

УДК 004.021:004.92

### **Бородавка Євгеній Володимирович**

Кандидат технічних наук, доцент, доцент кафедри інформаційних технологій проектування та прикладної математики, *orcid.org/0000-0002-7476-9387*

*Київський національний університет будівництва і архітектури, Київ*

### **Квасневський Владислав Михайлович**

Спеціаліст відділу розробок

*ТОВ AllbauSoftware, Київ*

## **МЕТОДИ ПОБУДОВИ ОБ'ЄКТІВ У КОМП'ЮТЕРНІЙ ГРАФІЦІ**

*Анотація.* В даній роботі розглянуто основні методи побудови графічних об'єктів, що найчастіше застосовуються в системах автоматизованого проектування, на основі життєвого циклу будівельного об'єкта. Розглянуті методи проаналізовано на предмет їх переваг і недоліків, а також наведено принципи зберігання даних, що використовуються в кожному з методів. У статті окремо розглянуто двовимірні та тривимірні об'єкти, оскільки методи побудови для них доволі різняться і не завжди можуть бути розширені до тривимірного простору (для двовимірних об'єктів) або урізані до двовимірного простору (для тривимірних об'єктів). Запропоновані в статті методи можуть використовуватися як окремо, так і в комплексі, залежно від поставлених задач.

**Ключові слова:** графічний об'єкт; примітив; полігональна модель; твердотільна геометрія; триангуляція

### **Вступ**

Під час розробки будь-якої системи автоматизованого проектування або системи забезпечення автоматизації всього життєвого циклу будівельного об'єкта перед розробниками виникає проблема адекватного відтворення елементів будівлі на екрані комп'ютера. Причому в різних ситуаціях необхідно відтворювати як реальний вигляд об'єктів, так і їх плоске зображення в різних проекціях.

Для побудови будь-яких об'єктів в комп'ютерній графіці є значна кількість методів, що базуються на використанні різних графічних примітивів. В даній статті будуть розглянуті лише ті з них, що забезпечують максимальну простоту, універсальність і швидкість побудови.

### **Аналіз останніх досліджень і публікацій**

У роботі [1] проведено детальний аналіз та дослідження основних алгоритмів триангуляції, наведено їх переваги і недоліки, здійснено підрахунок швидкодії різних алгоритмів з використанням різних структур даних.

У роботі [2] було виявлено основні графічні примітиви, що використовуються для опису будівельних елементів та подана структура базового елемента.

У роботах [3; 4] подано концепцію моделі будівельного об'єкта та структури даних, що необхідні для її зберігання. На основі цих даних потрібно визначитися з методами побудови графічної інтерпретації розглянутих будівельних об'єктів та їх елементів.

### **Мета статті**

Метою даної роботи є аналіз та дослідження наявних методів побудови об'єктів у комп'ютерній графіці та визначення, які з них найбільше підходять для практичного використання в системах автоматизованого проектування на основі життєвого циклу будівельних об'єктів.

### **Виклад основного матеріалу**

#### **Методи побудови двовимірних об'єктів**

Для побудови підсистем двовимірної комп'ютерної графіки використовують сукупність методів, що дозволяють здійснювати перетворення систем плоских координат, а також методи моделювання різних плоских об'єктів: відрізків, ламаних, багатокутників, кіл, фракталів та кривих. Використання сукупності цих методів дозволяє відтворювати двовимірні об'єкти будь-якої складності. Але для уніфікованої обробки будь-яких двовимірних об'єктів необхідно стандартизувати їх опис. Тобто не використовувати велику кількість різноманітних графічних примітивів, а максимально скоротити їх типи, що використовуються у процесі проектування.

В якості основних графічних примітивів для уніфікованої двовимірної графіки пропонується використовувати такі примітиви: відрізок, прямокутник, ламана, коло та багатокутник. За допомогою цих графічних примітивів можна відобразити будь-які двовимірні об'єкти.

З точки зору реалізації це мають бути класи, що успадковуються від базового абстрактного класу «фігура» (рис. 1).

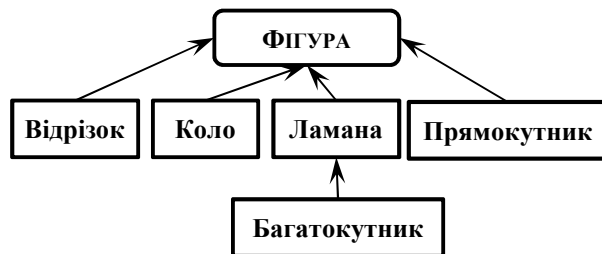


Рисунок 1 – Узагальнена діаграма класів для 2D примітивів

У процесі реалізації подана діаграма класів може доповнюватися іншими класами, що можуть бути необхідні для розширення функціональності кінцевого продукту. Наприклад, для впровадження підтримки редагування об'єктів можуть знадобитися додаткові класи, що призначені саме для цього.

Використання B-сплайнів, кривих Без'є та NURBS-кривих не є обов'язковим. З одного боку вони надають потужні механізми для максимально гладкого відтворення будь-яких криволінійних об'єктів, але з іншого боку потребують більше ресурсів для розрахунків, зберігання і побудови.

### Методи побудови тривимірних об'єктів

Основними методами побудови тривимірних об'єктів є оболонки і конструктивна твердотільна геометрія (*Constructive Solid Geometry, CSG*). Оболонки можуть бути побудовані на основі параметричних поверхонь (білінійна, поверхня Без'є, B-сплайн поверхня, NURBS-поверхня) або на основі плоскогранних (*faced*) поверхонь, найпростішими з яких є граничні (*boundaryrepresentation*) моделі.

Твердотільна геометрія застосовується для об'єктів, які можна описати аналітично, тому сфера її застосування досить обмежена. До того ж сама реалізація такого підходу досить складна.

Найбільш уніфікованою виглядає гранична модель, в якій застосовуються плоскі просторові багатокутники або, що навіть краще, трикутники. Граничні моделі мають чотири основні способи їх подання: явне, список вершин, список ребер та «крилате» подання (*winged-edge representation*). Для прикладу розглянемо граничну модель

прямокутного паралелепіпеда, що є найбільш вживаним тривимірним тілом під час моделювання будівельних об'єктів (рис. 2).

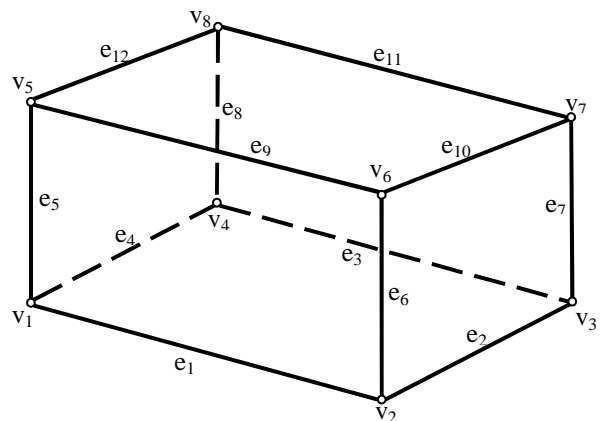


Рисунок 2 – Полігональна модель паралелепіпеда

В явному поданні об'єкт складається з набору граней, кожна з яких є полігоном, що складається з послідовності координат вершин (табл. 1).

Таблиця 1 – Приклад явного подання

ГРАНІ	КООРДИНАТИ
f <sub>1</sub>	x <sub>1</sub> y <sub>1</sub> z <sub>1</sub> , x <sub>2</sub> y <sub>2</sub> z <sub>2</sub> , x <sub>6</sub> y <sub>6</sub> z <sub>6</sub> , x <sub>5</sub> y <sub>5</sub> z <sub>5</sub>
f <sub>2</sub>	x <sub>2</sub> y <sub>2</sub> z <sub>2</sub> , x <sub>3</sub> y <sub>3</sub> z <sub>3</sub> , x <sub>7</sub> y <sub>7</sub> z <sub>7</sub> , x <sub>6</sub> y <sub>6</sub> z <sub>6</sub>

Недоліки такого подання в тому, що по-перше, співвідношення граней задані неявно, а по-друге, координати кожної вершини з'являються стільки разів, скільки граней мають цю вершину.

Щоб обійти повторюваність координат вершин, можна виокремити їх в самостійну структуру. В такому випадку з гранями асоціюються не координати вершин, як в попередньому випадку, а їх індекси в масиві координат вершин (табл. 2).

Таблиця 2 – Приклад подання у вигляді списку вершин

ВЕРШИНИ	КООРДИНАТИ	ГРАНІ	ВЕРШИНИ
v <sub>1</sub>	x <sub>1</sub> y <sub>1</sub> z <sub>1</sub>	f <sub>1</sub>	v <sub>1</sub> v <sub>2</sub> v <sub>3</sub> v <sub>4</sub>
v <sub>2</sub>	x <sub>2</sub> y <sub>2</sub> z <sub>2</sub>	f <sub>2</sub>	v <sub>6</sub> v <sub>2</sub> v <sub>1</sub> v <sub>5</sub>
v <sub>3</sub>	x <sub>3</sub> y <sub>3</sub> z <sub>3</sub>	f <sub>3</sub>	v <sub>7</sub> v <sub>3</sub> v <sub>2</sub> v <sub>6</sub>
v <sub>4</sub>	x <sub>4</sub> y <sub>4</sub> z <sub>4</sub>	f <sub>4</sub>	v <sub>8</sub> v <sub>4</sub> v <sub>3</sub> v <sub>7</sub>
v <sub>5</sub>	x <sub>5</sub> y <sub>5</sub> z <sub>5</sub>	f <sub>5</sub>	v <sub>5</sub> v <sub>1</sub> v <sub>4</sub> v <sub>8</sub>
v <sub>6</sub>	x <sub>6</sub> y <sub>6</sub> z <sub>6</sub>	f <sub>6</sub>	v <sub>8</sub> v <sub>7</sub> v <sub>6</sub> v <sub>5</sub>
v <sub>7</sub>	x <sub>7</sub> y <sub>7</sub> z <sub>7</sub>		
v <sub>8</sub>	x <sub>8</sub> y <sub>8</sub> z <sub>8</sub>		

Відмітимо, що список вершин впорядкований за годинниковою стрілкою, якщо дивитися ззовні паралелепіпеда. Таке подання корисно в багатьох алгоритмах, таких як видалення невидимих поверхонь і розрахунок освітленості полігональних

граней. Однак в такому поданні залишається багато недоліків явного. Наприклад, задача пошуку ребер інцидентного задання вершини все одно вимагає повного перебору.

У моделі «список ребер» грань подається набором ребер і вершини грані визначаються через ребра (табл. 3).

Таблиця 3 – Приклад подання у вигляді списку ребер

РЕБРА	ВЕРШ.	ВЕРШ.	КООРД.	ГРАНІ	РЕБРА
$e_1$	$v_1v_2$	$v_1$	$x_1y_1z_1$	$f_1$	$e_1e_2e_3e_4$
$e_2$	$v_2v_3$	$v_2$	$x_2y_2z_2$	$f_2$	$e_9e_6e_1e_5$
$e_3$	$v_3v_4$	$v_3$	$x_3y_3z_3$	$f_3$	$e_{10}e_7e_2e_6$
$e_4$	$v_4v_1$	$v_4$	$x_4y_4z_4$	$f_4$	$e_{11}e_8e_7e_3$
$e_5$	$v_1v_5$	$v_5$	$x_5y_5z_5$	$f_5$	$e_{12}e_5e_4e_8$
$e_6$	$v_2v_6$	$v_6$	$x_6y_6z_6$	$f_6$	$e_{12}e_{11}e_{10}e_9$
$e_7$	$v_3v_7$	$v_7$	$x_7y_7z_7$		
$e_8$	$v_4v_8$	$v_8$	$x_8y_8z_8$		
$e_9$	$v_5v_6$				
$e_{10}$	$v_6v_7$				
$e_{11}$	$v_7v_8$				
$e_{12}$	$v_8v_5$				

Таким чином, для кожного ребра задається напрямок. Наприклад, ребро  $e_1$  направлено (має позитивний напрямок) від точки  $v_1$  до точки  $v_2$ . Грані також орієнтовані, тобто ребра задані за годинниковою стрілкою, якщо дивитися на паралелепіпед ззовні.

Модель «крилате» подання, розширює список ребер шляхом додавання інформації про взаємне розташування граней (рис. 3).

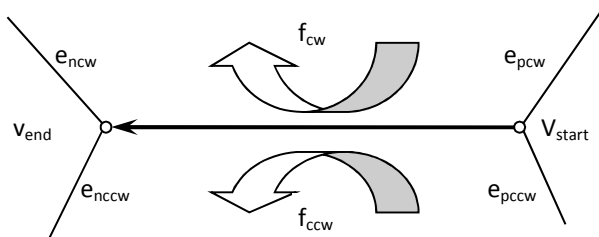


Рисунок 3 – «Крилате» подання

Оскільки кожне ребро з'являється точно в двох гранях, то рівно два ребра з'являються після нього в цих гранях. До того ж один раз ребро з'являється в позитивній орієнтації, а один раз – в негативній.

В «крилатому» поданні використовується асоціація ребра з наступними двома ребрами в гранях. Вони позначаються як  $ncw$  (*nextclockwise*) і  $pccw$  (*nextcounterclockwise*). В даному випадку  $ncw$  означає наступне ребро, що знаходиться в тій грані, де дане ребро з'являється в позитивному напрямку, а  $pccw$  – наступне ребро в іншій грані. Таким чином, починаючи з ребра, що прямо зв'язане з гранню,

можна отримати всі інші інцидентні даній грані ребра, слідуючи за посиланнями  $ncw$  і  $pccw$ . В найбільш загальному випадку в структуру включають також посилання на попередні ребра в сусідніх гранях.

Який тип подання обрати залежить від конкретної реалізації.

Іншим доволі швидким і уніфікованим методом подання просторових об'єктів є триангуляційна модель. *Триангуляція* – планарний граф, всі внутрішні області якого є трикутниками. Будь-який плоский просторовий багатокутник можна звести до набору трикутників. Трикутники в свою чергу мають значну перевагу в швидкості відображення, оскільки всі сучасні графічні адаптери вміють оптимізувати рендеринг трикутників на апаратному рівні, що значно прискорює їх відображення. До того ж триангуляційна модель значно спрощує побудову багатокутників з отворами, оскільки отвори фізично не заповнюються трикутниками після триангуляції таких полігонів. Існує досить багато алгоритмів триангуляції, але всі вони намагаються побудувати трикутники, що в сукупності наближаються до триангуляції Делоне.

Триангуляція Делоне – це випукла триангуляція, що задовольняє умові Делоне, сформульованій в теоремі про порожнисту кулю [1] (рис. 4).

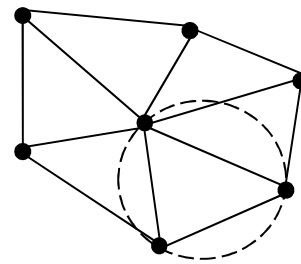


Рисунок 4 – Триангуляція Делоне

Триангуляція Делоне добре збалансована – її трикутники спрямовуються до рівносторонніх, а система трикутників завжди має опуклу границю.

## Висновки

Розглянуті в статті методи побудови двовимірних та тривимірних об'єктів є універсальними і дозволяють змоделювати будь-який об'єкт на комп'ютері за допомогою комп'ютерної графіки. Звичайно якість моделей таких об'єктів буде не найкращою у порівнянні з іншими методами, але в даному випадку розглядається використання моделей об'єктів для систем автоматизації на основі життєвого циклу будівельних об'єктів, де універсальність є пріоритетною.

**Список літератури**

1. Скворцов А.В. Триангуляция Делоне и ее применение. — Томск: Издательство Томского ун-та, 2002. — 128 с.
2. Бородавка Є.В. «Ядро» моделі будівельного об'єкта на основі базового набору графічних примітивів та атрибутів / Є.В. Бородавка // Управління розвитком складних систем. — 2013. № 15. — С. 111 – 114.
3. Бородавка Є.В. Способи подання моделі будівельного об'єкта / Є.В. Бородавка // Управління розвитком складних систем. — 2011 — №8. — С. 101-107.
4. Бородавка Є.В. Методи сортування геометричних об'єктів та їх реалізація на прикладі плагіна автонумерації для САПР Allplan / Є.В. Бородавка, В.М. Квасневський // Управління розвитком складних систем. — 2014. — № 18. — С. 128 – 132.
5. Квасневський В.М. Геометричні методи побудови отворів та гільз для інженерних мереж в САПР Allplan / В.М. Квасневський, Є.В. Бородавка // Управління розвитком складних систем. — 2015. № 22. — С. 128 – 133.
6. Borodavka Y.V. Building model conception / Y.V. Borodavka, M.I. Tsiutsira // In Proceedings of the WORLD Science Conference “Science and Education – Our Future”. — 2014. — Issue №3, November 24-26, 20014, Abu Dhabi, UAE. — pp. 53 – 55.
7. Бородавка Є.В. Автоматизовані засоби підтримки життєвого циклу будівельного об'єкта / Є.В. Бородавка // Матеріали семінару міжнародної науково-практичної конференції «САПР Allplan. Інноваційне проектування в архітектурі і будівництві» — Київ, КНУБА, 2014. — С. 10-13.
8. Малюх В. Н. Введення в сучасні САПР: Курс лекцій. — М.: ДМК Пресс, 2010. — 192 с. — ISBN 978-5-94074-551-8
9. Ніклаус Вірт "Алгоритми та структури даних". — М.: Світ, 1989. — ISBN 5-03-001045-9
10. Бородавка Є.В. Модель розширюваної системи автоматизації життєвого циклу будівельного об'єкта / Є.В. Бородавка // Управління розвитком складних систем. — 2010. — №4. — С. 69-71.

Стаття надійшла до редколегії 27.10.2015

**Рецензент:** д-р техн. наук, проф. С.В. Цюцюра, Київський національний університет будівництва і архітектури, Київ.

**Бородавка Евгений Владимирович**

Кандидат технических наук, доцент, доцент кафедры информационных технологий проектирования и прикладной математики, [orcid.org/0000-0002-7476-9387](http://orcid.org/0000-0002-7476-9387)

Киевский национальный университет строительства и архитектуры, Киев

**Квасневский Владислав Михайлович**

Специалист отдела разработок

ООО Allbau Software, Киев

**МЕТОДЫ ПОСТРОЕНИЯ ОБЪЕКТОВ В КОМПЬЮТЕРНОЙ ГРАФИКЕ**

**Аннотация.** В данной работе рассмотрены основные методы построения графических объектов, чаще всего применяемые в системах автоматизированного проектирования, на основе жизненного цикла строительного объекта. Рассмотренные методы проанализированы на предмет их преимуществ и недостатков, а также приведены принципы хранения данных, используемых в каждом из методов. В статье отдельно рассмотрены двумерные и трехмерные объекты, поскольку методы построения для них сильно отличаются и не всегда могут быть расширены до трехмерного пространства (для двумерных объектов) или урезаны до двумерного пространства (для трехмерных объектов). Предложенные в статье методы могут использоваться как отдельно, так и в комплексе, в зависимости от поставленных задач.

**Ключевые слова:** графический объект; примитив; полигональная модель; твердотельная геометрия; триангуляция

**Borodavka Yevgeniy**

Doctor of Philosophy, Docent, associate professor at Information technologies of Design and applied mathematics department, [orcid.org/0000-0002-7476-9387](http://orcid.org/0000-0002-7476-9387)

Kyiv National University of Construction and Architecture, Kyiv

**Kvasnevskiy Vladyslav**

Specialist of Development Department, [orcid.org/](http://orcid.org/)

Allbau Software Ltd., Kyiv

METHODS OF CONSTRUCTING OBJECTS IN COMPUTER GRAPHICS

**Abstract.** This paper describes the main methods of graphic objects that are often used in computer-aided design based on the building project lifecycle. The methods are analyzed for their strengths and weaknesses, as well as the principles are the storage of data in each of the methods. The article examined separately the two-dimensional and three-dimensional objects, as methods for constructing them quite different and cannot always be extended to three-dimensional space (two-dimensional objects) or cut to two-dimensional space (for three-dimensional objects). Proposed in article methods can be used separately or in combination, depending on the task. For 2D objects we are considering class diagram of graphics objects with elementary hierarchy. For 3D object we suggest to use either boundary representation or triangulated models.

**Keywords:** *graphic primitives; polygonal model; solid-state geometry; triangulation*

References

1. Skvortsov, A.V. (2002). *Delaunay triangulation and it application*. Tomsk: Tomsk university publishing, 128.
2. Borodavka, Y.V. (2013). *The core of building object model based on common set of geometric primitives and attributes*. *Management of Development of Complex Systems*, 15, 111-114.
3. Borodavka, Y.V. (2011). *The representation methods of building object*. *Management of Development of Complex Systems*, 8, 101-107.
4. Borodavka, Y. & Kvasnevskiy, V. (2014). *Methods of Geometric Objects Sorting and Their Implementation Based on Elements Auto-Numeration Plugg-In for AllPlan*. *Management of Development of Complex Systems*, 18, 128–134.
5. Kvasnevskiy, V. & Borodavka, Y. (2015). *Geometric methods of construction holes and sleeves for engineering networks in CAD System Allplan*. *Management of Development of Complex Systems*, 22, 128-133.
6. Borodavka, Y. & Tsiutsira, M. (2014). *Building model conception*. In *Proceedings of the WORLD Science Conference "Science and Education – Our Future"*. 3, November 24-26, 20014, Abu Dhabi, UAE, 53–55.
7. Borodavka, Y.V. (2014). *Automated tools for building object lifecycle support*. In *Proceedings of the international scientific and practical conference «CAD Allplan. Innovative development in architecture and construction»*. Kyiv, KNUCA, 10-13.
8. Malukh, V. (2010). *Introduction to Modern CAD: Lectures*. M.: DMK Press, 192. – ISBN 978-5-94074-551-8.
9. Wirth, Niklaus. (1989). *Algorithms and Data Structures*. New York: Wiley – ISBN 5-03-001045-9.
10. Borodavka, Y.V. (2010). *Model of extensible computer aided system for construction object lifecycle*. *Management of Development of Complex Systems*, 4, 69-71.

---

Посилання на публікацію

- APA Borodavka, Ye. & Kvasnevskiy, V. (2015). *Methods of constructing objects in computer graphics*. *Management of Development of Complex Systems*, Issue 24, 106 – 110.
- ГОСТ Бородавка Є.В. *Методи побудови об'єктів в комп'ютерній графіці [Текст] / Є.В. Бородавка, В.М. Квасневський // Управління розвитком складних систем. – 2015. – № 24. – С. 106 – 110.*