

# Programování pro IDRISI

Stanislav Šumbera

# Idrisi obecně

- napsáno v prostředí Delphi
- „příliš“ modulární řešení
- modul (samostatný proces) představuje nejmenší jednotku pro řízení běhu zpracování obrazu
- rastrový i vektorový formát „rst“, „vct“ je publikovaný
- API (programové rozhraní) je velmi zjednodušené

# Možnosti programování

- pomocí maker – využití Makro modeleru v GUI (grafickém prostředí) Idrisi
- pomocí OLE automation (téměř nic navíc oproti makrům !) – vlastní kód (uživatelský modul) lze jednoduše integrovat do GUI Idrisi
- pomocí přímého zápisu do souboru – Vstupně/Výstupní rutiny si musí programátor napsat sám (není API)

# využití OLE automation

- API poskytuje převážně podporu integrace uživatelských modulů do prostředí Idrisi.
- možno programovat z VB, C++, Python, .NET – tj. jakékoli prostředí podporující standard COM
- není API pro přístup do rastrového souboru !

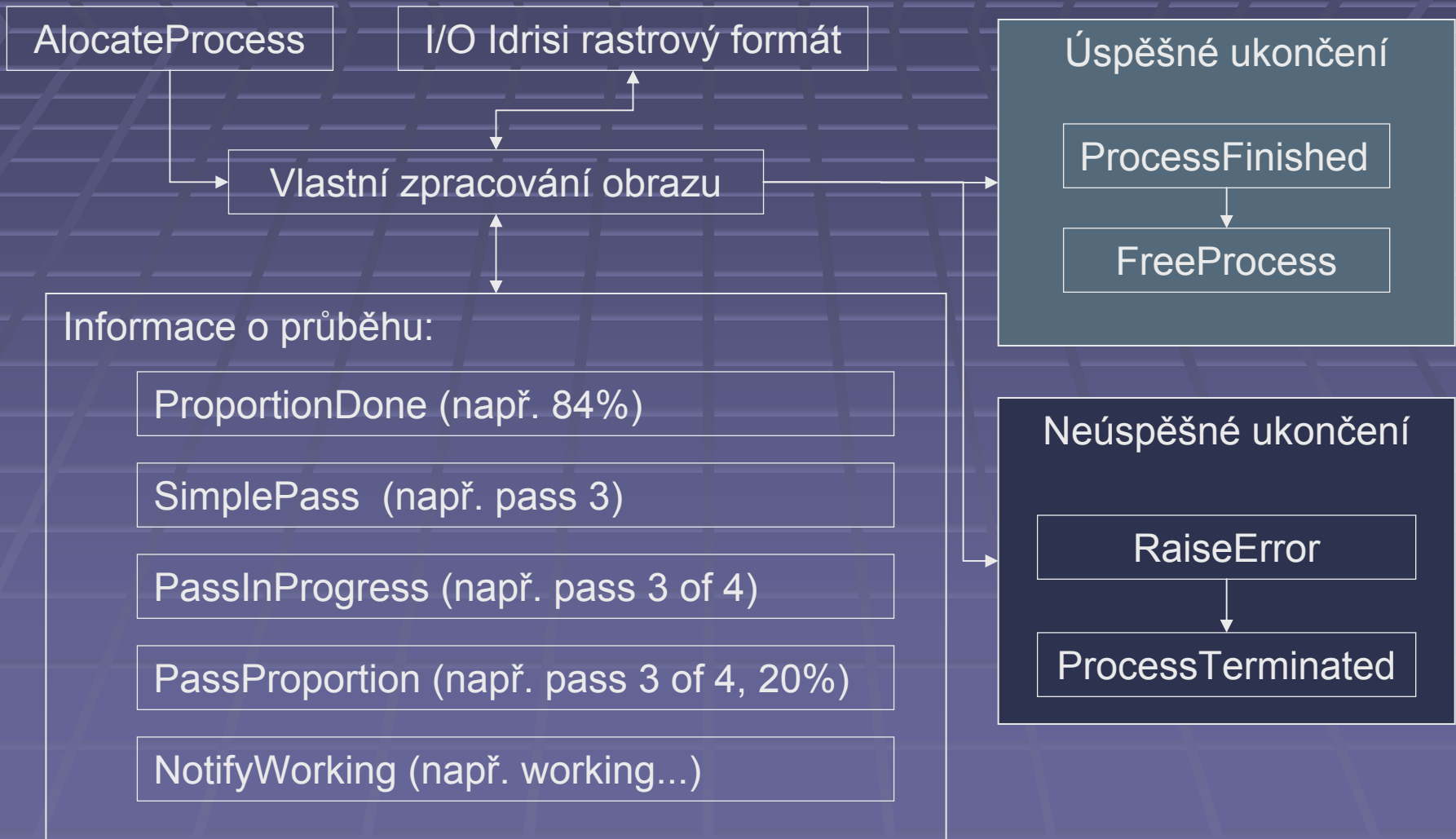
# využití OLE automation práce s projektem a adresáři

- **Get/SetWorkingDir** – práce s projektovým adresářem
- **Get/SetResourceDir** – práce s pomocnými adresáři
- **Get/Set CurrentProjectFileName** – nastavení aktuálního projektu
- **Get\_GeorefDir, Get\_SymbolDir, Get\_CartaLinxPath** – obdržení příslušných specializovaných adresářů

# využití OLE automation práce s moduly

- **Run\_Macro** – spuštění makra
- **Run\_Module** – spuštění modulu
- **Run\_Process** - spuštění modulu a zobrazení výsledku
- **Display\_File** – zobrazení souboru
- **AddLayerToDisplay** – přidání vrstvy do kompozice zobrazení
- **AllocateProcess** – uživatelské zahájení vlastního procesu s možností informovat Idrisi o jeho průběhu
- **ProportionDone** – zpětné volání do Idrisi o průběhu-stavu uživatelského modulu
- **GetProgress** – informace o průběhu daného procesu
- **ProcessFinished, ProcessTerminated** – zaslání statusu ukončení uživatelského procesu

# Průběh uživatelského procesu



# využití OLE automation práce s Idrisi prostředím

- **GetFromDocFile** – obdržení metadat rastrového souboru
- **GetFromVec/ValDocFile** – obdržení metadat vektorového souboru
- **ReadVLXFile** – obdržení informací o připojených databázích
- **CalculateRaster/VecMinMax** – obdržení informace o velikosti rastru, resp. vektoru
- **CalculateResolution** - informace o rozlišení rastru
- **CalculateRG24minmxResolution** - informace o maximálních a minimálních hodnotách RGB složek v rastru



# využití OLE automation ostatní

- `Get/Set_MainWindowState` – nastavení / informace o stavu hlavního okna Idrisi
- `Get/Set_IniString` – nastavení / informace o uživatelsky definovaném parametru v ini souboru
- `Get/Set_Language` – nastavení / informace o jazyku prostředí
- `Get_****Extension` – vrací přípony specifických souborů (např. vst, rdc, vdc, avl, vct, apod.)
- `Get/Set_AutoOutputPrefix` – nastavení / informace o předponě dočasných souborů
- `Get/Set_DefaultQual/QuantPal` – nastavení / informace o specifické paletě
- `Get/Set_useAutoDisplay` – nastavení / informace o automatickém zobrazení výsledku
- `Get/Set_AutoShowLegend/Title` – nastavení / informace o automatickém zobrazení legendy nebo názvu v obrazovém souboru

# přímý zápis do Idrisi souborů - rastry

- přímé (sekvenční) binární 8,16,24-bitové hodnoty (tj. nekomprimované, nestrukturované)
- datové formáty:
  - byte
  - integer – 16 bit/pixel se znaménkem
  - real – IEEE 4 byte
  - RGB 8 – reprezentace kódů barev
  - RGB 24 – „klasický“ R-G-B formát BMP, v Idrisi pouze ke zobrazení
- chybí IEEE 64 bitový formát (podporuje např. tif)
- chybí standardní API pro čtení/zápis rastrů

# přímý zápis do Idrisi souborů - vektory

- hlavička souboru obsahuje typ vektoru:
  - Body = 1;
  - Linie = 2;
  - Polygony = 3;
  - Texty = 4;
- přímé (sekvenční) binární 8,16,32-bitové hodnoty (tj. nekomprimované) v dokumentované struktuře
- datové formáty:
  - byte – 8 bit/pixel
  - integer – 16 bit/pixel
  - longint – 32 bitový se znaménkem
  - single – IEEE 4 byte
  - double – IEEE 8 byte
- chybí standardní API pro čtení/zápis vektorů

# „Služby“ IRC Brno

- IRC Brno nabízí programové knihovny včetně zdrojových kódů v C :
  - pro zápis/čtení rastrového formátu RGB8
  - pro řetězení algoritmů (ER mapper metoda)
  - zhruba 20 algoritmů:
    - vyhlazení lineární/nelineární
    - template matching pomocí křížové korelace
    - vyhledání vrcholů korun stromů
    - detekce hran
    - morfologické operace
    - specializované metody, např. difuze, region growing, výpočet entropie apod.

# Kontakt

[stanislav@sumbera.com](mailto:stanislav@sumbera.com)

<http://www.sumbera.com>