions that are used for the analysis by some Data Mining tools. Except that, they are widely used in CAGD applications for designing. The article presents the object model of splines that is independent of the concrete application but can serve as the basis for any application which uses splines and reduces the gap among such applications and their incompatibility. It was modeled by object-oriented modeling language UML. We also propose how to save splines in an ODMG compliant database. The prototype of a system was developed in Java and is presented in the article as well.

1 Introduction

Nowdays every organization has to consider data as an important resource if they want to be successful in every day more competitive world. This is the main reason for the growing use of data analysis tools such as OLAP (On-Line Analytical Processing) or Data Mining (Koutsoukis, Mitra, Lucas, 1999). One of the methods that Data Mining tools use for discovering connection among large amounts of data is the approximation or the regression. In this case the user gets the connection among the variables in the form of mathematical functions and the prediction of values of input variables in the points where they aren't known. The development in the field of mathematics and statistics proved that piecewise polynomial functions called splines served the best for the purpose of approximation or regression (De Boor, 1978), (Friedman, 1991), (Eubank, 1999), (Pagan, Ullah, 1999). The example of Data Mining tool that uses splines is called MARS (Salford Systems) – Multivariate Adaptive Regression Splines.

Another area where splines are widely used is Computer Aided Geometric Design (CAGD) where splines are used for designing. Designing with splines began in France and USA in 1960s because of the needs in car and aviation industry (Farin, 1997).

Splines are briefly introduced in section 2. The reader who is not familiar with the terms can read more about the splines in the extensive literature that is referred to at the References.

In the field of using splines, the problem lies in the gap among different applications that use splines. Since they do not use the common object model, their compatibility may be difficult. Our proposed solution is the use of the object model that is independent of the concrete application, DBMS and programming language, and can serve as the basis for each application that use splines. The object model of the splines was developed by the authors with the well-known Unified Modeling Language - UML (Booch, Rumbaugh, Jacobson, 1999) and is described in section 3.

The article also describes a solution for storing splines in a way that guarantees the maximum possible portability among different applications that use splines (section 4). The solution is based on the common object model. ODMG standard (Cattell et al., 2000) is used for the purpose of storing the produced objects. The prototype of such application was developed in the promising programming language Java as well and is described in section 5.

2 Splines

Polynomials have been widely used for the interpolation and approximation because it is rather simple to calculate their coefficients. But by increasing the number of points to be interpolated or approximated it is not always possible to find satisfying interpolation or approximation within the class of polynomials. Polynomials of a higher degree can have oscillations (Schumaker, 1993), which means that some other class of functions had to be introduced. Splines are such class of functions. They are piecewise polynomial functions that are smooth in the joint points. Spline can be expressed as sequence of polynomial function, one for each polynomial segment, or as a special form of spline like

B-spline or Bézier curve (de Boor, 1978), (Farin, 1997), (Schumaker, 1993).

The splines used by Data Mining tools usually belong to the first group, since in most cases they are expressed as a sequence of polynomials that are smooth in the joint points. Figure 1 shows an example of the cubic spline that consists of two polynomial segments and is smooth in the joint point $x = 1$:

$$y = \begin{cases} x^3; & x \leq 1 \\ 2x^3 - 3x^2 + 3x - 1; & x > 1 \end{cases}$$

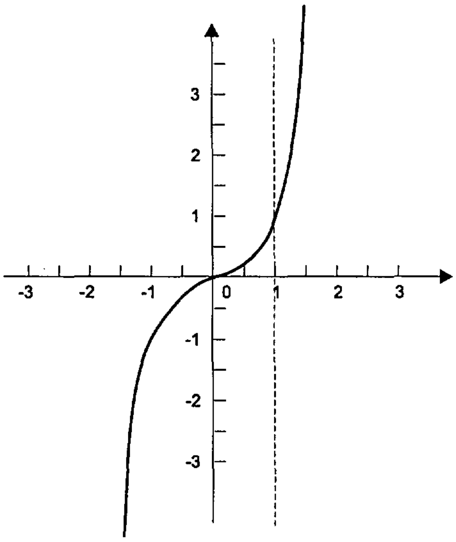

Figure 1: Cubic spline

B-splines are used for the special representation of polynomial splines that have additional advantages such as better numerical stability (Schumaker, 1993). They can be defined in several ways. Probably the easiest way to define them is the recursive formula

$$\begin{aligned} B_{i,1}(x) &= \begin{cases} 1; & t_i \leq x < t_{i+1} \\ 0; & \text{otherwise} \end{cases} \\ B_{i,k}(x) &= \frac{x - t_i}{t_{i+k-1} - t_i} B_{i,k-1}(x) + \\ &\quad + \frac{t_{i+k} - x}{t_{i+k} - t_{i+1}} B_{i+1,k-1}(x), \end{aligned}$$

where $\mathbf{t} = (t_i)_{i=0}^{m+k}$ is the no decreasing sequence of real numbers called the vertices and

$$\frac{x - t_i}{t_{i+k-1} - t_i} B_{i,k-1}(x)$$

and

$$\frac{t_{i+k} - x}{t_{i+k} - t_{i+1}} B_{i+1,k-1}(x)$$

is considered as 0 when $t_{i+k-1} - t_i = 0$ or $t_{i+k} - t_{i+1} = 0$.

B-spline of order k $B_{i,k}(x)$ is a polynomial of order k that has an important property:

$$\begin{aligned} B_{i,k}(x) &\geq 0, \text{ when } x \in [t_i, t_{i+k}], \\ B_{i,k}(x) &> 0, \text{ when } x \in (t_i, t_{i+k}), \\ B_{i,k}(x) &= 0, \text{ when } x \notin [t_i, t_{i+k}]. \end{aligned}$$

The reason for introducing B-splines is the definition of B-spline function and B-spline curve. The first one is mostly used for the approximation or interpolation and the second for designing.

B-spline function of order k with the sequence of vertices $\mathbf{t} = (t_i)_{i=0}^{m+k}$ is every linear combination of B-splines of order k, that is

$$f(x) = \sum_{i=0}^{m} \alpha_i B_{i,k}(x),$$

where α_i, $i = 0, 1, \ldots, m$ are real numbers. Several algorithms for interpolation or approximation with B-spline functions are known (De Boor, 1978), (Eubank, 1999).

B-spline curve is expressed parametrically with (Farin, 1997)

- B-splines of order k,
- the sequence of vertices $\mathbf{t} = (t_i)_{i=0}^{m+k}$ and
- the control points $V_0, V_1, \ldots, V_m$, where each point has x and y koordinate:

$$V_i = (x_i, y_i).$$

Figure 2 shows the cubic B-spline curve (order 4) with control points V_0, V_1, V_2, V_3, V_4, $V_5 = V_0$ and vertices $t_0 = -1$, $t_1 = 0$, $t_2 = 1$, $t_3 = 2$, $t_4 = 3$, $t_5 = 4$, $t_6 = 5$, $t_7 = 6$, $t_8 = 7$, $t_9 = 8$. The control polygon is the broken line that connects the control points. It is drawn in the figure too.

B-spline curve actually consists of two B-spline functions, one for each coordinate. By adding another coordinate we get B-spline surfaces that are also used for designing.

It is important to notice that B-splines, B-spline functions and B-spline curves are very convenient to be processed by computers since very little data have to be stored for their reproduction. One of their main advantages is that changing one segment only makes influence on the segments that are close to it.

Bézier curves are most widely applied in CAGD tools but they are actually the special form of B-spline curves. They were introduced in 1960s by Bézier and De Casteljau but they are named after Bézier while De Casteljau's work has not been published. They are defined as

$$\mathcal{B}_n[V_0, V_1, \ldots, V_n; a, b; t] = \sum_{i=0}^{n} V_i B_i^n(t), \; a \leq t \leq b,$$

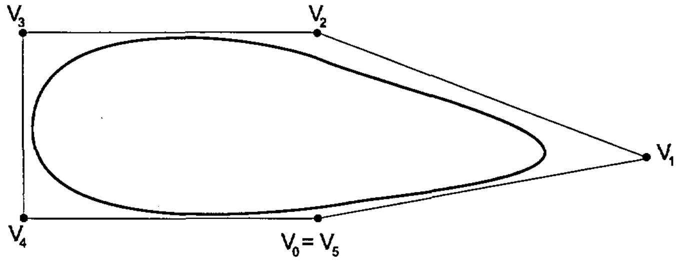

Figure 2: B-spline curve and its control polygon

where

$$B_i^n(t) = \binom{n}{i}\frac{(t-a)^i(b-t)^{n-i}}{(b-a)^n}, \qquad a \le t \le b, \quad i = 0, 1, \ldots, n$$

are Bernsteins polynomials of degree n and points $V_0, V_1, \ldots, V_n$ are control points.

Designing with B-spline curves or Bézier curves (Farin, 1997) is very easy. The control polygon mimics the curve since the curve is close to the polygon. If a control point is moved the curve segments close to it move too. Every operation such as rotating, moving or changing the size of the curve can be done by performing it on the control polygon and by redrawing the curve. In case a desired designing could not be done by moving existing control points, additional control points can be introduced. This process is called subdivision (Farin, 1997).

There are several other classes of spline functions that are described in (Farin, 1997) and (Schumaker, 1993). Other interesting properties of splines are described there as well.

3 The object model of the system

Standard language for object-oriented modeling UML has been used for the development of the model, since it is possible to map from a model in the UML to an object-oriented programming language such as Java or C++, or even to tables in a relational database or the persistent store of an object-oriented database (Booch, Rumbaugh, Jacobson, 1999). The developed model which is described bellow can serve as the basis for any application that uses splines because it is independent of the programming language and the application. Only the parts of the system that are independent of the concrete application have been modeled, because any object model can be easily extended with the parts that are specific for the application. The model is divided into four packages. If an application needs only a specific kind of splines then only one or more packages have to be employed. It is important to point out that in the object model not only data but also the operations on splines are modeled.

We used several diagrams for modeling the system. The behavior of the system was modeled with the use case diagram and some sequence diagrams. Figure 3 shows the use case diagram of the system. As it is evident from the figure we have two kinds of users (`Researcher` and `Designer`) that are presented as actors. The first one is in interaction with the use case `Approximation with splines` and `Interpolation with splines`, but the second one is in interaction with the use case `Designing with splines`. All the mentioned use cases include the use case `Saving spline` and the third one includes `Changing shape of spline` and `Create spline`.

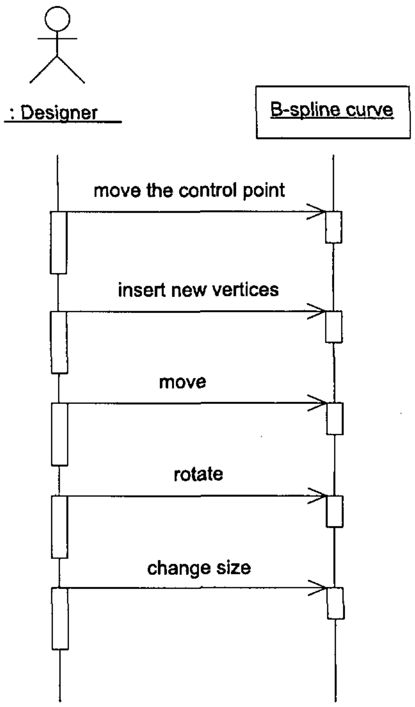

Figure 4: Sequence diagram for B-spline curve

We have modeled use case `Changing shape of spline` with sequence diagram. It is shown in Figure 4 for the B-spline curve. Similar diagram can be drawn for the Bézier curve.

The structure of the system was modeled with class diagrams. A type of certain parameters has been chosen but the rest are left to the phase of more detailed modeling or

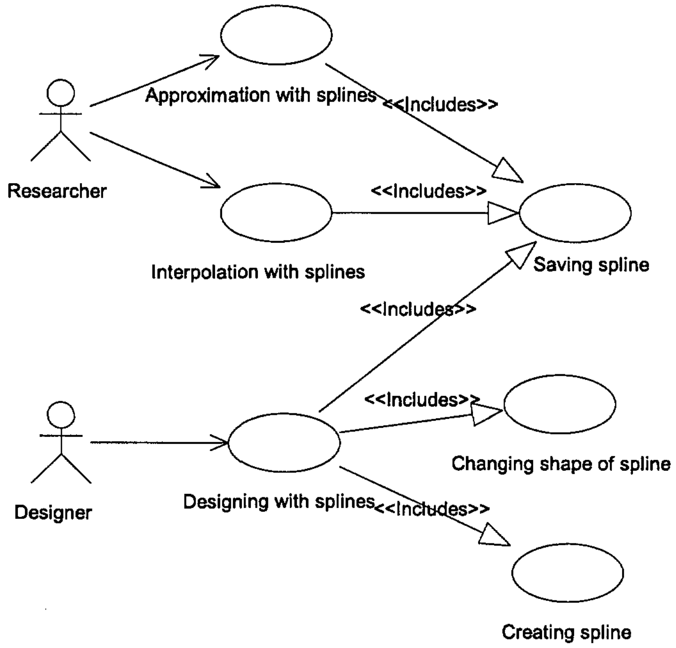

Figure 3: Use case diagram of the system

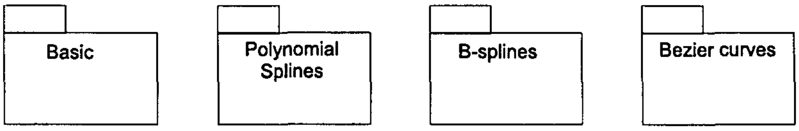

Figure 5: Main class diagram of the system

implementation. The messages captured in sequence diagrams are very important since they helped us to determine the operations of the classes. Figure 5 shows the main class diagram of the system that is divided into four packages.

Package Basic captures classes that form the basis for all other classes such as Polynomial, Interval and Point. As evident from Figure 6, which shows the main class diagram of the package, all the modeled classes have constructor create() and destructor delete(). Classes have also other attributes and operations that are self-describing. Classes LinearFunction, QuadraticFunction and CubicFunction are modeled specially because they represent the most widely used classes of functions for the approximation or interpolation. They inherit all attributes and operations from class Polynomial but some of them are overridden since they can be implemented differently.

The package called Polynomial splines consists of the classes needed to represent piecewise polynomial functions or polynomial splines. The central class of the package is called PiecewisePolynomialFunction but we have also modeled some other classes that all inherit attributes and operations from this class. The class diagram and details about attributes and operations are evident from Figure 7. All classes in the package are collections of polynomials where collection can be a list, an array or of some other type. The best way of implementing operations is to call the methods of the corresponding polynomial segments. Attribute continuityAtJointPoints represents a degree of continuity at every joint point (-1 stands for uncontinues function, 0 for continues, 1 for first derivative being continues, etc.). Other classes in the package inherit everything from the main class but some attributes are different. Similar as for polynomials some operations can be implemented differently. In case one's objective is to implement some special algorithms for approximation or interpolation with splines, one can simply introduce a new class to the system that inherits attributes and operations from modeled classes and have some additional attributes and operations.

Package B-splines consists of classes that model B-splines, B-spline functions and B-spline curves. The relationships of the classes are shown in Figure 8. The meaning of most attributes and operations is obvious but some of them need comments. Many operations of the class B-Curve are derived from translating the messages from the sequence diagram to corresponding operations. Operation moveControlPoint(), for example, serves to moving some control point of the curve to some other lo-

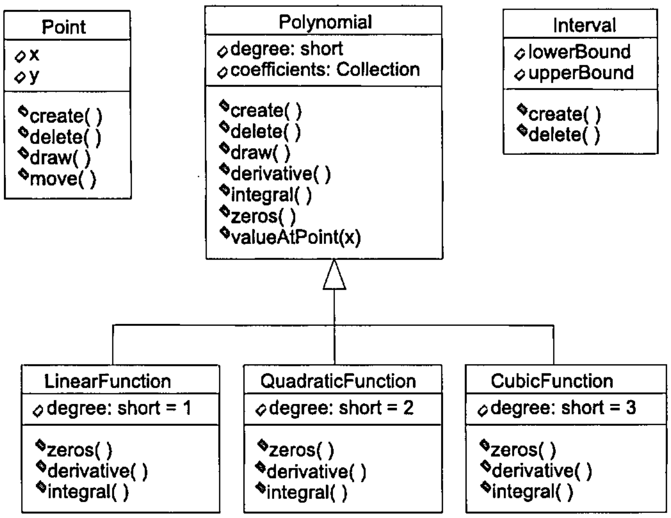

Figure 6: Class diagram of the package `Basic`

cation. In that case, a part of the curve has to be redrawn. The classes are connected with the relationship aggregation. New classes that inherit from the modeled classes can be simply added to the package.

Package `Bezier curves` is very similar to the package `B-splines` and is described in (Indihar Štemberger, 2000), where many other details about the model are also described.

It is very important to understand that the described model is independent of the purpose of using splines. It can serve as the basis of the system where splines are used for approximation or interpolation and of the system for designing with splines as well, because it is a common dominator of every such system.

4 Storing splines

In modern information systems we have to deal with complex objects. Classical relational DBMS are not the appropriate solution for storing such objects. The development in this field has gone further with the concepts of object or object-relational databases. The concept of the database has been extended to the point that it includes the execution of the processes as well (Domanjko, Heričko, Rozman, 1997). Therefore ono does not need to distinguish between applications and databases anymore.

The idea is to store splines in object-oriented database. Namely, they have already been used for storing splines produced by CAD applications (Cattell, 1994), but not in connection with Data Mining applications.

The fact is that there is no general agreement in the literature (Cattell, 1994), (Kim, 1995), (Khoshafian, 1995) about the definition of the object-oriented database. We understand the object-oriented database as the combination of object-orientation and database capabilities. It implies that the so-called "true" object-oriented databases and object-relational databases both belong to that category.

The ODMG standard (Cattell et al., 2000) was used to store splines. The standard is independent of the programming language and the DBMS. It can be used for storing objects in "true" object-oriented database and with special mappings, which map objects into relations, to the relational database as well.

By the ODMG standard the schema of object-oriented database can be written in one of the programming languages that have the ODMG mapping (C++, Smalltalk and Java) or in independent language ODL (Object Definition Language). Class diagrams from section 3 can be converted to ODL. Figure 9 shows the example of such schema for the classes from Figure 7.

The use of ODMG standard ensures the portability among different supported programming languages and also among different DBMS.

5 Prototype

The prototype of the system has been developed as well. We used Java programming language and the Poet ODBMS. The user interface of the prototype is presented in Figure 10. A user can enter the points he wants to approximate by a cubic spline simply by clicking the mouse. The approximation is obtained by algorithm described in (Eubank, 1999). The produced cubic spline corresponds to the class `PiecewiseCubicFunction` in the model. Actually it is the ancestor of that class since it has additional operation called `approximate()` which serves to obtaining the approximation to the given data. The spline can be stored in the database and the stored splines can be retrieved from the database.

One has to mention again that ODMG standard can be used for "true" object-oriented DBMS like the one we used

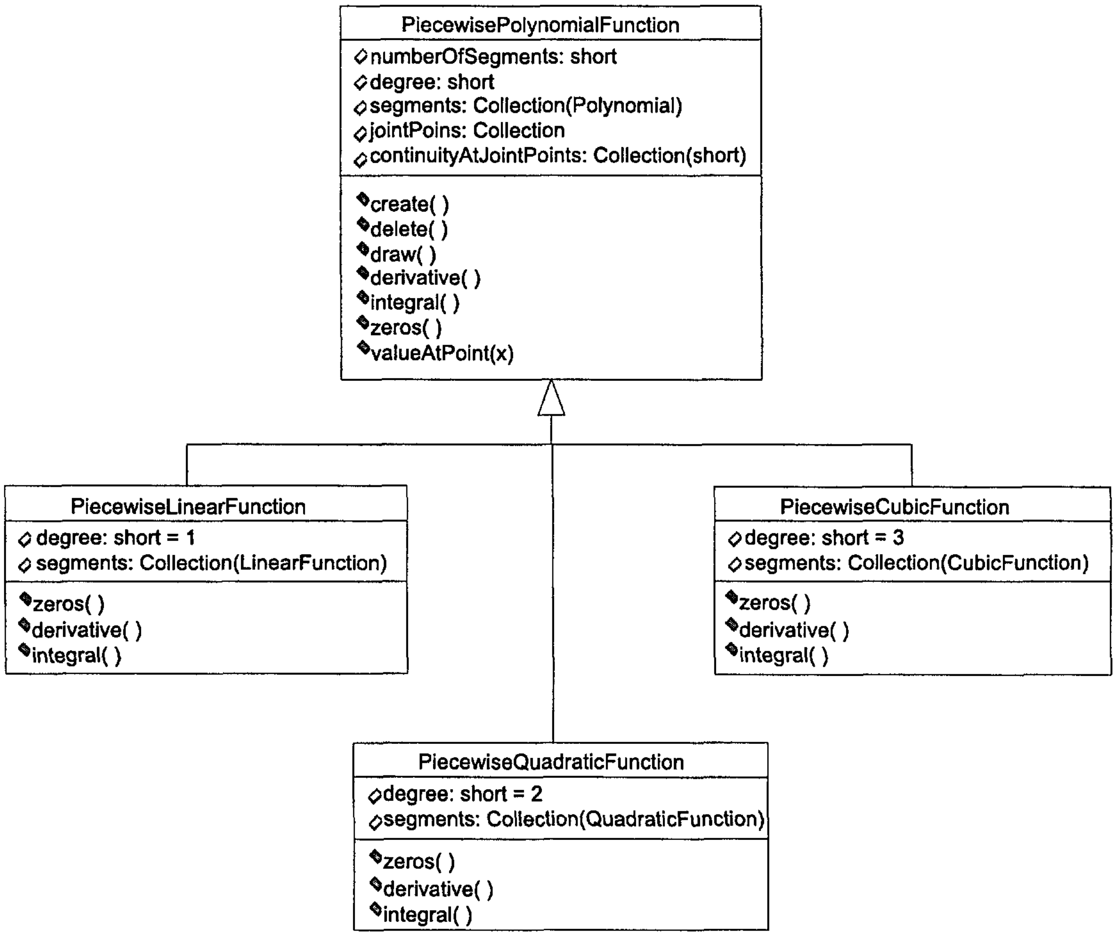

Figure 7: Class diagram of the package `Polynomial splines`

and also with the special mapping (Java Blend for example) for the relational DBMS.

6 Conclusion

We believe that the presented model can serve as a common dominator of a system that uses splines. Each application of that kind can be based on it and extended with additional classes. We are shure that the choice of UML modeling languege was correct since the variety of the environments which are suitable for the implementation of the UML models is very reach. The use of the standard ODMG, that has been developed by the major vendors of object-oriented database management systems and is the only existing standard in this field (Alagić, 1999), for describing the model of saving splines guarantees the maximum possible portability of the system. Produced objects can be stored in any DBMS that supports the standard. It is important to add that not only attributes but also the methods of the classes are persistent. The prototype has been developed with the purpose of expressing our ideas. Our intention was not developing Data Mining tool but in our opinion vendors of such tools should consider these ideas.

References

[1] Alagić Suad: A Family of the ODMG Object Models, Third East European Conference ADBIS'99 (Lecture Notes in Computer Science, Vol 1691), Berlin et al.: Springer, 1999, pp. 14-30.

[2] Booch Grady, Rumbaugh James, Jacobson Ivar: The Unified Modeling Language User Guide. Reading [etc.]: Addison Wesley Longman, 1999.

[3] Cattell Rick G. G.: Object Data Management. Reading: Addison-Wesley Publishing Company, 1994.

[4] Cattell Rick G. G., Barry Douglas, Bartels Dirk, Berler Mark, Eastman Jeff, Gamerman Sophie, Jordan David, Springer Adam, Strickland Henry, Wade Drew: The Object Database Standard ODMG 3.0. San Francisco: Morgan Kaufmann Publishers, 2000.

[5] De Boor Carl: A practical Guide to Splines. New York: Springer-Verlag, 1978.

[6] Data Mining and Knowledge Discovery in Databases, published by UNICOM Seminars, 1998.

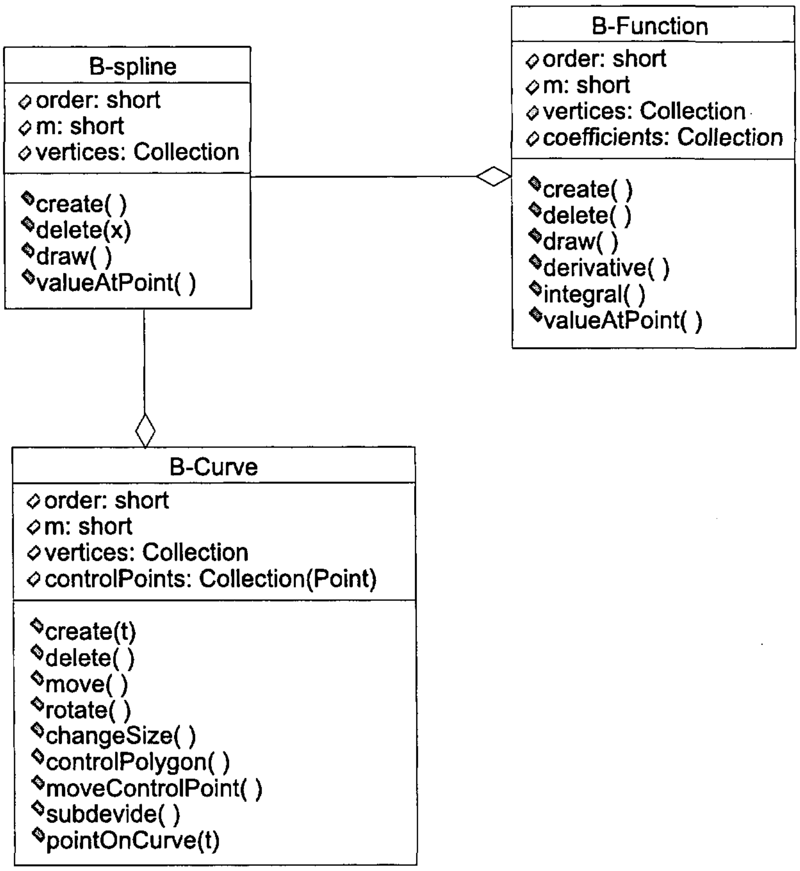

Figure 8: Class diagram of the package `B-splines`

[7] Domanjko Tomaž, Heričko Marjan, Rozman Ivan: The usage of object-oriented databases, COTL, 1 (1997), 2, in Slovene. [URL: http://lisa.uni-mb.si/cot/cotl/april97/index.shtml].

[8] Eubank Randall L.: Nonparametric Regression and Spline Smoothing (Statistics, Textbooks and Monographs, V. 157), Marcel Dekker, 1999.

[9] Farin Gerald: Curves and Surfaces for CAGD. Boston: Academic Press, 1997.

[10] Friedman J.H.: Multivariate Adaptive Regression Splines. Annals of Statistics, 19 (1991), pp. 1-141.

[11] Indihar Štemberger Mojca: Splines in object-oriented database environment. Ph.D. Thesis, Ljubljana: University of Ljubljana, Faculty of Economics, 2000.

[12] Kim Won, editor: Modern Database systems – The Object Model, Interoperability and Beyond. New York: Addison-Wesley Publishing Company, 1995.

[13] Khoshafian Setrag, Abnous Razmik: Object Orientation. New York: John Wiley & Sons, 1995.

[14] Koutsoukis N.S., Mitra G., Lucas C.: Adapting Online Analytical Processing for Decision Support: The interaction of information and decision technologies, Decision Support Systems, 26 (1999), pp. 1-30.

[15] Pagan Adrian, Ullah Aman: Nonparametric Econometrics, Cambridge: Cambridge University Press, 1999.

[16] Poet, [URL: http://www.poet.com/].

[17] Salford Systems, [URL: http://www.salford-systems.com/].

[18] Schumaker Larry L.: Spline Functions: Basic Theory. Malabar, Florida: Krieger Publishing Company, 1993.

```
class PiecewisePolynomialFunction
{
    attribute short degree;
    attribute short numberOfSegments;
    attribute Array<Polynomial> segments;
    attribute array<float> jointPoints;
    void create();
    void delete() raises (no_function);
    void draw();
    float valueAtPoint(in float x) raises(not_defined);
    bag<float> zeros();
    PiecewisePolynomialFunction derivative();
    PiecewisePolynomialFunction integral();
};

class PiecewiseLinearFunction extends
    PiecewisePolynomialFunction
{
    attribute Array<LinearFunkction> segments;
    bag<float> zeros();
    array<float> derivative();
    PiecewiseQuadraticFunction integral();
};
```

```
class PiecewiseQuadraticFunction extends
    PiecewisePolynomialFunction
{
    attribute Array<QuadraticFunction> segments;
    bag<float> zeros();
    PiecewiseLinearFunction derivative();
    PiecewiseCubicFunction integral();
};

class PiecewiseCubicFunction extends
    PiecewisePolynomialFunction
{
    attribute Array<CubicFunction> segments;
    bag<float> zeros();
    PiecewiseQuadraticFunction derivative();
    PiecewisePolynomialFunction integral();
};
```

Figure 9: ODL definitions for some classes

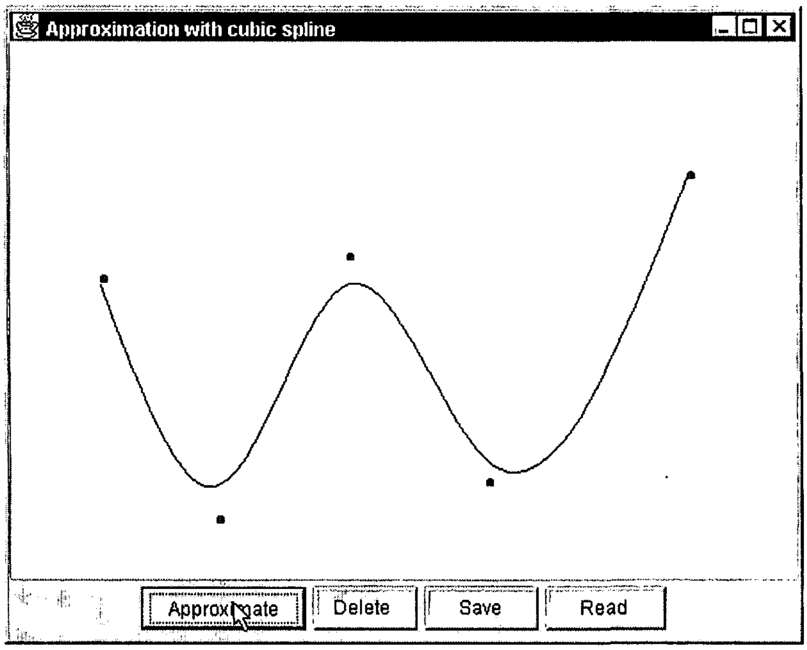

Figure 10: The user interface of the prototype