

АРХІТЕКТУРА СПЕЦІАЛІЗОВАНОГО ПРОГРАМНОГО МОДУЛЯ ГІС: ПОБУДОВА КАРТ ВЕКТОРНОГО ФОРМАТУ

© Любінський Б.Б., Пеняк І.О., 2014

Наведено підхід до побудови карт векторного формату. Сформовано функціональні вимоги і побудовано алгоритми масштабування і позиціонування графічних примітивів і оцифровування растрових карт у векторні. Спроектовано архітектуру модуля побудови карт векторного формату спеціалізованої геоінформаційної системи.

Ключові слова: програмне забезпечення, геоінформаційна технологія, векторний формат, архітектура, інвентаризація, просторовий аналіз.

An approach for the vector format maps creation is presented. The relevant graphics primitives scaling-and-positioning algorithms were developed along with the functional requirements forming. The algorithms that digitize raster maps into the vector ones are described as well. The entire architecture of module for the creation of the vector format maps, intended to the specialized geographic information system is designed.

Key words: software, geoinformation technology, vector format, architecture, inventory, spatial analysis.

Вступ. Постановка завдання та його актуальність

У сучасних геоінформаційних системах є можливість відображати картографічну інформацію з подальшим аналізом. Спеціалізована геоінформаційна система передбачає не тільки побудову модулів відображення, аналізу, збереження, але й створення модуля побудови цифрових карт. У комп'ютерних технологіях підготовки до створення карт використовуються векторні цифрові картографічні дані. Для оцифровування карт, тобто створення з растру векторної карти з подальшим присвоєнням основної атрибутивної інформації та класифікації картографічних даних, необхідно використовувати спеціалізоване програмне забезпечення [5,6]. Для проведення просторової інвентаризації парникових газів вхідними даними є векторні карти, статистична інформація, а також різноманітні коефіцієнти та параметри. Статистичну інформацію про результати господарської діяльності у різних секторах можна безкоштовно отримати зі статистичних управлінь. Побудова алгоритму і проектування модуля побудови карт векторного формату є актуальною задачею, оскільки цифрові карти аналізованої місцевості не є безкоштовні.

Аналіз останніх досліджень

Значний внесок у побудову геоінформаційних технологій просторової інвентаризації парникових газів зробили Р.А. Бунь, Х.В. Гамаль та М.Ю. Лесів. В роботах цих авторів використовуються ліцензійні карти векторних форматів для територіального аналізу процесів емісій у різних секторах господарської діяльності [1–3].

Для великих територій просторова інвентаризація парникових газів не проводиться. Натомість здійснюють традиційну інвентаризацію для країни загалом, для чого використовують методики IPCC [4] та рекомендоване програмне забезпечення – Excel.

Мета та формування цілей статті

Мета роботи – побудова архітектури модуля спеціалізованого програмного забезпечення для побудови векторних карт на основі фізичних карт з подальшою прив'язкою атрибутивної інформації. Такі цифрові карти векторного формату використовуються як джерело вхідних даних

для аналізу та просторової інвентаризації парникових газів на регіональному рівні, а також для візуалізації результатів такої інвентаризації.

Для досягнення цієї мети було поставлено такі завдання:

- сформулювати основні вимоги до модуля перетворення растрових карт на векторний формат;
- побудувати алгоритм оцифрування растрових карт у векторні;
- спроектувати гнучку архітектуру модуля перетворення растрових форматів на векторні з подальшою можливістю присвоєння атрибутивної інформації;
- побудувати алгоритм масштабування і позиціонування графічних примітивів карти;
- описати роботу модуля побудови карт векторного формату в спеціалізованому програмному забезпеченні.

Побудова графічного модуля

Для створення власних цифрових карт слід дотримуватись такого алгоритму:

- 1) використати фізичну карту аналізованої місцевості;
- 2) оцифрувати карту з метою отримання карти растрового формату;
- 3) відкрити растрову карту, встановивши відповідний процент прозорості для обведення контурів оцифрованої карти;
- 4) створити новий шар векторної карти, використавши модуль побудови векторних карт спеціалізованого програмного забезпечення із встановленням граничних точок карти і по контурах растрової карти побудувати її векторний аналог.

У модулі побудови векторних карт реалізовано такі функціональні можливості:

- 1) побудова здійснюється у поточному або у новому шарі (layer);
- 2) якщо документ багатшаровий, то побудова відбувається на верхньому активному шарі (активним вважається видимий шар);
- 3) підтримка побудови таких об'єктів карти:
 - лінія, полілінія;
 - полігон;
 - точка;
 - прямокутник;
 - еліпс;
 - текст;
- 4) можливість встановлювати:
 - заливку для об'єктів типу полігон, еліпс чи прямокутник;
 - товщину і шаблон для контуру об'єктів;
 - кольори для заливки і контуру;
- 5) можливість відмінити опцію рисування об'єктів під час процесу створення карти;
- 6) видалення об'єктів;
- 7) виділення об'єктів в області для малювання і синхронізація з базою даних та навпаки;
- 8) створення карти відбувається в режимі редагування;
- 9) розмір документа змінюється автоматично залежно від кількості доданих шарів.

Загальну структуру модуля створення карт представлено на рис. 1. В одному вікні програми можна відкрити декілька документів одночасно. Для проведення аналізу в документі можна відкривати декілька шарів, які містять необхідну геоінформацію, накладаючи їх один на одного. Шар карти складається з графічних примітивів, які позначають дорогу, населений пункт тощо. Функціональність документа реалізовано в класі *GraphicDocument*. Документ сам по собі не може бути збережений, оскільки векторні формати підтримують тільки рівень шарів. Масштабування документа реалізовано в класі *GraphicDocument* властивістю *CurrentScale*. Шар карти є контейнером об'єктів. Накладання декількох шарів на одну карту дає можливість групувати певні об'єкти. Наприклад, карту України можна представити декількома різними шарами:

- географічна: рельєф;
- географічна: річки;

- адміністративна;
- карта доріг.



Рис. 1. Структура карти

За потреби можна використовувати комбінації цих карт, наприклад, для побудови карти мостів можна накласти географічну карту з річками і карту доріг. Основна роль шару – бути контейнером для об'єктів карти. Шар приймає координати розташованих у ньому об'єктів. Функціональність шару реалізовано у класах *BaseGraphicLayer* і *GraphicLayer*. На карті відображаються графічні примітиви, які реалізовані в класі *GraphicFeature* і похідні від нього : *MyText*, *MyPolilyne*, *MyRegion*, *MyBasePoint*. Основні функції, які реалізує клас *GraphicFeature*:

- 1) правильне позиціонування графічного об'єкта в межах шару;
- 2) визначення локальних координат і розмірів графічного об'єкта на основі глобальних;
- 3) масштабування графічного об'єкта;
- 4) відображення;
- 5) трансформація;
- 6) визначення площі і периметра графічного об'єкта.

Використовуючи бібліотеки *GDAL/OGR* [7], зчитують загальну інформацію про шар карти, а також інформацію про налаштування і координати для кожного графічного об'єкта і поміщають її в об'єкт класу *Layer*. У класі *VectorLayerToGraphicLayerAdapter* з допомогою методу *CreateGraphicLayer* класу *BaseGraphicLayer* перетворюють цю інформацію на графічний шар. На графічному шарі рисують нові об'єкти. *GraphicLayer* клас є похідним від *BaseGraphicLayer*, а *BaseGraphicLayer* є похідним від *System.Windows.Controls.Canvas*. Алгоритм побудови зображення реалізовано методом *Redraw* класу *BaseGraphicLayer*, який викликає *BitmapSource :: Redraw* для кожного графічного об'єкта. Для відображення графічного примітиву створюється новий шар (*GraphicLayer*). Клас *BaseGraphicLayer* містить змінну *VisualList*, яка є контейнером графічних об'єктів. Нарисований графічний примітив додається і зберігається в *VisualList*. Збереження шару реалізовано в класі *BaseGraphicLayer*, який виконує перетворення шару класом *GraphicLayerToVectorLayerAdapter*. На рис. 2 зображено діаграму компонентів модуля.

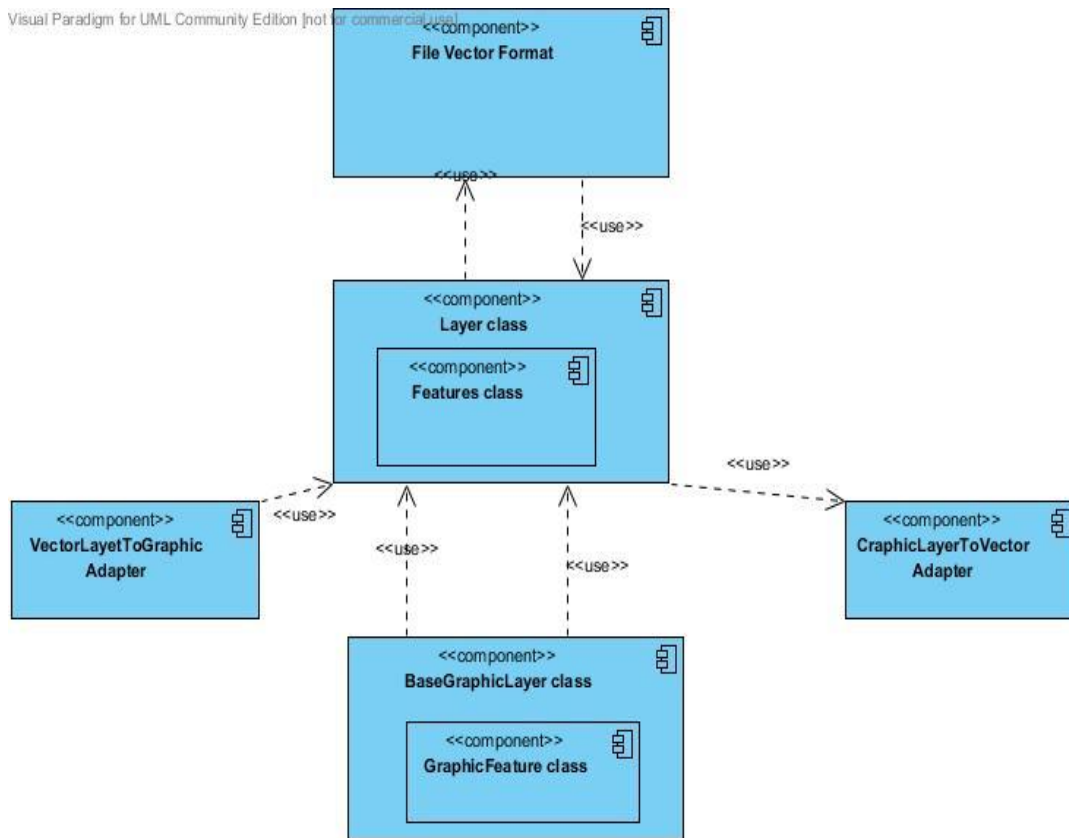


Рис.2. Діаграма компонентів модуля

На рис. 3 проілюстровано спрощену діаграму класів цього модуля побудови карт.

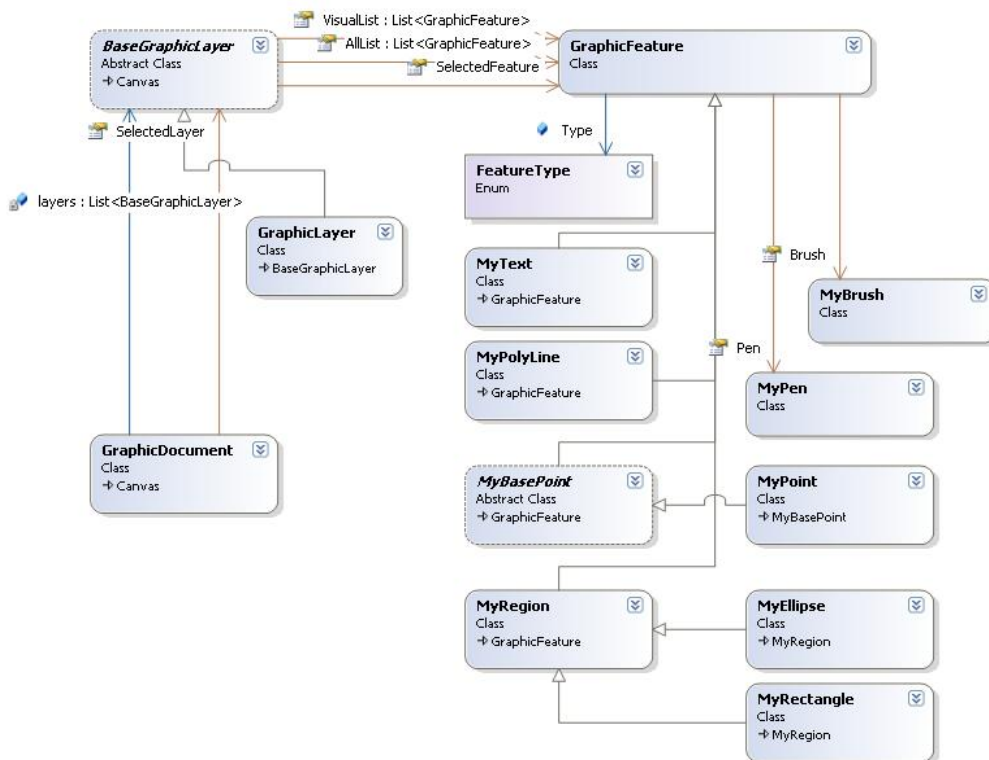


Рис. 3. Спрощена діаграма класів модуля побудови карт векторного формату

Програмне забезпечення модуля побудови карт векторного формату

Розглянемо роботу модуля побудови карт векторного формату, графічний інтерфейс якого подано на рис. 4.

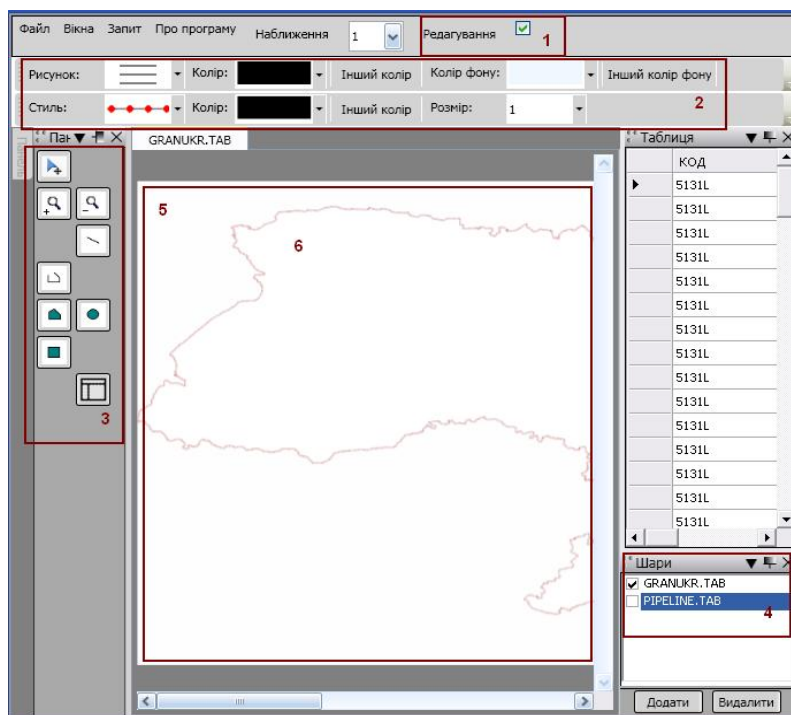


Рис. 4. Графічний інтерфейс модуля

На рис. 4 виділено такі елементи:

1) прапорець “Редагування” (побудову нових об’єктів дозволено лише тоді, коли прапорець “Редагування” ввімкнено; функціональність переведення документа в режим редагування і навпаки виконується з використанням методів *MakeEditable* і *MakeNoEditable* класу *GraphicLayer*);

2) панель інструментів (використовується для вибору типу фігури графічного об’єкта – лінія, полілінія, полігон, еліпс і прямокутник тощо, також є інструменти для зміни масштабу і накладання сітки);

3) панель налаштувань (ця панель змінюється на основі вибраного елемента в панелі інструментів; якщо активним є полігон, еліпс чи прямокутник, то в панелі налаштувань будуть доступні налаштування для заливки і контуру фігури; якщо активними є лінія чи полілінія, то тільки налаштування контуру);

4) панель шарів (клас *GraphicDocument* містить поле *layers* типу *List<BaseGraphicLayer>*, в якому зберігається список усіх шарів; наявні такі можливості: додавання/видалення шару – методи *Add* та *Remove* класу *GraphicDocument*, встановлення шару видимим/невидимим, а також встановлення почерговості шарів – методи *ShiftUp* та *ShiftDown* класу *GraphicDocument*; побудова може відбуватися тільки на верхньому видимому шарі; при створенні нового графічного шару, задаються координати границь, тоді всі наступні об’єкти матимуть координати в межах цього шару);

5) область документа і область шару – клас *GraphicLayer* (у деяких випадках вони збігаються, наприклад, коли відкритий один шар в документі або коли координати шарів збігаються);

6) графічний об’єкт (графічний об’єкт рисують за допомогою мишки; залежно від того, що вибрано в панелях інструментів і налаштувань, створюється об’єкт певного типу і додається до шару).

Масштабування і позиціонування

Векторні формати не містять інформації про розміри і позицію шарів. Розміри документа і позиція шарів формуються на основі координат графічних об’єктів.

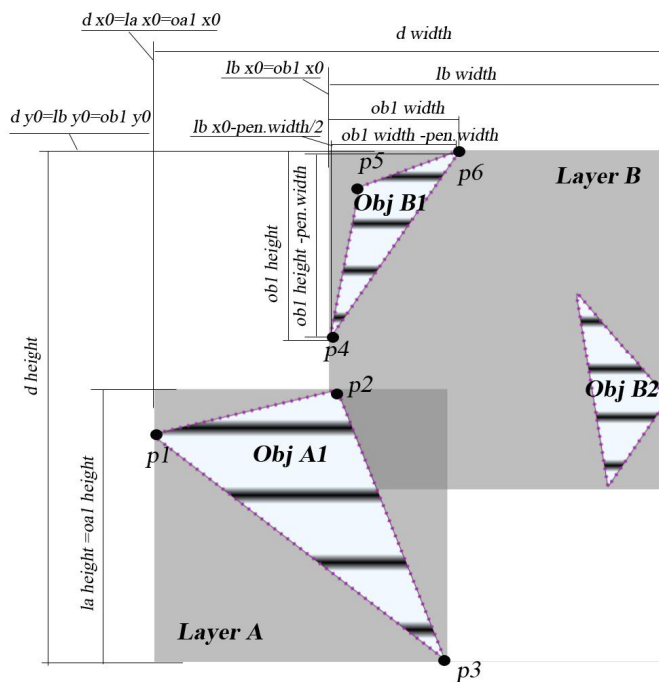


Рис. 5. Позиціонування документа, шарів і графічних об'єктів

На рис. 5 зображено документ, який містить два шари: *Layer A*, *Layer B*. Координати цих шарів не збігаються. *ObjA1*, *ObjB1*, *ObjB2* – це графічні об'єкти, екземпляри класу *GraphicFeature*. Їхні реальні координати містяться в полі *GlobalPoints*. При створенні документа на основі цих точок розраховують реальні відстані між крайніми горизонтальними і вертикальними точками, на основі чого обчислюють масштаб (*CurrentScale*), який буде однаковий: як для документа і шару, так і для графічних примітивів. На основі масштабу глобальні координати перетворюються на локальні. Товщина контуру не масштабується.

Для прикладу обчислимо локальні координати для двох шарів і виконаємо масштабування. Нехай створено два шари з такими координатами:

LayerB з координатами $\{20; 21; -52; -51\}$ та

LayerA з координатами $\{19,5; 20,5; -51,5; -50,5\}$,

на яких відображено декілька графічних об'єктів, координати яких знаходяться в межах координат шарів.

Глобальними координатами (*GlobalPoints*) для об'єкта *A1* є:

$p1 = \{19.511; 51.164277\};$

$p2 = \{20.037; 51.290277\};$

$p3 = \{20.347; 50.514277\};$

$p4 = \{19.511; 51.156277\}.$

Точки розміщено в прямокутнику $GlobalPointsRect = \{19.511; 50.514277; 0.836; 0.776\}$. Два інші об'єкти описують такі прямокутники:

$GlobalPointsRect = \{20.017, 51.444277, 0.374, 0.542\};$

$GlobalPointsRect = \{20.729, 51.012277, 0.262, 0.558\}.$

У процесі накладання отримуємо такі параметри:

$minGlobalX = 19,511;$

$maxGlobalX = 20,991;$

$GlobalWidth = 1,48;$

$minGlobalY = 50,514;$

$maxGlobalY = 51,99;$

$GlobalHeight = 1,48.$

Область для відображення (*Document.Size*) у межах $[-761.195; 757.113]$. Значення *CurrentScale* обчислюється за формулою: $(dWidthnoal.Pen / 2nob2.Pen / 2) / GlobalWidth$

Точка *p1* – це крайня точка в документі по осі *x*, відповідно зміщення всього документа по цій осі дорівнює 19,511. Точка *p4* – це крайня точка в документі по осі *y*, відповідно зміщення всього документа по цій осі дорівнює 51,99. Для спрощення обчислень товщину контура не враховують. Локальні координати графічних об'єктів описують зміщенням відносно властивості *PointsRect*.

На рис. 5 зображено, що зміщення шару *A* відносно початку документа по осі *x* дорівнює нулю. Шар *B* має зміщення по осі *x*, а по осі *y* зміщення дорівнює нулю.

Приймаємо точку $\{dx; dy\}$ документа за нульову точку і розраховуємо розташування усіх об'єктів відносно цієї точки. Надалі розрахунки проводяться тільки для осі *x*.

Точки шару *A* обчислюємо за такими формулами:

$$\begin{aligned} p1.x &= (GlobalPoints[0].x-dx) \cdot CurrentScale ; \\ p2.x &= (GlobalPoints[1].x-dx) \cdot CurrentScale -lb.x0 ; \\ p3.x &= (GlobalPoints[2].x-dx) \cdot CurrentScale -lb.x0 . \end{aligned}$$

Для шару *B* крайньою точкою по осі *x* є точка *p4*. Вона визначає зміщення шару *B* відносно документа по осі *x*:

$$\begin{aligned} p4.x &= (GlobalPoints[0].x-dx) \cdot CurrentScale ; \\ p5.x &= (GlobalPoints[1].x-dx) \cdot CurrentScale -lb.x0 ; \\ p6.x &= (GlobalPoints[2].x-dx) \cdot CurrentScale -lb.x0 . \end{aligned}$$

Ширину і висоту об'єктів і шару обчислюємо за формулами:

$$\begin{aligned} ObjA1.Width &= GlobalPointsRect.Width \cdot CurrentScale ; \\ ObjA1.Heigth &= GlobalPointsRect.Heigth \cdot CurrentScale ; \\ ObjB1.Width &= GlobalPointsRect.Width \cdot CurrentScale ; \\ ObjB1.Heigth &= GlobalPointsRect.Width \cdot CurrentScale . \end{aligned}$$

Висновки

Сформовано основні вимоги до модуля перетворення растрових карт на векторі. Побудовано алгоритми масштабування і позиціонування графічних примітивів і оцифрування растрових карт у векторні. Спроектовано архітектуру модуля побудови карт векторного формату спеціалізованої геоінформаційної системи для просторового аналізу та інвентаризації парникових газів. Описано роботу модуля побудови карт векторного формату в спеціалізованому програмному забезпеченні.

1. Гамаль Х. В. Геоінформаційна технологія для просторового аналізу парникових газів в енергетичному секторі : дис. ... канд. техн. наук : 05.13.06 / Христина Володимирівна Гамаль. – Львів : Національний університет “Львівська політехніка”, 2009. – 246 с. 2. Інформаційні технології формування кадастру емісій парникових газів Львівщини / Р. А. Бунь, Н. О. Шпак, Б. М. Матоліч та ін.; За ред. Р. А. Буня та Н. О. Шпака. – Львів : Видавничий дім “Укрпол”, 2010. – 272 с. 3. Лесів М. Ю. Математичне моделювання та просторовий аналіз емісії парникових газів у прикордонних регіонах України : дис. ... канд. техн. наук : 05.13.06 / Мирослава Юріївна Лесів. – Львів : Національний університет “Львівська політехніка”, 2011. – 195 с. 4. 1996 IPCC Guidelines for National Greenhouse Gas Inventories. [Електронний ресурс] – Режим доступу – <http://www.ipcc-nggip.iges.or.jp/public/gl/guidelin/ch2wb3.pdf>. 5. Любінський Б. Б. Спеціалізоване програмне забезпечення для географічного аналізу та інвентаризації парникових газів / Б. Б. Любінський, Р. А. Бунь // Моделювання та інформаційні технології. – 2011. – № 59. – С. 129-135. 6. Любінський Б. Б. Архітектура спеціалізованих програмних модулів для географічного аналізу об'єктів при інвентаризації парникових газів / Б. Б. Любінський, Р. А. Бунь // Штучний інтелект. – Донецьк, 2011. – № 4. – С. 303-309. 7. GDAL: Geospatial Data Abstraction Library. Available at: <http://www.gdal.org/>.