

Dokumentace

Grafický editor automatů

1.1 Návrh

Grafické rozhraní musí obsahovat menu, nástrojovou lištu (angl. *toolbar*), stavový řádek (angl. *statusbar*) a hlavní obsahové okno, na které se bude vykreslovat grafická reprezentace modelů. Většina funkcí musí být dostupná z nástrojové lišty, pouze některé nastavení přímo nesouvisející s ovládáním, je možno dát do menu. U konečných automatů stačí v přechodu napsat pouze čtený znak, u zásobníkových to musí být i čtený a zapisovaný řetězec ze zásobníku, ve formátu:

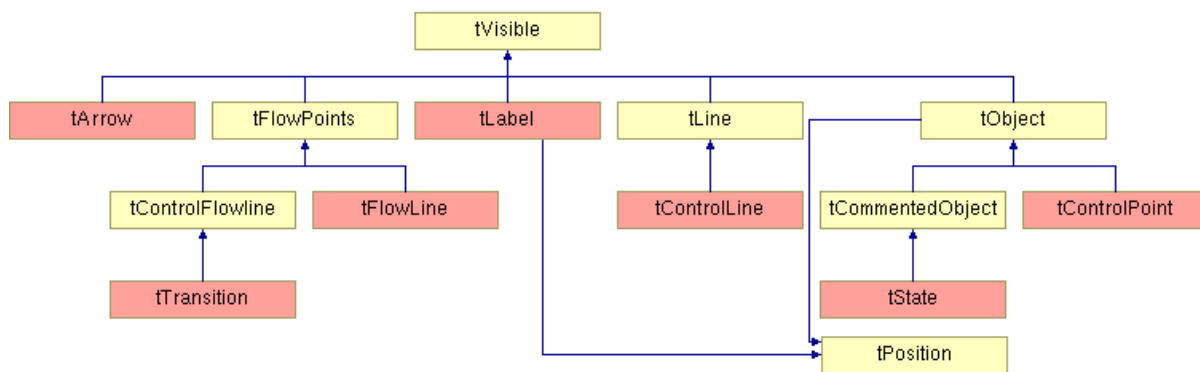
ČtenýZnak-ČtenýŘetězecZeZásobníku/ZapisovanýŘetězecNaZásobník

Pokud je prázdný zapisovaný znak na zásobník nebo čtený znak ze zásobníku, objeví se místo prázdného místa zelené „e“.

Aplikace musí být schopná simulace resp. interpretace automatu. Dále musí být možný export do XML formátu (jeho návrh je v příloze), do LaTeXu a do dalších grafických formátů jako je BMP, JPG atd.

1.1.1 Hierarchie tříd grafických objektů

Červeně jsou zvýrazněny objekty přímo zobrazené v automatu.

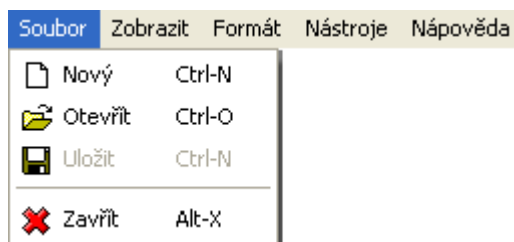


1.2 Používání aplikace

Při spuštění aplikace je na ploše vidět pouze lišta menu, nástrojová lišta a stavový řádek. Jinak je plocha prázdná. V menu jsou povoleny pouze ty položky, které jsou potřeba. To je z toho důvodu, že automaty mohou být dvou druhů, a tak by nebylo chytré zobrazit implicitně jeden druh automatu. To by někteří uživatelé přivítali a některým by to bylo pouze na obtíž. Tento postup, že se při startu

nezobrazí „nic“ a uživatel si vlastní projekt musí vytvořit sám, používá spousta aplikací a tak jsme se touto cestou rozhodli také jít.

Automat můžeme vytvořit dvěma způsoby. Buď vytvoříme automat nový nebo ho načteme ze souboru XML. Při zvolení první možnosti se objeví průvodce (anglicky *wizard*), ve kterém si zvolíme název automatu a zda se jedná o zásobníkový nebo konečný automat. Pokud chceme načíst automat ze souboru, tak pouze vybereme soubor a aplikace si načte všechny potřebné parametry z onoho souboru.



Dále je ještě potřeba se zmínit o položkách ve složce Nápověda. Těmi položkami je samotná Nápověda a informace o aplikaci. V nápovědě lze nalézt stručný přehled funkcí programu. Položka „O programu“ nám poví informace o verzi programu.

1.3 Tvorba automatu

Při vytvoření automatu, ať konečného či zásobníkového, se nám zobrazí pracovní plocha. Úplně vlevo nahoře jsou pro lepší orientaci vidět osy souřadného systému. Při pohybu myši na ploše je vpravo na stavovém řádku (angl. *status bar*) vidět aktuální poloha myši.


1.3.1 Nástroje pro práci

Brzo zjistíte, že spousta akcí lze provést více způsoby. Sami si zvyknete na některý způsob a budete ho používat.




1.3.1.1 Nástrojová lišta

V nástrojové liště (angl. *toolbar*) jsou dány všechny nástroje, které jsou potřeba pro editaci automatu. Nástrojů je 10 a mají klávesové zkratky. Klávesami 1-0 na klávesnici se dá snadno přepínat mezi tlačítky nástrojové lišty. Tlačítka mají i jiné klávesové zkratky, ty si ukážeme při popisu samotných tlačítek. Na první pohled je vidět, že jsou zde 4 tlačítka pro mazání objektů. Některé objekty jsou hodně malé, tudíž by mohlo snadno dojít k překlepům. Například pokud budeme chtít smazat řídicí bod a netrefíme se na něj, smaže se mi celý přechod. Toto se stát nemůže, pokud na smazání každého objektu máme jiný nástroj.

 **Vlastnosti** – klávesa „1“ a „p“


Nejvýhodnější je využívání této funkce z kontextového menu. Umožňuje editovat vlastnosti přechodů a stavů. U přechodů to je pravidlo, u zásobníkového automatu navíc čtené a zapisované znaky na zásobník. U stavů se dá definovat jejich přesná poloha, název a typ stavu. Daný stav pak automaticky nastaví dostatečnou velikost pro pojmутí textového popisu.

Ovládání: Kliknutí levým tlačítkem na přechod či stav.

 **Pohyb** – klávesa „2“ a „m“


Režim pro pohyb s objekty nebo s celým automatem.

Ovládání: Kliknout na objekt či plochu levým tlačítkem myši, držet a táhnout. Při puštění tlačítka se objekt umístí.

 **Přidat stav** – klávesa „3“ a „a“

Přidání stavu. To je možné pouze na místo, kde v okolí není žádný jiný stav.

Ovládání: Prosté kliknutí levým tlačítkem myši na místo, kam chceme stav přidat.

 **Odebrat stav** – klávesa „4“ a „r“

Odebrání stavu. S tím souvisí i odebrání všech přechodů, které jsou na tento stav navázány.

Ovládání: Prosté kliknutí levým tlačítkem myši na stav, který chceme odebrat.

 **Přidat přechod** – klávesa „5“ a „t“

Přidání přechodu mezi dvěma stavy. Přidá se prostá přímka. Další editaci je nutné provést dalšími nástroji.

Ovládání: Kliknout na výchozí stav a táhnout se stlačeným tlačítkem do stav, do kterého přechod povede. Puštěním tlačítka se přechod vytvoří.

 **Odebrat přechod** – klávesa „6“ a „d“

Odebrání přechodu mezi dvěma stavy. S tím souvisí i odebrání všech Beziérových polygonů.

Ovládání: Prosté kliknutí levým tlačítkem myši na přechod, který chceme odebrat.

 **Přidat řídicí bod** – klávesa „7“ a „c“

Přidání řídicího bodu, kterým bude křivka procházet. Automaticky se vytvoří spojitost křivky C1, pokud jsou nadefinované nějaké řídicí body Beziérova polygonu.

Ovládání: Kliknout a držet na přechod, na který chceme řídicí bod přidat, táhnutím se stlačením tlačítkem nasměrujeme na místo, kde má být bod umístěn. Puštěním tlačítka se bod ustálí.

 **Odebrat řídicí bod** – klávesa „8“ a „v“

Odebráním řídicího bodu se přechod nesmaže, pouze se spojí ty části, které byly provázány s mazaným bodem.

Ovládání: Stejně jak u mazání stavů.

 **Přidat řídicí bod Beziérova polygonu** – klávesa „9“ a „b“

Při přidávání se automaticky udělá spojitost C1, pokud jsou na přechodu nějaké řídicí body. Rovnou se křivka vykresluje, takže je názorně vidět, jak bude křivka vypadat.

Ovládání: Stejně jak u přidávání řídicích bodů.

 **Odebrat řídicí bod Beziérova polygonu** – klávesa „0“ a „n“

Stejná funkce jak u odebrání řídicího bodu, jen se maže řídicí bod Beziérova polygonu.

Ovládání: Stejně jak u odebrání řídicího bodu

1.3.1.2 Kontextové menu

Kontextové menu se zobrazuje vždy při stisknutí pravého tlačítka myši. Jaké položky bude obsahovat, je určeno podle toho, který objekt pravým tlačítkem vyberete. Např. pokud není zacílen žádný objekt, obsahuje pouze položku „přidat stav“. Pro úplnost je nutno zdůraznit, že funkce jsou stejné jako při použití nástrojové lišty. Navíc lze z této nabídky editovat i formát objektů, což v nástrojové liště nelze. O formátu objektů bude zmínka níže.

1.3.2 Režimy zobrazení

Jsou dva režimy, ve kterých je možné automat zobrazit:

- Plné zobrazení
- Zobrazení bez pomocných objektů

Implicitní je plné zobrazení, kde jsou vidět všechny objekty (včetně řídicích bodů a Bezierových polygonů). Zobrazení bez pomocných objektů lze nastavit v menu „Zobrazit“ zaškrtnutím položky „Skryt pomocné objekty“. V tomto režimu zobrazení je automat exportován do grafických formátů, což je užitečné například pro tvorbu statických prezentačních materiálů, kde působí pomocné objekty spíše rušivým dojmem.

Je také možné vypnout **označování**. Při „njetí“ na libovolný objekt je objekt automaticky zvýrazněn nastavenou barvou. Tomuto říkáme **označování**. Označování lze vypnout v menu „Zobrazit“ položkou „Vypnout označování“.

1.3.3 Nastavení formátu

Formát objektů je jednou ze základních vlastností vzhledu automatu. Každému objektu se mohou zvlášť nastavit tyto vlastnosti:

- **Styl tužky**
- **Styl tužky při označení**
- **Styl výplně**
- **Styl výplně při označení**
- **Viditelnost**
- **Viditelnost při označení**
- **Priorita zobrazení** – v které vrstvě bude objekt zobrazen

Dále pak u některých objektů:

- **Kvalita** – u křivky počet čar, ze které se skládá
- **Styl šipky** – vyplněná, nevyplněná
- **Měřítko výšky u šipky** – číslo s desetinou čárkou, kterým se vynásobí šířka šipky
- **Měřítko šířky u šipky** – číslo s desetinou čárkou, kterým se vynásobí výška šipky
- **Font** – lze nastavit u popisku či u stavu

Formát lze nastavit buď u každého objektu zvlášť, nebo i globálně pro celý automat. To se nastavuje v menu „Formát“ položkou „Formát objektů“.

U zásobníkového automatu je ještě potřeba nastavit obsah na zásobníku. To se provádí v menu „Formát“ položkou „Formát automatu“.

1.3.4 Nástroje

Aplikace obsahuje nástroje pro simulaci práce automatu. Pro správnou funkčnost simulace je vyžadován deterministický automat. K ověření determinismu automatu slouží nástroj **Verifikace** v menu „Nástroje“. Verifikace projde podmínky determinismu a povolí simulaci (položka menu **Animace**) či nikoliv.

Po provedení verifikace je možné automat animovat. Animátor se spustí v menu „Nástroje“ položkou **Animace**. Před animací je vyžádáno zadání vstupního řetězce. Animace je možná až do rozhodnutí, zda je řetězec přijímán daným automatem nebo ne. Animace probíhá na bázi označování objektů. U zásobníkového automatu se navíc otevře okno s aktuálním obsahem zásobníku.

Dále se dá použít nástroj **Ortho**, který je užitečný při vytváření přechodů, pokud chceme mít ortogonálně orientované hrany.

1.4 Export

Základním formátem pro export modelů je formát **XML**. Strukturu dokumentu, která je v aplikaci použita uvádím v příloze. XML je jediný formát, ze kterého lze n této aplikaci automat i načítat.

Dále může být automat exportován do formátu **TeX**, pomocí prostředí *picture*, které může být snadno vloženo do LaTeX dokumentu. Bohužel tento formát podporuje Beziérova křivky pouze s jedním řídicím bodem a používá jiný souřadný systém, takže výstup bude horizontálně překlopen a pokud v automatu používáme víc řídicích bodů, tak výstup nebude přesný. Aplikace tedy umožňuje jen částečnou podporu exportu do tohoto formátu.

Export do grafických formátů pracuje pouze s částí vymezenou hraničními body automatu. Dále je možné při exportu automat trochu přiblížit a to až o 300% (zvýší se ovšem nepřesnost). Exportuje se automat bez pomocných objektů.

Grafické formáty pro export jsou:

- [BMP](#)
- [PNG](#)
- [JPG](#)
- [XPM](#)

1.5 Implementační poznámky

1.5.1 Implementace označování

Pro optimalizaci rychlosti označování objektů jsou do pomocné zásobníkové datové struktury ukládány všechny viditelné objekty a při provádění označování jsou zpracovávány pouze tyto a nikoli všechny objekty na pracovní ploše.

1.5.2 Vykreslování na plochu

Nepříjemnému blikání při překreslování pracovní plochy jsme se vyhnuli pomocí tzv. Double bufferu. To znamená, že obraz se nekreslí přímo na zobrazovací plochu, ale nejdříve do paměti a poté se celý bitmapa v paměti naráz vykreslí. Původně jsme očekávali od použité grafické knihovny *wxWidgets* zajištění tohoto pokročilejšího způsobu zobrazování, který je při práci s grafikou častý.

1.5.3 Portabilita

Každý operační systém a dokonce i uživatel používá jinou znakovou sadu. Např. v OS Windows to je v českém prostředí nejčastěji Windows-1250 či unicode; v OS Linux je pravidlem iso-8859-2 nebo dokonce unicode. To mělo za důsledek odlišné (špatné) zobrazování diakritiky znamenalo. Problém

pomohla vyřešit ve wxWidgets třída wxLocale, která text na znakovou sadu daného systému převádí automaticky a zároveň zajišťuje podporu více jazyků. Proto je možné používat uživatelské rozhraní v češtině i angličtině, kde problém znakových sad není tak markantní. Binární soubory s překlady řetězců jsou dodávány spolu s binárním souborem programu a konverze kódování je již prováděna za přímo běhu. Pokud soubory jazyka u programu nebudou přiloženy, bude program zobrazen v angličtině. Program byl prvotně testován v multiplatformním prostředí na systémech:

- Windows ve vývojovém prostředí Visual Studio .NET a pomocí kompilátoru GCC
- Linux, v kompilátoru GCC

Pozdější fáze projekty byly prováděny již pouze na OS Windows s využitím profesionálního vývojářského prostředí Visual Studio .NET. Přesto by přenositelnost zdrojových kódů i na další platformy měla být zachována především díky využívání knihovny wxWidgets.