

УДК 656:004.75



ВІЗУАЛЬНЕ МОДЕЛЮВАННЯ ІНФОРМАЦІЙНИХ РЕСУРСІВ ТРАНСПОРТНОЇ ІНФРАСТРУКТУРИ

В.О. Алексієв¹¹ ХНАДУ, м. Харків, Україна, aleksiyev@gmail.com

Розглянуто візуальне моделювання гетерогенних розподілених мереж транспортних систем як програмної основи інтелектуалізації інфраструктури транспортних організацій. Запропоновано застосування мови UML для візуального моделювання GRID-систем у транспортних додатках.

ВІЗУАЛЬНЕ МОДЕЛЮВАННЯ, РОЗПОДІЛЕНІ МЕРЕЖІ, ТРАНСПОРТНА ІНФРАСТРУКТУРА

Вступ

Сучасна транспортна інфраструктура міст та регіонів є сукупністю інтелектуальних систем планування та моделювання транспортних мереж, керування дорожнім рухом та телематичними комплексами, які надають оперативну інформацію про стан дорожнього середовища та дозволяють синергетично взаємодіяти із всіма учасниками дорожнього руху. Інформація у транспортних системах існує у різному цифровому уявленні та графічному і мультимедійному вигляді. Цьому уявленню відповідає просторо-часове існування цифрового контенту. Інформаційна взаємодія користувачів відповідної ІТ інфраструктури транспортних організацій основана на електронному документообігу на рівні LAN або WAN. Розгляд властивостей цієї взаємодії потребує врахувань як сукупної продуктивності вузлів відповідної мережі, так і фінансових витрат на забезпечення відповідних енергоресурсів та сервісного обслуговування. Інформаційного простір транспортних організацій потребує оптимізації багатокритеріальної мережевої системи. Найбільш привабливою для такої оптимізації є концепція WEB 2.0, що є логічним продовженням розвитку колективного користування ресурсами Internet. Технології WEB 2.0 знімають потребу у програмуванні як для виконання задач обробки інформації відповідного електронного документообігу, так і при створенні своїх особистих WEB-ресурсів. Розглянемо невід'ємну частину використання такої технології – візуальне моделювання.

1. Аналіз досліджень та публікацій

Візуалізація, специфікування, конструювання й документування об'єктно-орієнтованих систем є призначенням мови UML. У дослідженнях Буч Г., Фаулер М., Ларман К. та інш. [1-3] запропоновано та доведено доцільність її застосування для створення комп'ютерних систем та мереж. У роботах росіян [4-7] показано, як за допомогою діаграм можна зробити візуальну модель системи з різних точок зору. Одна з діаграм, наприклад, може описувати взаємодію користувача із системою, інша – зміну станів системи в процесі її роботи,

третя – взаємодію між собою елементів системи і т.п. Складну систему можна представити у вигляді набору невеликих і майже незалежних моделей діаграм, причому жодна з них не є достатньою для опису системи й одержання повної уяви про неї, оскільки кожна з них опирається на певний аспект функціонування системи й виражає різний рівень абстракції.

Загальні проблеми інформатизації транспортної інфраструктури, її інформаційний розвиток на концептуальному рівні було ретельно розглянуто у дослідженнях [8-10]. У роботі [11] наведено відомості про застосування мови UML для візуального моделювання GRID – систем у транспортних додатках.

2. Постановка задачі

Уніфікована мова моделювання (Unified Modeling Language, UML) – це графічна мова для візуалізації, конструювання й документування систем, у яких головна роль належить програмному забезпеченню. Ця мова була розроблена для моделювання програм, що основані на об'єктно-орієнтованому підході. За допомогою UML можна розробити детальний план системи, що розроблюється, та яка містить не тільки її концептуальні елементи, такі як системні функції й бізнес процеси, але й конкретні особливості, наприклад, класи, написані на будь-якій мові програмування, схеми баз даних і програмні компоненти, що можуть повторно використовуватися.

Застосування технології UML для моделювання складних об'єктів та систем, наприклад, GRID-інфраструктури транспортної організації, є рішенням, якому практично не має альтернатив. Так, UML надає спільне подання структури таких систем у вигляді як графа, структурованого рисунка, так і забезпечує об'єктно-орієнтоване сприйняття систем. Поряд з цим слід відмітити, що візуальне моделювання на відміну звичайного підходу до алгоритмізації та структуризації транспортних систем надає просторове сприйняття проблеми. Однак у задачах створення інформаційних ресурсів на транспорті візуальне моделювання ще не знайшло достатньо широкого застосування.

Тому поставимо задачу показати на прикладах визначення топології серверних компонентів розподілених систем та супроводження програмних додатків, що програмні рішення, які основані на візуальному моделюванні, є зручними, гнучкими в умовах застосування як у промислових системах, так і у транспортних додатках.

3. Розподілені системи

Моделювання часу й простору – важливий елемент будь-якої розподіленої системи, що працює у реальному часі. Для візуалізації, конструювання й документування таких систем UML пропонує різні засоби, включаючи оцінки часу, часові діаграми, обмеження та позначені значення. Проектування систем реального часу й розподілених систем – складне завдання. Модель таких систем повинна висвітлити всі необхідні й достатні просторово-часові властивості системи.

Розподілені системи за своєю природою складаються з компонентів, фізично розташованих по різних вузлах. Дуже часте місце розташування (location) компонентів фіксується в момент установки системи. Але зустрічаються й такі системи, у яких компоненти мігрують із одного вузла на інший, наприклад у разі модернізації устаткування локальної обчислювальної мережі або введення нового сегменту, що сприяє зміні топології іншого. Засобами UML вигляд системи, з погляду розміщення, моделюється за допомогою відповідних діаграм розміщення, що описують топологію процесорів і пристроїв, на яких функціонує система. Артефактами є модулі, що виконуються, бібліотеки, таблиці, які розміщуються в цих вузлах. Кожний екземпляр вузла володіє власними екземплярами тих або інших артефактів, а кожний екземпляр артефакту належить тільки одному екземпляру вузла (хоча різні екземпляри артефакту одного виду можуть перебувати на різних вузлах).

Що торкається комп'ютерних ресурсів транспортних організацій, їх розвиток передбачає принцип «конкуренції» або «виживання» найбільш ефективних зв'язків згідно з властивостями самоорганізації. Слід зазначити, що можлива поява нового об'єднання комп'ютерів та інший розподіл завдань між окремими комп'ютерами. Особливо це важливо для вирішення проблем комп'ютерного забезпечення розвитку транспортної інфраструктури міст та регіонів. Такому уявленню проблем розвитку комп'ютерних ресурсів у транспортних додатках відповідає його подальше візуальне моделювання.

При комплексному використанні мережі Internet виникає необхідність у рішенні проблеми захисту інформації та локальної мережі в цілому. Особливо суттєво це питання стає, коли організація має загальнодоступні (публічні) Internet-сер-

віси (web-сервери, ftp-сервери, поштові сервіси, та інші), які розміщені в загальній локальній мережі. Підхід до створення захищених систем полягає у поділі локальної мережі та публічних серверів на окремі частини. Зона, у якій будуть розміщені публічні сервіси, називається демілітаризованою зоною – DMZ (Demilitarized Zone). Суть такої зони полягає в тому, що вона не входить безпосередньо ні у внутрішню, ні у зовнішню мережу, а доступ до неї може здійснюватися тільки за заздальгідь заданими правилами мережевого екрана.

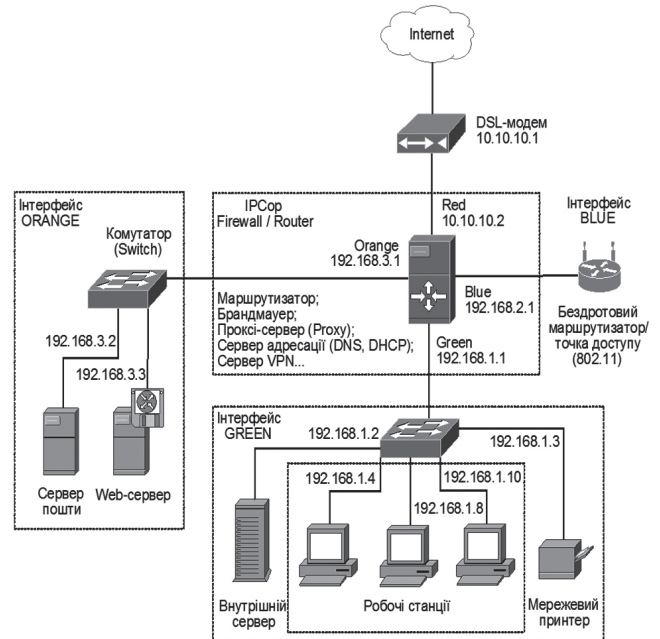


Рис. 1. Топологія мережі на базі рішення IPCop

Демілітаризована зона як правило служить для запобігання доступу із зовнішньої мережі до комп'ютерів внутрішньої мережі за рахунок виділення з локальної мережі в особливу зону всіх сервісів, що вимагають доступу із зовні. На серверах, розміщених в DMZ, не повинно бути ніякої інформації про користувачів та іншої конфіденційної інформації, не повинно бути особистих поштових скриньок співробітників – ці дані мають знаходитись у захищеній частині локальної мережі. (<http://hostinfo.ru/articles/487>).

Для роботи системи IPCop потрібно виділити x86-сумісний комп'ютер, бажано із об'ємом оперативної пам'яті порядку 512Мб. Наприклад, для проведення експерименту моделювання GRID-системи було обрано систему на базі Pentium-4 із 512Мб оперативної пам'яті та жорстким диском 80Гб.

При моделюванні топології розподіленої системи слід розглядати фізичне розташування як вузлів, так і артефактів. Якщо в центрі уваги перебуває керування конфігурацією системи, що розглядається, то моделювання розподілу вузлів є важливим для візуалізації, конструювання й документування

розміщення таких фізичних суттєостей, як модулі, що виконуються (бібліотеки та таблиці). Якщо більш важлива функціональність, масштабованість і пропускна здатність системи, то важливішим стає моделювання розподілу об'єктів, що реєструються. Компоненти й класи можуть бути матеріалізовані у вигляді артефактів. Так, на рис. 2 зображено діаграму розміщення системи на базі ICSop, де присутні артефакти: "processor" – комп'ютери, що виконують завдання серверних систем або робочих станцій та "network" – мережеве устаткування.

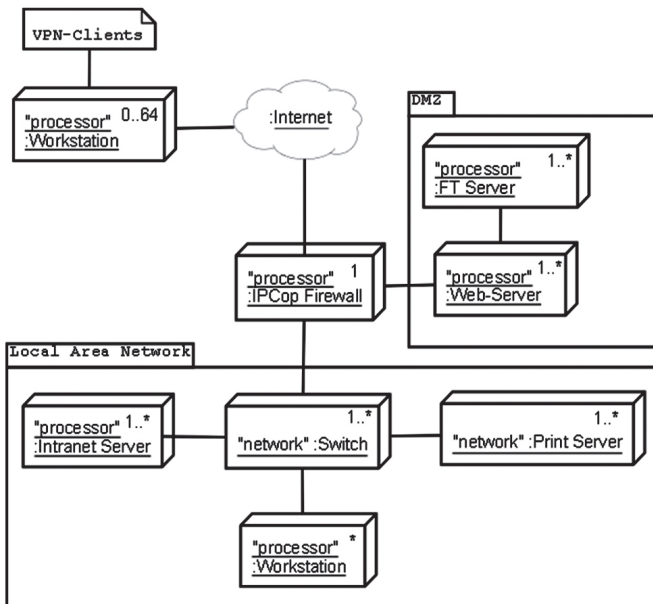


Рис. 2. Моделювання топології розподіленої системи

Моделювання розподілених систем засобами UML дозволяє ухвалити рішення щодо того, як розподілити об'єкти в системі. Питання розподілу об'єктів системи тісно пов'язані з питаннями паралелізму. Непродумане рішення та визначення напряму потоків даних може стати причиною дуже низької продуктивності, але складна топологія може привести до нестабільності. Слід зазначити, що мова UML призначена головним чином для моделювання програмних систем, хоча в комбінації з мовами моделювання апаратних засобів HDL (hardware description language) може виявитися досить корисною і при моделюванні апаратних систем. Крім того, UML має достатню потужність для моделювання топології автономних (stand-alone), вбудованих (embedded), клієнт-серверних (client-server) і безпосередньо розподілених (distributed) систем.

Таким чином, на діаграмі розміщення наведені вузли системи. В мові UML вузол (node) – це фізичний елемент, що існує на час виконання і є обчислювальним ресурсом. Вузол зображується у вигляді куба. Для того щоб вказати скільки однотипних елементів існує в мережі на діаграмі, використовується множина (multiplicity). Вона

записується у вигляді виразу з мінімальним та максимальним значенням. Множина може визначена як одиниця (1), нуль чи один (0..1), багато (0..*), один чи декілька (1..*). Для того щоб підкреслити значення вузлу на схемі розподіленої системи, використовуємо компонент UML-екземпляр. Екземпляром (instance) зветься певне уособлення абстракції, до якого можна застосувати операції та котрому властиве певний стан, що зберігає результати цих операцій.

Візуалізація, конструювання та документування великих систем передбачає роботу з багатьма класами, інтерфейсами, вузлами, компонентами, діаграмами та іншими компонентами. У разі масштабування таких систем буде необхідно організувати ці сутності у великі блоки. В мові UML, для організації елементів моделі групи, використовують пакети. Графічно вони зображуються у вигляді папки. Пакет об'єднує семантично близькі елементи, що мають тенденцію сумісно змінюватися. Структурний зв'язок між класами визначає набір зв'язків, які утворюються між об'єктами – екземплярами класів. Також на цій діаграмі присутні сутності, що анотують. Графічно вони представлені прямокутником із загнутих кутами, в ньому розміщуються текстовий чи графічний коментар.

На рис. 3 зображено діаграму пакетів, що уособлює структурну сутність програмного маршрутизатора ICSop. На цій діаграмі розглянуто роботу маршрутизатора із трьома мережевими картами, що забезпечують зв'язок локальної обчислювальної мережі з Internet та виділення демілітаризованої зони. На цій діаграмі застосовано властивості атрибуту – доступність (visibility).

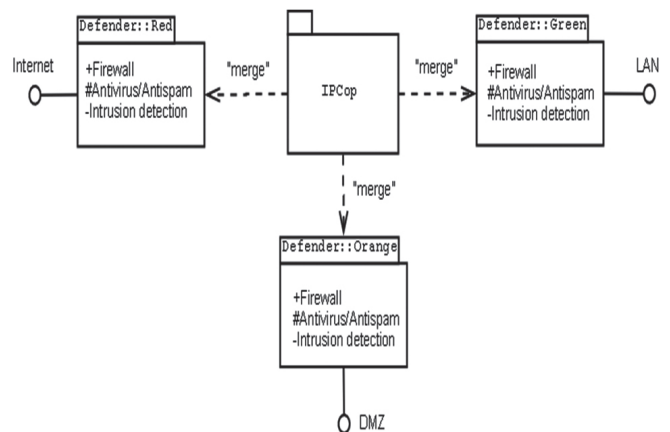


Рис. 3. Структурна схема маршрутизатора ICSop

4. Серверні компоненти

Засобами UML можна відокремлювати чотири рівня доступності: public (відкритий), protected (захисений), private (закритий), package (пакетний). Засоби явного зазначення доступності тих чи інших компонентів системи підвищують візуальне сприйняття в моделі, що розглядається. Для зв'язку

пакетів між собою на діаграмі використано залежність (dependency) – зв’язок між двома елементами моделі, в якій зміну одного елемента (незалежного) можна привести до зміни семантики другого елемента (залежного). Графічно такий зв’язок відбивається пунктирною лінією з напрямом. Також на діаграмі використовується графічний елемент інтерфейсу, що зображується у вигляді круга. На рис. 4 зображено структурну схему серверних компонентів, що складають систему IPSec. Ця схема зображена в термінах UML та дозволяє візуально спостерігати зв’язок між відповідними компонентами системи.

Система IPSec має модульну структуру і завдяки розширенням, що створюються спількою вільних розробників, може бути адаптована до рішен-

ня завдань будь-якої складності. Основою системи є пакетний фільтр (Firewall). Цей компонент захисту системи є стандартним рішенням для операційної системи Linux і базується на утиліті iptables. Завдяки цьому компоненту адміністратор системи може створювати та модифікувати правила, що управляють фільтрацією, перенаправленням (NAT – Network Address Translation) та модифікацією пакетів (<http://wikipedia.org>). Хоча технологія NAT є порівняно швидким рішенням для надання доступу до ресурсів Internet користувачам локальної обчислювальної мережі, та за умови використання IPSec, більш зручним рішенням буде застосування Proxu-серверу.

Процес адміністрування великих локальних обчислювальних мереж можна значно спро-

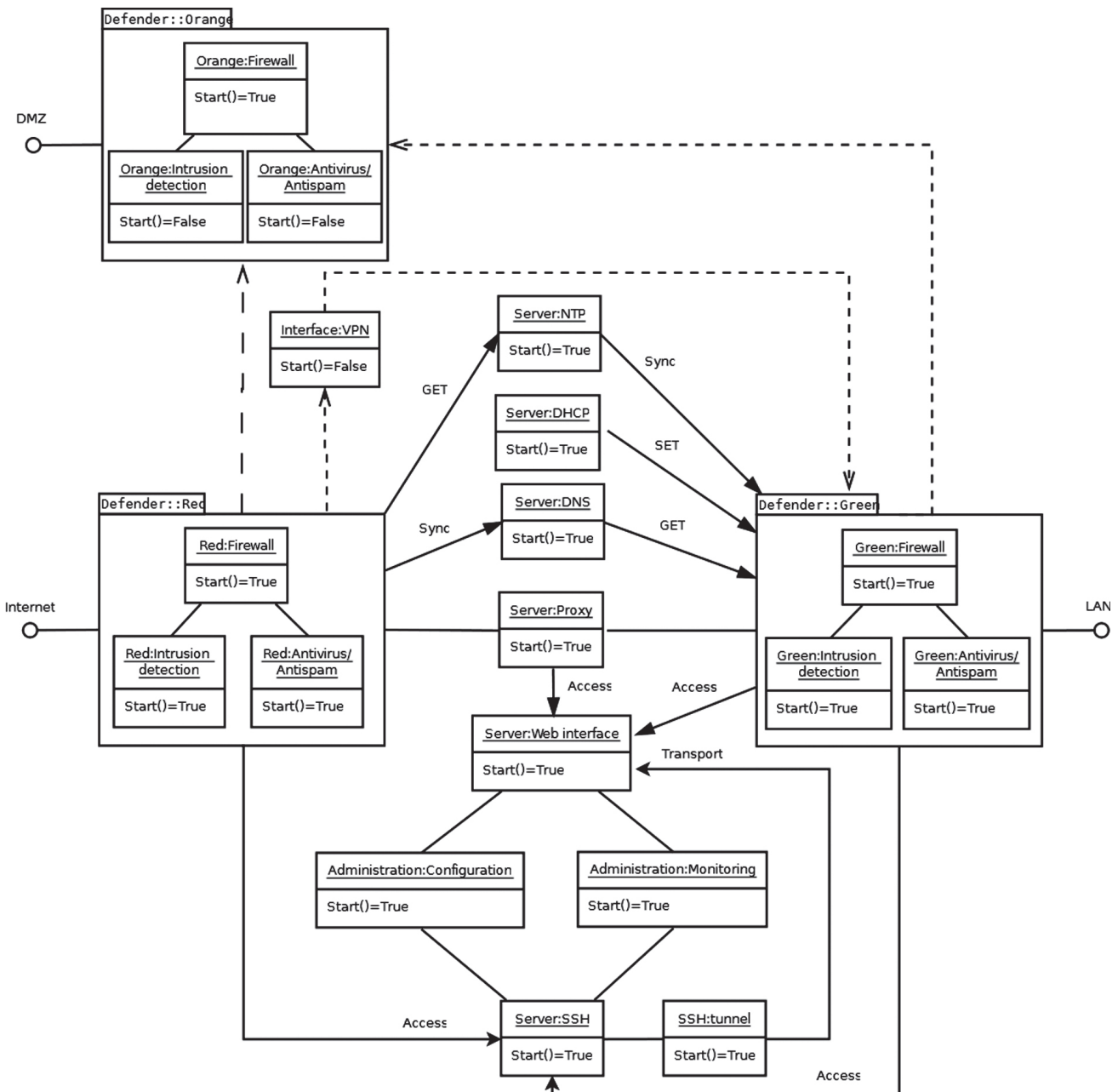


Рис. 4. UML-модель вузла

стити завдяки впровадженню технології DHCP. Цей протокол (англ. Dynamic Host Configuration Protocol – динамічна конфігурація вузла) дозволяє комп'ютерам автоматично одержувати IP-адресу та інші параметри, необхідні для роботи в мережі TCP/IP. Для цього комп'ютер звертається до спеціального серверу під назвою сервер DHCP. Адміністратор мережі може задати діапазон адрес, що розподіляються серед комп'ютерів. Це дозволяє уникнути ручного налаштування комп'ютерів мережі й зменшує кількість помилок. Протокол DHCP використовується в більшості великих мереж TCP/IP.

Відмінною рисою системи IPsec є адміністрування засобами web-інтерфейсу. Для цього одним із сервісів системи є web-server, доступ до якого можна отримати через внутрішній інтерфейс локальної мережі. Для адміністрування системи із зовні використовується протокол SSH. Secure Shell, SSH – мережевий протокол, що дозволяє проводити віддалене управління комп'ютером і передачу файлів. Схожий за функціональністю з протоколом Telnet, проте використовує алгоритми шифрування інформації, що передається.

При проектуванні GRID-системи основним є забезпечення групової роботи спільноти користувачів, які виступають у ролі віртуальної організації, тобто користувач такої системи незалежно від того, працює він у приміщенні організації чи знаходиться вдома, або в іншому місці з доступом до Internet, повинен отримати єдиний інтерфейс роботи у своєму віртуальному середовищі. Таку проблему з легкістю вирішує впровадження VPN-сервісу. VPN (віртуальна приватна мережа, англ. Virtual Private Network) – це логічна мережа, створена поверх інших мереж, на базі загальнодоступних каналів приватних віртуальних мереж, які дозволяють об'єднати віддалені офіси в єдину захищену корпоративну мережу з повним спектром телекомунікаційних послуг та з гарантованим якісним обслуговуванням. Безпека передавання пакетів через загальнодоступні мережі реалізується за допомогою шифрування, внаслідок чого створюється закритий для інших користувачів канал обміну інформацією.

Мова UML є зручним та багатофункціональним засобом графічного моделювання, але розробка діаграм без застосування комп'ютерних засобів не надасть проекту динаміки та гнучкості. Зараз на ринку домінують CASE-програмні засоби. Термін CASE (Computer-Aided Software Engineering) сьогодні розуміється досить широко. Раніше значення терміна обмежувалося питаннями автоматизації розробки програмного забезпечення, зараз воно набуло нового сенсу і охоплює процес розробки складних інформаційних систем у цілому. Тепер під терміном CASE-засіб розуміють-

ся програмні засоби, що підтримують процеси створення й супроводу таких систем, включаючи аналіз і формулювання вимог, проектування прикладних додатків та баз даних, генерування коду, тестування, документування, забезпечення якості, конфігураційне керування та керування проектом і т.п. CASE-засоби разом із системним програмним забезпеченням та технічними засобами утворюють повне середовище розробки (<http://www.citforum.ru/programming/case/gnatash/case/>).

Зараз існує багато програмних рішень CASE, що в свою чергу можуть підтримувати UML (http://en.wikipedia.org/wiki/List_of_UML_tools). Прикладом зручного та багатофункціонального пакету для розробки блок-схем, діаграм, структур та інших графічних схем є програмний засіб Microsoft Visio. Однак безпосередньо функціональність цього рішення не дозволяє генерувати код програми, або навпаки – за вихідним кодом створювати UML-модель проекту.

Visio допомагає перетворити технічні й бізнес-концепції у візуальну форму. Цей пакет із родини Microsoft Office призначений винятково для малювання діаграм. Visio має деякі додаткові можливості, але він є тільки засобом для ілюстрування документів MS Office.

5. Супроводження програмних додатків

Сучасні засоби розробки програмного забезпечення, як правило, є об'єкт-орієнтованими. Тому засоби UML значно доповнюють та оптимізують процес розробки та супроводження програмних додатків. Наприклад, при впровадженні програмного маршрутизатору, на базі IPsec потрібно було вирішити задачу інвентаризації наявних комп'ютерних систем, що входять до локальної обчислювальної мережі. Для цього реєструвалася інформація про фізичні адреси мережевих карт цих комп'ютерів. Ця інформація стала у нагоді для надання прав доступу до ресурсів Internet для певних груп, робочих станцій у локальній мережі. Найпростішим засобом отримання цих даних є команда "ipconfig /all" у операційній системі Windows та команда "ifconfig" у Unix-подібних операційних систем. Однак отримані дані потрібно було записувати в один файл, що складався адміністратором підрозділу, а вже потім передавався адміністратору серверу доступу. Зараз найбільш продуктивним засобом розробки програмного забезпечення фірми Microsoft є продукт Visual Studio Team Edition. Ця система у поєднанні з функціями пакету Visio надає змоги виконання зворотнього проектування, тобто на основі вихідного коду програми отримати UML модель проекту.

Розробка діаграм є популярною не тільки для наукових задач, але й у бізнесі (<http://www.>

computerra.ru/gid/raznosoft/316808/). Незважаючи на Linux- основу, проект Dia випускається ще для Windows- платформи (рис. 5).

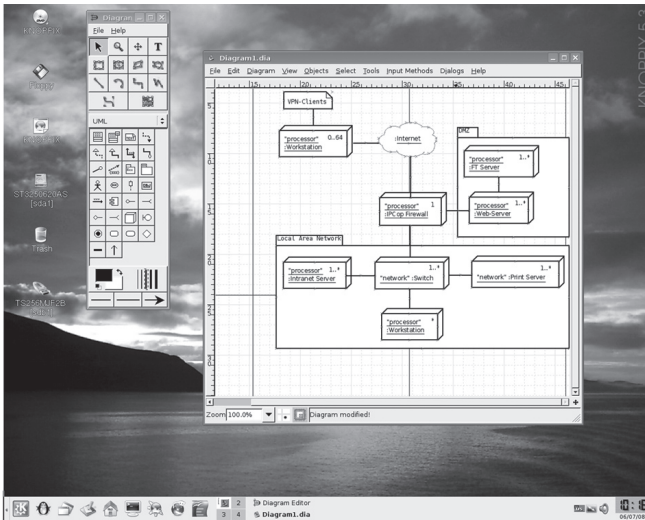


Рис. 5. Вікно проекту Dia у середовищі Linux KNOPPIX 5.3.1

Таким чином, для вирішення певних задач розробки UML-моделей можна обрати різні програмні засоби. Вони можуть бути як комерційними розробками, так і вільно поширюватися. Серед різноманітних засобів можна виділити комерційну систему Microsoft Visio та вільний кросплатформенний редактор Dia. На рис. 6 наведено відповідне формальне уявлення проходження завдань на рівні вузла обчислювальної мережі транспортного ситуаційного центру (ТСЦ).

Поряд з візуальним моделюванням необхідно постійно виконувати моніторинг поточного стану комп'ютерного ресурсу. Основним принципом об'єднання UML-моделі та засобів моніторингу є їх спільна програмна реалізація. Результат моніторингу обчислювальних процесів у відповідному вузлі відбиває фізичне уявлення його навантаження (рис. 7).

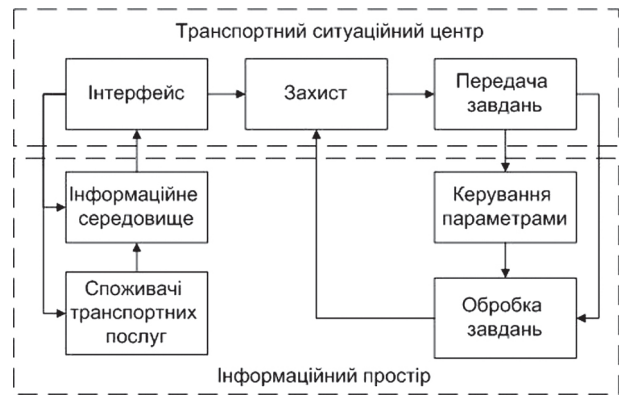


Рис. 6. Взаємодія споживачів

Висновки

Будь-які комп'ютерні ресурси організацій та підприємств, що мають стабільні тенденції до розвитку, мають і тенденції розвитку комп'ютерних ресурсів. ТСЦ координує використання цих гетерогенних розподілених ресурсів на основі застосування GRID-технології. Така технологія надає можливості застосування різноманітних ресурсів: обчислювальних, накопичення даних та комунікаційних. Звичайно таке об'єднання координується GRID-системою, а відповідна віртуальна мережа дозволяє технічно об'єднати розрізнені внутрішні GRID та кластери у єдиний інформаційний простір, яка координується вже єдиною GRID-системою, що надається користувачу як єдина віртуальна. Особливість використання для формування єдиного інформаційного простору з використанням GRID- технологій полягає не тільки у технічній організації розподілених обчислень гетерогенного середовища, але й у створенні певної соціальної структури, до якої залучаються користувачі системи. Такий принцип взаємодії комп'ютерних ресурсів та користувачів створює своєрідний механізм інтелектуалізації GRID- технологій. Засоби для створення віртуальних організацій користувачів надають технології WEB 2.0 поруч із системами

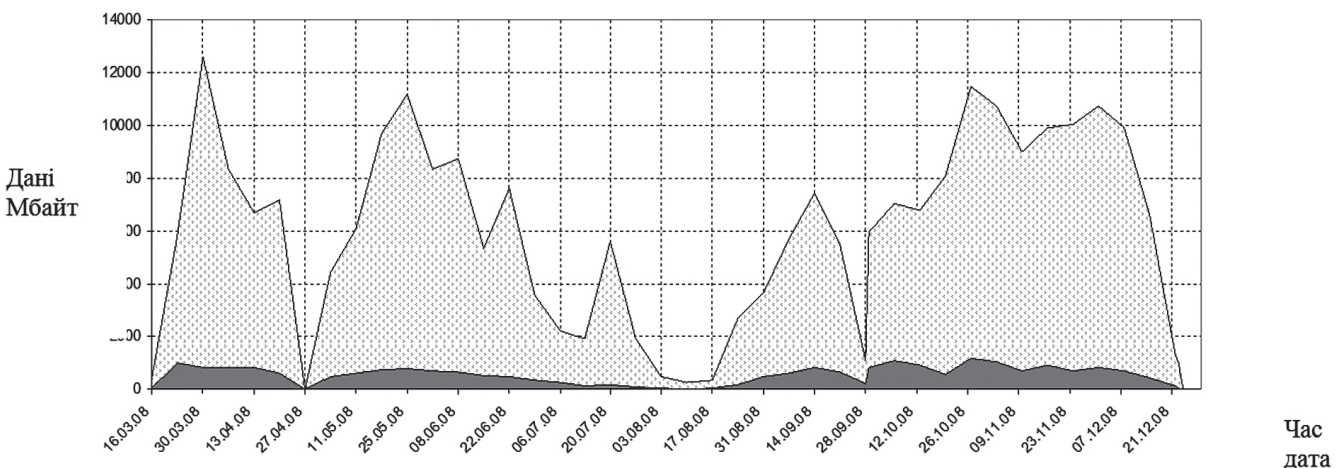


Рис. 7. Завантаження вузла протягом року: ■ – вихідна інформація, □ – вхідна

віртуалізації мережевої інфраструктури фізичної організації чи підприємства, наприклад, транспортного профілю.

Впровадження GRID-технологій надає не просто інформаційну систему, а своєрідний інтелектуальний регулятор, що поєднує переваги систем програмного керування із адаптивними системами, що працюють на основі синтезу керуючого впливу. Вона буде складовою частиною інтелектуальної технології управління рухом наземного транспорту великих міст та регіонів. З одного боку така система буде входити до простору WAN транспортної корпорації, а з іншого – в неї занурено LAN мережу різних транспортних організацій, що входять до відповідної корпорації (об'єднання).

Список літератури: 1. Буч, Г. Язык UML. Руководство пользователя. – 2-е изд. [Текст] / Г. Буч, Д. Рамбо, И. Якобсон; Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2007. – 496 с. 2. Фаулер, М. UML. Основы / М. Фаулер, К. Скотт; Пер. с англ. – СПб: Символ-Плюс, 2002. – 192 с. 3. Ларман, Крэг. Применение UML и шаблонов проектирования. – 2-е издание [Текст] / Крэг Ларман; Пер. с англ. – М.: Изд. дом «Вильямс», 2004. – 624 с. 4. Визуальное моделирование: теория и практика [Электронный ресурс] / Д.В. Кознов // Интернет-университет информационных технологий. – 2007. – 248 с. – Режим доступа: <http://www.intuit.ru/department/se/vismodtp/>. 5. Бабич, А.В. Введение в UML [Электронный ресурс] / А.В. Бабич // Интернет-университет информационных технологий. – 2008. – 176 с. – Режим доступа: <http://www.intuit.ru/department/se/intuml/>. 6. Леоненков, А.В. Нотация и семантика языка UML [Электронный ресурс] / А.В. Леоненков // Интернет-университет информационных технологий. – 2005. – 320 с. – Режим доступа: <http://www.intuit.ru/department/pl/umlbasics/>. 7. Тихонов, К. Карманное зазеркалье: мобильная дополненная реальность [Электронный ресурс] / К. Тихонов // Компьютерра. – 2008. – № 34. – Режим доступа: <http://www.sheep.ru/blog/txt/mobile-augmented->

realit. 8. Алексієв, О.П. Інформатизація транспортних систем [Текст] / О.П. Алексієв, В.О. Алексієв, В.О. Хабаров, Г.Г. Четвериков // Біоніка інтелекту: наук.-техн. журнал. – № 3 (74). – С. 52–57. 9. Богомолов, В.О. Проблема створення єдиного інформаційного простору транспортних організацій [Текст] / В.О. Богомолов, В.О. Алексієв // Автомобільний транспорт : Сб. науч. трудов. – 2009. – Вып. 25. – С. 222–225. 10. Богомолов, В.О. Концептуальне обґрунтування та синергетичний підхід до розвитку транспортних систем [Текст] / В.О. Богомолов, В.О. Алексієв // Інформаційно-керуючі системи на залізничному транспорті : Науково-технічний журнал. – 2009. – № 5 (78). – С. 59–63. 11. Алексієв, В.О. Застосування GRID – технології у транспортному ВНЗ: Навч.-методичний посібник [Текст] / В.О. Алексієв.-Харків: ХНАДУ, 2008. – 208 с.

Надійшла до редколегії 01.02.2011.

УДК 656:004.75

Визуальное моделирование информационных ресурсов транспортной инфраструктуры / В.О. Алексеев // Бионика интеллекта: науч.-техн. журнал. – 2011 – № 1 (75). – С. 107–113.

Рассмотрено визуальное моделирование гетерогенных распределённых сетей транспортных систем как программной основы интеллектуализации инфраструктуры транспортных организаций. Предложено применение языка UML для визуального моделирования GRID-систем в транспортных приложениях.

Ил. 7. Библиогр.: 11 назв.

UDC 656:004.75

Visual modeling of information resources of the transport infrastructure / V.O. Alekseyev // Bionics of Intelligence: Sci. Mag. – 2011. – № 1 (75). – P. 107–113.

We consider the visual modeling of heterogeneous distributed networks of transport systems as a policy framework intellectualization of transport infrastructure organizations. Proposed use of UML for visual modeling GRID-systems in transport applications.

Fig. 7. Ref.: 11 items.