

Delphi – možnosti grafů

Bc. Tomáš Selucký, Ústav statistiky a operačního výzkumu, Provozně ekonomická fakulta, Mendelova zemědělská a lesnická univerzita v Brně,
tselucky@centrum.cz

Abstrakt

Příspěvek se zabývá hodnocením vybraných komponent vývojového prostředí Delphi. Vlastní práce se skládá ze stručného popisu komponent, obsahujících grafy, shrnutí jejich vlastností s určitými doporučeními a konkrétních algoritmických problémů včetně ukázek zdrojového kódu. Závěrečná část potom hodnotí komponenty grafů z hlediska jejich možností a použití.

Klíčová slova

Delphi, graf, komponenta

Abstract

This contribution deals with evaluation of chosen components of develop environment Delphi. This paper focuses on concise description of components, involving charts, summary of their properties with some recommendations and on concrete algorithmic problems, including demonstrations of source code. The final part of this contribution evaluates components of charts in view of their possibilities and usage.

Key Words

Delphi, chart, component

1. Úvod

Vědní obor statistiky a jí vědy příbuzné se snaží především o kvantitativní pohled na realitu, tudíž jejich typickým vyjadřovacím prostředkem musí být číslo. Jako formalizované nástroje číselného vyjádření pak můžeme posuzovat dva základní¹. Jsou to:

- statistické tabulky a
- statistické grafy.

Výhoda číselného vyjádření pomocí grafu spočívá bezesporu v jeho názornosti a přehlednosti, přičemž samotné statistické grafy pak můžeme dělit následujícím způsobem. Jedná se o grafy²: prezentační, konstrukční a nomogramy.

Kromě možnosti takové grafy vytvářet v některé z kancelářských aplikací je možné využít některého nástroje ze statistických programových vybavení, popřípadě použít vývojové nástroje pro tvorbu aplikací, do kterých je potřeba implementovat komponenty grafů nebo jim komponenty příbuzné a související.

Práce se zabývá poslední uvedenou možností, tedy použitím vývojového softwaru fy Borland, a to Delphi. Tento produkt v sobě ukrývá spoustu možností, kterých je možno využít pro tvorbu vlastních aplikací, vytvářených přímo na míru potřebám výuky.

2. Cíl a metodika

Cílem práce je zhodnocení možností grafů v oblasti jejich vlastností a algoritmických řešení, které umožňují Delphi použít. Částečně jsou nastíněny i vybrané části zdrojového kódu.

Při tvorbě této práce byl použit vývojový nástroj Delphi 5 Standard fy Borland. Samotné Delphi představují kombinaci moderního programovacího jazyka, integrovaného vývojového prostředí (IDE) a vizuální knihovny komponent (VCL). Prostředí aplikace je uživatelsky velmi příznivé a ovládání nevybočuje z běžných aplikací, které pracují v grafickém prostředí. Stěžejní částí je poté programovací jazyk Object Pascal, na němž je postaveno celé IDE a VCL.

Jak již bylo částečně nastíněno, práce se zabývá komponentami obsahující grafy. V Delphi existuje několik takovýchto komponent, nicméně za stěžejní byly považovány první dvě uvedené. Graf, nebo alespoň nějakou jeho část můžeme nalézt v těchto oblastech:

- TChart,
- TDBChart,
- TDecisionGraph,

¹ více Mínařík, B.: *Statistika I, Popisná statistika (1. část)*, s. 19-21

² více l.c., s. 21-26

- TQRChart,
- TChartFx a
- TVtChart.

Každou z těchto komponent lze poměrně snadno zapracovat do vytvářené vlastní aplikace, avšak ve většině případů bývají použity právě první dvě.

Samotné výsledky práce jsou rozděleny do tří kapitol:

- první část se snaží o zjednodušené vymezení výše vyjmenovaných komponent se stručným popisem,
- druhá část se konkrétně zabývá jednotlivými vlastnostmi komponenty TChart, přičemž přináší i určité tipy a doporučení a
- v poslední části se zaměřujeme na praktické použití takovýchto grafů a jsou zde vysvětleny určité procedury včetně zdrojového kódu.

Poslední kapitola poté přináší konkrétní závěry, které vyplývají z jednotlivých částí kapitoly předcházející.

3. Výsledky

3.1 Komponenty grafů v Delphi

Již na první pohled samotné Delphi oplývají spoustou možností v oblasti grafů. Nyní však krátké shrnutí předcházejícího výčtu s vymezením specifikací:

- *TChart* – je nejdůležitější komponentou ve knihovně TeeChart. Vychází z TPanel a také dědí všechnu její funkcionalitu. Ve zkratce lze říci, že TChart je vlastně standardní komponentou TPanel s rozšířenými možnostmi grafiky.
- *TDBChart* – vychází ze TChart a dědí veškerou funkcionalitu. Je-li řada grafu připojena ke komponentě TDBChart, je zobrazena jako datový zdroj u vlastnosti Series.
- *TDecisionGraph* – rozhodovací graf ukazuje jediný součet, který je kalkulován přes počet úrovní. Osy grafu korespondují s dimenzemi, které jsou poskytovány u rozhodovacího zdroje objektu.
- *TQRChart* – komponenty TQRChart pocházejí z komponent QuickReport TQRPrintable.
- Zvláštními případy jsou poté *TChartFx* a *TVtChart*, které se vymykají určitým grafickým standardům v této verzi Delphi.

3.2 Komponenta třídy TChart a popis jejích vlastností

Nejdůležitější komponentou je v rámci knihovny TeeChart jednoznačně komponenta TChart.

Lze ji shrnout následujícími charakteristikami:

- vytváření komponenty jak v době návrhu, tak v době běhu aplikace,
- možnost existence mnoha grafů na jednom formuláři a
- komponenty knihovny TeeChart mohou být také ve formě ActiveX.

Nyní je třeba shrnout vybrané vlastnosti této nejdůležitější komponenty. Následující tabulka tedy podává podrobný výklad o jednotlivých charakteristikách i s částečným výkladem (Tab. 3.2.1).

Tab. 3.2.1: Vybrané nestandardní vlastnosti komponenty TChart

Vlastnosti	Popis
AllowPanning	Znamená umožnění „sledování“ a nastavení kontroly v oblasti runtime rolování – vybíráme z možností: <ul style="list-style-type: none">• <i>pmNone</i> – zakázání rolování,• <i>pmVertical</i> – je umožněno pouze vertikální rolování,• <i>prHorizontal</i> – rolování pouze horizontálně,• <i>pmBoth</i> – povoleno rolovat v obou směrech. Standardní nastavení je <i>pmBoth</i> .
AllowZoom	Jde o takové nastavení, při němž je možné, či nikoliv, na grafu provést runtime Zoom, tedy je možné zvětšit některou z jeho částí. Možnosti: <ul style="list-style-type: none">• <i>true</i> – zoomování povoleno,• <i>false</i> – zákaz zoomování. Je-li zoomování povoleno, pak je možné přidržet levého tlačítka myši a tažením kurzoru zvětšit, popř. zmenšit požadovanou část grafu.
anchors	Souvisí též s nastavením vlastnosti <i>Align</i> – jde o přidržení u některého z okrajů „rodičovské“ komponenty.
AnimatedZoom AnimatedZoomSteps	Je-li nastavena vlastnost <i>AllowZoom</i> na <i>true</i> a dojde-li skutečně k zoomování, pak v případě nastavení vlastnosti <i>AnimatedZoom</i> na <i>true</i> , bude tento zoom proveden právě v tolika krocích, kolik je

	<p>zadáno ve vlastnosti <i>AnimatedZoomSteps</i>. Standardně je vlastnost <i>AnimatedZoom</i> nastavena na <i>false</i>, nicméně jedná se o velmi sympatický grafický efekt. Pokud je graf navíc větších rozměrů, jednoznačně doporučuji přenastavit na hodnotu <i>true</i>.</p>
AxisVisible	<p>Booleanovská volba, zda budou vidět osy grafu (jsou celkem čtyři, touto volbou lze ovládat všechny najednou).</p>
BackColor	<p>Volba základního pozadí formuláře.</p>
BackImage BackImageInside BackImageMode	<p>Určuje obrázek na pozadí grafu, a to buď místo barevného pozadí (viz. předchozí příklad), nebo v panelu, na kterém je graf umístěn. Ve vlastnosti <i>BackImage</i> (TBitmap) je umístěn samotný obrázek. Je-li hodnota vlastnosti <i>BackImageInside</i> nastavena na <i>true</i>, poté je obrázek zobrazen přímo jako pozadí, v opačném případě (<i>false</i>), je vykreslen v panelu. Související vlastnost <i>BackImageMode</i> pak určuje, jakým způsobem bude obrázek vykreslen. Můžeme volit z následujících možností:</p> <ul style="list-style-type: none"> • <i>pbmStretch</i> – obrázek bude přizpůsoben rozměrům grafu, • <i>pbmTile</i> – pozadí bude obrázkem „vykachlíkováno“, • <i>pbmCenter</i> – obrázek nebude přizpůsoben, bude pouze umístěn na střed. <p>V případě použití obrázku je třeba si uvědomit, že se jedná o zbytečné zatížení velikosti aplikace obrázkem.</p>
BackWall	<p>První z vlastností řešená pomocí dalších dialogů, v tomto případě je možné nastavit vlastnosti levé, spodní i zadní stěny grafu, jejich pozadí, orámování, atd.</p>
BevelInner BevelOuter BevelWidth	<p>Určuje ohraničení celé komponenty grafu – <i>BevelInner</i>: vnitřní rám, <i>BevelOuter</i>: vnější rám, <i>BevelWidth</i> – šířka vnitřního a vnějšího rámu. Možnostmi jsou následující:</p> <ul style="list-style-type: none"> • <i>bvNone</i> – okraj neexistuje, • <i>bvLowered</i> – okraj vypadá nižší, • <i>bvRaised</i> – okraj působí vyšším dojmem, • <i>bvSpace</i> – okraj působí jako prázdné místo.
BottomAxis	<p>Jde o další vlastnost, která vzhledem ke své složitosti musí být řešena dalšími dialogy k tomu určenými. Důležité je především nastavení</p>

	měřítka, titulků, nápisů, stupnicí, pozic, atd.
BottomWall	Viz. vlastnost <i>BackWall</i>
Chart3DPercent	Jedná-li se o graf, který má být vyjádřen 3D, pak hodnota určuje „velikost třetího rozměru grafu“. Hodnoty jsou v rozsahu 1-100 (%).
ClipPoints	V případě volby hodnoty <i>true</i> je zajištěno, že vykreslení bodů bude uvnitř hranic grafu, čímž se předchází přepsání jiných částí grafů.
DepthAxis	Jedna z dalších vlastností, které si vyžádali samostatný formulář. Viz. vlastnost <i>BottomAxis</i> .
Foot	Další z vlastností, která musí být ošetřena širším způsobem. Řeší se samozřejmě font, okraj, vzorek, pozadí, zarovnání, atd.
Frame	Nastavení, které se týkají rámu grafu.
Gradient	Tato vlastnost je opět jednou ze složitějších, a proto si také vyžádala samostatný formulář. Přináší shrnutí v podobě barevných schémat a obrázků.
LeftAxis	Jedna z dalších vlastností, které jsou řešeny v rámci Delphi samostatným formuláře. Viz. vlastnost <i>BottomAxis</i> .
LeftWall	Více viz. <i>BackWall</i> .
Legend	Formulář, který byl zaveden z důvodu používání legendy u grafů.
MaxPointsPerPage	Je-li nastavena hodnota větší než <i>nula</i> (nula nastavena implicitně), pak dochází k vnitřnímu rozdělení řad bodů na stránce. V praxi to znamená rozdělení na určité skupiny sloupců (počet odpovídá zadanému počtu), které budou postupně zobrazeny. Je potom logické, že poslední strana obsahuje méně řad bodů. Tato vlastnost úzce souvisí s vlastností <i>Page</i> .
Monochrome	Bude-li hodnota nastavena na <i>true</i> , poté dochází k vykreslení grafu pouze v barvách černé a bílé. V opačném případě jsou zobrazeny barvy.
Page	Vlastnost <i>Page</i> určuje počet stran grafu. Je-li <i>MaxPointsPerPage</i> větší než <i>nula</i> , dojde k vnitřnímu rozdělení všech řad bodů na stránky. Navigace potom probíhá pomocí následujících příkazů: <ul style="list-style-type: none"> • předchozí stránka: <code>Chart1.PreviousPage</code>, • následující stránka: <code>Chart1.NextPage</code>.
PrintProportional	Je-li hodnota nastavena <i>true</i> , pak dojde k tisku v přibližně stejné

	velikosti jako na obrazovce (princip WYSIWYG).
RightAxis	Viz. vlastnost <i>BottomAxis</i> .
ScaleLastPage	Velmi zajímavá vlastnost – udává, jakým způsobem bude zobrazena poslední stránka grafu (předpokládá se použití vlastností <i>Page</i> a také <i>MaxPointsPerPage</i>).
SeriesList	Pokud nejsou data zadávána při běhu aplikace, tedy jsou řešena předchozími výpočty, jedná se o nejdůležitější vlastnost. Opět je řešena pomocí složitějšího formuláře.
Title	Vlastnost, která řeší veškeré titulkování. Má složitější podobu, a proto je řešena dalším formulářem.
TopAxis	Viz. vlastnost <i>BottomAxis</i> .
View3D	Je-li hodnota nastavena na <i>true</i> , poté se přidává třetí rozměr grafu.
View3DOptions	Pokud je graf typu 3D, pak je nutné zadat vše, co s tím souvisí. Tato vlastnost je opět řešena samostatným formulářem.
View3DWalls	Je-li hodnota <i>true</i> , je stěnám přidán 3D efekt.

3.3 Praktické použití komponenty Chart

3.3.1 Zadávání údajů

Nejdůležitější praktickou otázkou při řešení komponenty *TChart* je bezpochyby obtížnost zadávání dat do ní. Začněme však od začátku. Umístění komponenty je velice jednoduchou záležitostí, stačí totiž pouze vyhledat panel, který nese název *Additional*, a poté kliknout na návrhové okno formuláře.

Druhým krokem musí být ošetření některých záležitostí, týkajících se především grafické stránky, v úvodní události formuláře, tedy proceduře *OnCreate*. Nyní však už k zadání příslušných údajů, které mají být v grafu vykresleny. Základní a jednoduchou možností je využití následujícího postupu:

1. specifikace vlastnosti *SeriesList* – řešíme kliknutím na tlačítko „trojtečky“ v nástroji zvaném *Object Inspector*,
2. tvorba nové řady – tlačítko *Add...*,
3. výběr grafu – kruhový, sloupcový, páskový, atd.,
4. zadání zdrojového kódu s určitými hodnotami (předpokladem je pojmenování úvodní řady *Series1*), a to např. takto:

```

...
Series1.Clear; //vyčištění původních hodnot
Series1.Add(40, 'První sloupec', clBlue); //zadání první sloupce
Series1.Add(60, 'Druhý sloupec', clRed); //zadání druhého sloupce
Series1.Add(50, 'Třetí sloupec', clWhite); //zadání třetího sloupce
...

```

Obvykle však pracuje s více hodnotami, tudíž nelze takovýmto způsobem jednotlivé hodnoty neefektivně zadávat. Proto můžeme využít například pole, do něhož se hodnoty uloží, a poté dojde jen k jejich výpisu prostřednictvím následujícího zápisu:

```

...
type //definice
    pole_hodnot = array [0..9] of byte;
var //deklarace
    pole: pole_hodnot;
    i: byte;
...
Series1.Clear; //vyčištění původních hodnot
for i := 0 to 9 do
    Series1.Add(pole[i], IntToStr(i + 1) + '. sloupec', clBlue);
    //zadání sloupců
...

```

Po zadání hodnot (tato část není vypisována), dojde nejprve ke smazání původních hodnot, které jsou v grafu implicitně zobrazeny, a následně v rámci jednoduchého příkazu cyklu k výpisu hodnot nových.

3.3.2 Export ve formě bitmapy – použití v HTML

Komponentu Chart je možné používat přímo ve vlastních aplikacích, nebo také zařídit její export do formátu *.BMP. Takovýmto způsobem máme možnost zpracovat vykreslené grafy například do lokálních webových stránek³. Jakým způsobem je tedy možné daný graf uložit? Celý zdrojový kód vypadá velmi jednoduše:

```

...
Chart1.SaveToBitmapFile(ExtractFilePath(Application.ExeName) + 'obr.bmp');
...

```

³ Použití soubory typu *.BMP není ve webových stránkách obvyklé a efektivní. Nicméně se nacházíme na lokálním disku, takže přenosové problémy v tomto případě nenastanou.

Do webových stránek se poté grafy dostanou běžným způsobem, a to za pomoci známého tagu pro grafiku – IMG. Samotné kolečko pak můžeme uzavřít opět v Delphi, neboť můžeme použít komponentu WebBrowser, vše zobrazit v naší aplikaci a za pomoci API funkce vyvolat tisk takto vytvořené stránky.

4. Závěr

Vývojové prostředí Delphi představuje výbornou pomůcku každého programátora, který vytváří aplikace směřující na platformu Windows, nicméně cílem této práce nebylo hodnocení celého produktu, ale vybraných komponent, a to konkrétně těch, které se týkají grafů – především pak komponenty TChart. Použití těchto komponent ve vlastních aplikacích přináší určité problémy, které jsou dány především rozsahem veškerých možností, a tudíž nutností tyto možnosti programátorsky řešit a neustále ošetřovat.

Na druhou stranu však struktura jednotlivých grafů, kromě zmíněného velkého množství vlastností, má ve své podstatě jednoduchou, přičemž propracovanou, strukturu, která je tak typická pro samotné Delphi.

Závěrem je tedy třeba říci, že algoritmické řešení je velmi kvalitní a propracované a navíc naprosto zachovává jednoduchý a přehledný zápis v Object Pascalu. Jednoznačnou nevýhodou je potom nutnost řešit a ošetřovat příliš mnoho vlastností samotné komponenty grafu, aby bylo dosaženo požadovaného grafického efektu.

5. Literatura

- [1] MINAŘÍK, B. *Statistika I, popisná statistika (I. část)*. 1. vyd. Brno: MZLU v Brně, 2000. 98 s. ISBN 80-7157-421-X.
- [2] MINAŘÍK, B. *Statistika I, popisná statistika (II. část)*. 1. vyd. Brno: MZLU v Brně, 2000. 107 s. ISBN 80-7157-427-9.
- [3] PÍSEK S. *Delphi – začínáme programovat*. 2. vyd. Praha: Grada Publishing, 2002. 328 s. ISBN 80-247-0547-8.
- [4] PÍSEK S. *Delphi – praktické příklady*. 1. vyd. Praha: Grada Publishing, 2002. 220 s. ISBN 80-247-0323-8.
- [5] SVOBODA L. – VONEŠ P. – KONŠAL T. – MAREŠ M. *1001 tipů a triků pro Delphi*. 2. vyd. Praha: Computer Press, 2002. 390 s. ISBN 80-7226-529-6.
- [6] KADLEC V. *Učíme se programovat v Delphi a jazyce Object Pascal*. 1. vyd. Praha: Computer Press, 2001. 288 s. ISBN 80-7226-245-9.
- [7] KADLEC V. *Delphi, hotová řešení*. 1. vyd. Brno: Computer Press, 2003. 312 s. ISBN 80-251-0017-0.

- [8] HOLAN T. *Delphi v příkladech*. 2. vyd. Praha: BEN – technická literatura, 2001. 208 s. ISBN 80-7300-033-4.
- [9] HLAVENKA J. – SEDLÁŘ R. – HOLČÍK T. – KUČERA M. – SCHNEIDER Z. – MACH J. *Vytváříme WWW stránky a spravujeme moderní web*. 5. vyd. Praha: Computer Press, 2001. 437 s. ISBN 80-7226-494-X.
- [10] Delphi Help.
- [11] Internetové zdroje.