

УДК 004.94+519.876.5

В.В. Точилін

Розробка моделі проектування компонентної бази програмних засобів дослідження комп'ютерної мережі на основі інтероперабельного середовища

На основе исследования современных программно-инструментальных средств проектирования компьютерных сетей и их функционирования разработана архитектура и экспериментальная реализация интероперабельной среды для проектирования имитационных моделей конструктивных элементов и протоколов передачи данных компьютерных сетей.

Based on the study of modern software design tools of computer networks and their functioning architecture and an experimental implementation of an interoperable environment for the design of simulation models of structural elements and data communication protocols computer networks is developed.

На основі дослідження сучасних програмно-інструментальних засобів проектування комп'ютерних мереж та їх функціонування розроблено архітектуру та експериментальну реалізацію інтероперабельного середовища для проектування імітаційних моделей конструктивних елементів та протоколів передачі даних комп'ютерних мереж.

Вступ. Задача дослідження майбутньої комп'ютерної мережі (КМ) на етапі її проектування, допомога адміністраторам КМ при штатному та, особливо, позаштатному (аварійному) режимах роботи КМ є, без сумніву, нагальною та актуальною.

Найбільш розповсюдженим методом розв'язання цієї задачі є побудова моделі КМ та подальше дослідження її програмно-інструментальними засобами імітаційного моделювання [1], такими, як *OmNet++* (<http://www.omnetpp.org/>), *ns2* (<http://www.isi.edu/nsnam/ns/>), *OPNET Network Planner* (<http://www.opnet.com/>) тощо.

Вказані програмно-інструментальні засоби надають можливість побудувати модель КМ, використовуючи вбудовану бібліотеку моделей компонентів мережі (моделі мережевого обладнання, протоколів, каналів зв'язку та ін.). Для створення відсутніх моделей компонентів КМ програмно-інструментальні засоби мають унікальний програмний інтерфейс (*API*).

Якість результатів моделювання значною мірою залежить від точності та якості моделей компонентів КМ. На результати впливає також наявність моделей компонентів КМ, що моделюють обладнання та протоколи, які застосовуються в КМ.

Отже, для якісного моделювання КМ програмно-інструментальний засіб повинен мати якісну і повну бібліотеку моделей компонентів КМ.

Однак слід зазначити, що створення моделей компонентів КМ потребує значних зусиль. Процес створення моделі компонента КМ, відповідальної за роботу нового обладнання або протоколу, потребує поглибленого знання принципів побудови моделей компонентів КМ для конкретного програмно-інструментального засобу, принципів роботи і експлуатаційних параметрів описуваного обладнання або протоколу, а також навичок програмування.

Зважаючи на вказані труднощі, програмуванням бібліотеки компонентів КМ здебільшого займаються корпорації – розробники програмно-інструментального комплексу. У свою чергу це спричиняє значні затримки з появою моделей нових протоколів та обладнання, а також високу ціну користування таким програмно-інструментальним засобом.

Виникає актуальне завдання швидкого та якісного створення моделей компонентів КМ для конкретного програмно-інструментального засобу.

Автором запропоновано середовище проектування, за допомогою якого можливо побудувати необхідні (відсутні) компоненти КМ без

навичок програмування і вивчення взаємодії компонентів та програмно-інструментального засобу імітаційного моделювання.

Впровадження даного середовища дозволить користувачам програмно-інструментальних засобів імітаційного моделювання (адміністраторам КМ, інженерам та дослідникам) уникнути очікування коли розробник впроваджує необхідні компоненти до своєї програми та суттєво скоротити витрати на самостійну розробку.

Розробка архітектури середовища проектування компонентів комп'ютерної мережі

Дослідження принципів побудови моделей компонентів КМ в різних програмно-інструментальних засобах дозволило запропонувати наступні вимоги до цих засобів імітаційного моделювання КМ:

- простий та широковідомий засіб проектування моделей;
- доступність засобів та методів формальної верифікації побудованої моделі;
- інтероперабельність побудованої моделі (здатність бути застосованою як компонент КМ в усіх програмно-інструментальних засобах, які відповідають даним вимогам).

Вивчення фахової літератури дозволяє зробити узагальнюючий висновок, що найбільш ефективно для проектування моделей в межах запропонованих вимог може бути застосовано математичну модель, базовану на узагальнених стохастичних мережах Петрі (*GSPN*) [2].

Вперше мережі Петрі були запропоновані в 1962 р. для аналізу асинхронних паралельних процесів [3, 4].

Як засіб моделювання мережі Петрі мають багато переваг, а саме:

- математичний апарат мереж є широкоживим та легким для вивчення;
- як графічний засіб мережі Петрі можна застосовувати для наочного відображення процесів, які відбуваються в моделі системи;
- відносно легко можна зробити редагування моделі, побудованої на базі мереж Петрі;
- для опису паралельних процесів, які відбуваються в комп'ютерних мережах, узагальнені стохастичні мережі Петрі найбільш придатні,

оскільки модель *GSPN* є найбільш адекватною до реальної функціональної системи. Результатом цього є більш точні прогнозовані характеристики поведінки системи, що проектується [2, 5];

• існує велика кількість методів аналізу та верифікації моделі [6–8], побудованої за допомогою *GSPN*, які базуються на формальних методах аналізу мережі Петрі. Аналізуючи такі властивості, як досяжність, обмеженість, активність тощо, а також використовуючи методи формального дослідження мережі Петрі, можна бути більш впевненим в коректності даної моделі [5, 9, 10].

Отже, застосування узагальнених стохастичних мереж Петрі для побудови моделей компонентів КМ відповідає першим двом вимогам, запропонованим для вирішення завдання швидкого та якісного створення моделей компонентів КМ.

Для досягнення вимог третього пункту, автор пропонує створення середовища, що базується на програмному інтерфейсі (*API*) для побудови компонентів КМ програмно-інструментального засобу і надає універсальну, незалежну від програмно-інструментального засобу, платформу побудови моделей компонентів [1, 11].

Розроблене середовище отримало назву *Інтероперабельне середовище проектування компонентів комп'ютерної мережі* (рис. 1).

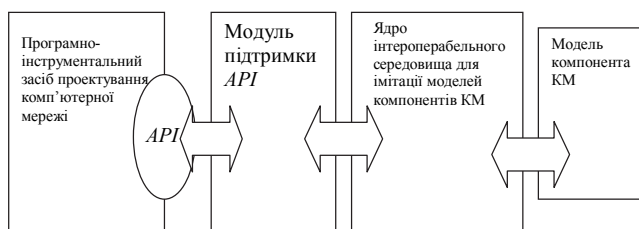


Рис. 1. Функціональна схема середовища проектування

Основним модулем вказаного середовища побудови моделей компонентів КМ є *Ядро інтероперабельного середовища*. Цей модуль являє собою *Петрі-машину*, яка під час моделювання, який задається програмно-інструментальним засобом, імітує модель компоненту КМ, побудовану за допомогою мереж Петрі.

За транслявання запитів програмно-інструментального засобу в *Ядро інтегрованої середовища* відповідальний *Модуль підтримки API* (див. рис. 1), який залежить від конкретного програмно-інструментального засобу.

Отже, після розробки та впровадження в програмно-інструментальний засіб *Модуль підтримки API*, за допомогою цього програмно-інструментального засобу можливо буде проектувати КМ з підтримкою будь-яких моделей компонентів КМ, розроблених за даною технологією.

Впровадження інтегрованої середовища в програмно-інструментальний засіб *OmNet++*

Експериментальну апробацію інтегрованої середовища проектування конструктивних елементів імітаційних моделей було проведено для програмно-інструментального засобу імітаційного моделювання *OmNet++* [12].

OmNet++ – модульний програмний комплекс моделювання дискретних подій в мережі. Цей комплекс можна застосувати при вирішенні наступних проблем:

- моделюванні
 - провідної та безпроводної комп'ютерної мережі;
 - мережевого протоколу;
 - мереж організації черг;
 - мультипроцесорів та інших розподілених апаратних систем;
- валідації (перевірка) встановленим вимогам архітектури апаратної системи;
- оцінюванні аспектів роботи складних систем програмного забезпечення;
- моделюванні будь-якої дискретної системи, в якій об'єкти спілкуються за допомогою повідомлень.

OmNet++ пропонує інфраструктуру та інструменти для моделювання, де імітаційна модель будується на основі її топології та моделей її елементів (наприклад, для комп'ютерної мережі такими елементами є: канали зв'язку, обладнання передавання даних, методи доступу, протоколи та ін.).

Керувати моделюванням в *OmNet++* можна під різними інтерфейсами користувача. Існує

розвинений графічний інтерфейс з динамічним відображенням процесу моделювання, який можна використовувати для демонстрації роботи моделі та її налаштування. За допомогою інтерфейсу командного рядка можливо автоматизувати деякі аспекти моделювання (послідовне моделювання декількох моделей, підбір параметрів модулів тощо).

В *OmNet++* модель складається з модулів, котрі спілкуються за допомогою повідомлень (рис. 2). *Простими модулями* називають модулі, написані мовою програмування C++ з використанням бібліотеки класів моделювання. *Прості модулі* можуть бути згруповані для отримання *складених модулів*, які в свою чергу теж можуть бути згруповані.

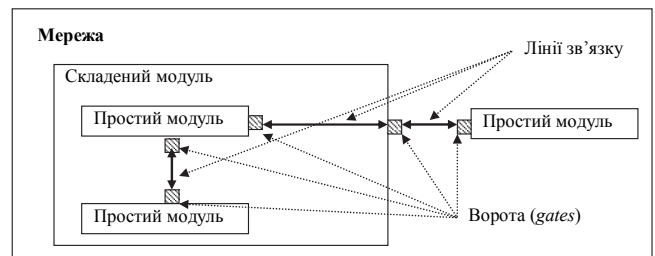


Рис. 2. Прості та складені модулі програмно-інструментального засобу *OmNet++*

Кількість рівнів ієрархії групування модулів є довільною, що сприяє більш коректному відображенню логічних зв'язків фактичного об'єкта дослідження в моделі.

Модулі, як правило, мають параметри, за допомогою яких налаштовують його поведінку. Складені модулі можуть передавати параметри для своїх підмодулів.

Модель в *OmNet++* називається *мережею* і являє собою самостійний складений модуль.

При проектуванні нема різниці між складеними та простими модулями, що дозволяє користувачу змінювати модулі, замінювати прості модулі на складені та навпаки, не торкаючись модулів вищого рівня ієрархії.

Спілкування модулів відбувається за допомогою повідомлень, які мають як зумовлені дані, наприклад, час моделювання, так і довільні.

Відправлення та прийом повідомлень в модулі здійснюється через *ворота (gates)*, яких є два типи: ворота входу та виходу.

Ворота можуть об'єднатися для передачі повідомлень лінією зв'язку (див. рис. 2).

Складені модулі тільки передають повідомлення між власними підмодулями та зовнішніми модулями. В той же час прості модулі безпосередньо продукують, отримують та перетворюють повідомлення.

Впровадження інтероперабельного середовища в програмно-інструментальний засіб *OmNet++* здійснено як окремий простий модуль *PetryNet*. За допомогою параметрів цього модуля можливо вказати модель компонента КМ, побудовану за допомогою узагальнених мереж Петрі, а також налаштувати її поведінку.

Послідовність дій при створенні та дослідженні моделі компонента КМ з використанням інтероперабельного середовища

Для побудови моделі компонента КМ з використанням інтероперабельного середовища, дослідження його поведінки, а також реакції на нього моделі КМ, необхідно виконати наступні кроки:

1. Створення моделі компонента КМ за допомогою математичного апарату узагальнених стохастичних мереж Петрі, застосовуючи для цього будь-який програмний комплекс побудови моделей на базі мереж Петрі (наприклад, *TimeNet*, *GreatSPN* тощо [13–15]).

2. Дослідження та верифікація побудованої моделі з використанням формальних методів дослідження мереж Петрі.

3. Опис простого модуля *OmNet++* з вказівкою на файл, який містить опис моделі мережі Петрі;

4. Побудова моделі КМ в інструментально-програмному комплексі *OmNet++* з використанням створеного модуля;

5. Дослідження поведінки КМ та вказаних параметрів мережі Петрі за допомогою аналітичного інструментарію *OmNet++*.

6. Дослідження поведінки модуля компонента КМ з використанням файла журналу подій в мережі Петрі.

Створення та верифікація моделі мережі комутації пакетів. Розглянемо спрощену модель мережі комутації пакетів з встановлен-

ням з'єднання [16]. Для побудови такої моделі автор використав програмний засіб аналізу узагальнених мереж Петрі *TimeNet* [15] (рис. 3).

TimeNet – графічний інтерактивний інструментально-програмний засіб для моделювання за допомогою стохастичних мереж Петрі. Він був розроблений в *Institut für Technische Informatik* при *Technische Universität Berlin* студентами, програмістами та магістрами під керівництвом професора *Günter Hommel*. Розробка почалася в 1991 р. Прототипом інструменту слугував пакет *DSPNexpress*. *TimeNet* написано мовою програмування *C++*, а графічний інтерфейс, починаючи з *V.4*, побудований на мові програмування *Java*.

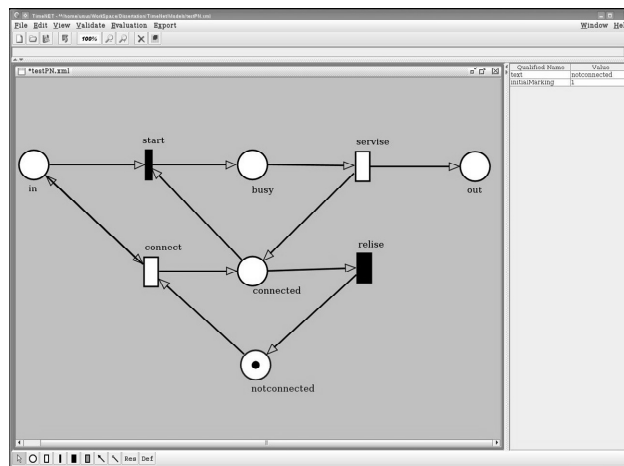


Рис. 3. Модель *GSPN*, орієнтованої на зв'язок мережі комутації пакетів

Розглянемо дану модель.

В моделі мережі комутації пакетів з встановленням з'єднання зв'язок між джерелом і приймачем має відбутися перед передачею першого пакету (спрацювання переходу *connect*). Для встановлення зв'язку витрачається певний час. Неefективно встановлювати зв'язок для передачі одного пакета, тому система підтримує зв'язок, поки необхідно передавати пакети. За відсутності останніх протягом певного часу зв'язок припиняється (спрацьовує перехід *relise*).

Встановлення зв'язку відбувається під час прибуття першого пакета для передачі (позиція *in*), в цій позиції знаходяться маркери, що асоціюються з пакетами, які можна передавати у мережі зв'язку. Передача відбувається послідовно, один за одним.

Значення всіх позицій та переходів даної моделі подано в таблиці.

Значення позицій та переходів в моделі, орієнтованій на зв'язок мережі комутації пакетів

Позиції	
<i>In</i>	Буфер пакетів, що повинні передаватися по мережі
<i>Connected</i>	Зв'язок встановлено
<i>Notconnected</i>	Зв'язок не встановлено
<i>Busy</i>	Передача одного пакета даних
<i>Out</i>	Буфер пакетів, переданих по мережі
Переходи	
<i>Connect</i>	Встановлення зв'язку
<i>Relise</i>	Розрив зв'язку
<i>Start</i>	Початок передачі пакета даних
<i>Servise</i>	Передача одного пакета даних

В позицію *in* потрапляють маркери, коли програмно-інструментальний засіб передає пакет до *GSPN*-моделі. А з позиції *out* маркери, перетворюючись у пакети, потрапляють знову до програмно-інструментального засобу.

Для верифікації побудованої моделі можна використовувати формальні методи дослідження мереж Петрі. Існує велика кількість праць, які досліджують це питання [4, 5, 10, 16].

Опис модулю орієнтованого на зв'язок мережі комутації пакетів. Для використання моделі мережі комутації пакетів з встановленням зв'язання в програмно-інструментальному засобі *OmNet++* побудовано дочірній модуль *demoPetriNet*. Цей модуль наслідує простий модуль інтероперабельного середовища *PetryNet*. Використовуючи прийняту в *OmNet++* мови опису (*NED, Network Description*) задано параметри поведінки побудованого модулю (рис. 4).

На рис. 4 подано опис моделі *demo PetryNet*.

По-перше, вказано шлях до файла, який описує модель мережі комутації пакетів (параметр *PetriNetFile*). Для зв'язку модуля *demoPetriNet* з іншими модулями *OmNet++* описано наступні ворота (секція *gates*):

in – вхідні ворота, *out* – вихідні ворота.

Для зв'язання даних воріт з позиціями мережі Петрі застосовано наступні правила:

RuleIn_ (назва вхідних воріт) = перелік вхідних позицій – коли у вхідні ворота по лінії зв'язку надходить пакет даних – у вказаних позиціях доставляється по одному маркеру;

RuleOut_ (назва вихідних воріт) = перелік вихідних позицій – коли в усіх вказаних позиціях з'являються маркери, то з кожної позиції видаляється по одному маркеру, а через вказані вихідні ворота виходить пакет даних.

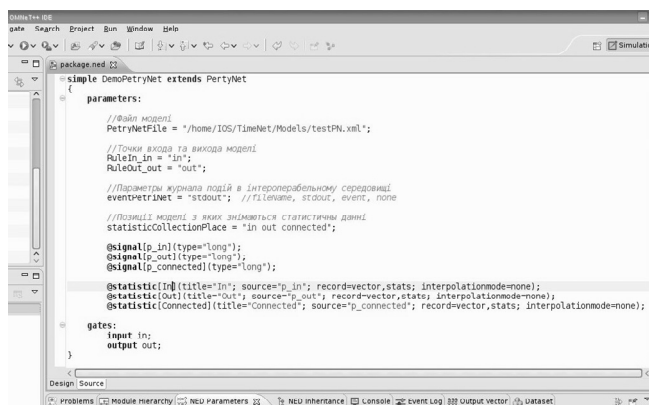


Рис. 4. Опис моделі, орієнтованої на зв'язок мережі комутації пакетів в системі *OmNet++*

Далі задається спосіб запису журналу подій, які відбуваються в мережі Петрі (параметр *EventPetriNet*).

Можливі наступні варіанти:

- *event* – запис журналу в вікні емуляції *OmNet++*;
- *stdout* – вивід на консоль;
- *none* – без виводу журналу подій;
- шлях до файла – запис журналу подій у вибраний файл.

Можливо також задати позиції, кількість маркерів в яких буде відслідковано аналітичною частиною програмно-інструментального засобу імітаційного моделювання *OmNet++*.

Для цього вкажемо інтероперабельному середовищу, які позиції необхідно контролювати (параметр *statisticCollectionPlace*), а також задамо *OmNet++*, яким чином відслідковувати ці параметри (директиви *@signal* і *@statistic*).

Побудова моделі КМ в інструментально-програмному комплексі *OmNet++* з використанням створеного модуля та її імітація. Після того як модуль *demoPetriNet* побудовано, його можна використовувати при моделюванні КМ в програмно-інструментальному засобі *OmNet++* (рис. 5).

Моделювана КМ являє собою чотири клієнтських комп'ютери (*etherHost* – *etherHost3*),

які за допомогою свіча (*etherSwitch*) зв'язані з сервером (*Server*). На каналі від свіча до сервера (висхідному каналі) встановлено модуль мережі комутації пакетів, побудованої за допомогою мереж Петрі (*demoPetriNet_UpChanel*). Модулі *splitter* і *splitter1* необхідні для розділення висхідного та низхідного каналів.

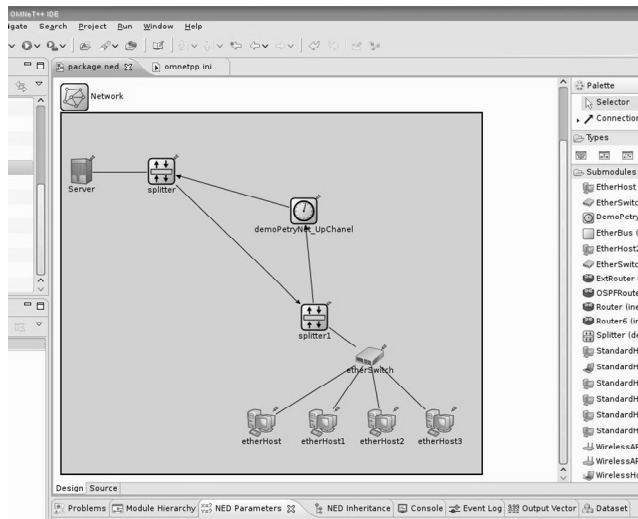


Рис. 5. Топологія КМ з використанням модуля *demoPetriNet*

Для дослідження побудованої КМ треба запустити імітацію.

Дослідження поведінки КМ та вказаних параметрів мережі Петрі за допомогою аналітичного інструментарію OmNet++. Програмно-інструментальний засіб імітаційного моделювання *OmNet++* дозволяє досліджувати параметри і поведінку моделі КМ (рис. 6).

Інтероперабельне середовище дозволяє також відстежувати поведінку маркерів у довільній (вказаній при описі моделі) позиції мережі Петрі (рис. 7).

Дослідження поведінки модуля компонента КМ з використанням файла журналу подій в мережі Петрі. Ядро інтероперабельного середовища під час імітації моделі дозволяє створювати журнал подій, які відбуваються в мережі Петрі.

Це забезпечує контроль поведінки мережі Петрі для виявлення помилок в проектуванні або параметрах запуску мережі Петрі.

Журнал подій дозволяє також аналізувати поведінку моделі на основі мережі Петрі під час

впливу на неї *реальної* комп'ютерної мережі (рис. 8).

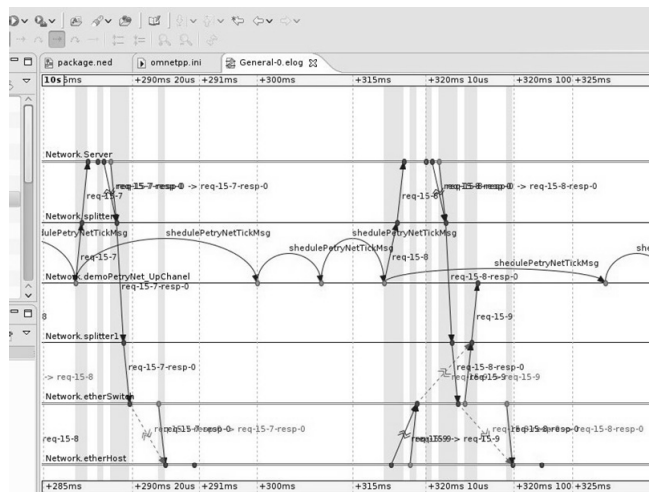


Рис. 6. Графік поведінки модельованої КМ

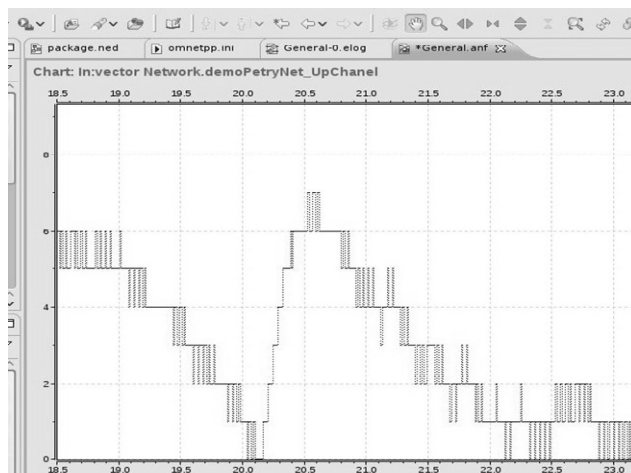


Рис. 7. Динаміка зміни кількості маркерів в позиції вхідного буферу *in*

Висновки. Впровадження програмно-інструментальних засобів імітаційного моделювання для проектування нових та дослідження існуючих комп'ютерних мереж безпосередньо залежить від якості та актуальності компонентної бази даних програмно-інструментальних засобів. Підтримка актуальності на даний час потребує значних фінансових витрат від розробників таких засобів, що також обумовлює їх ціну.

Запропоноване інтероперабельне середовище дозволяє створити моделі конструктивних елементів КМ без поглибленого знання програмування та знання особливостей *API* конкретного програмно-інструментального засобу імітаційного моделювання. Розробку таких мо-

делей можуть здійснити навіть користувачі програмних засобів імітаційного моделювання.

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	Note:											
3	PL - place											
4	- Number of markers indicated are in the position after the tick											
5	TR - transition											
6	- use the following notation:											
7	() passive transition											
8	(-) open transition											
9	(+) active transition											
10	(>) close transition											
11	(>>) open & close transition											
12	simTime	TR:	connect	relise	service	start	PL:	busy	connected	in	notconnected	out
13	(int)							0	0	1	1	0
14	0.01		>	-	-	-		0	0	0	0	0
1049	10.36		>	-	-	-		0	0	0	0	0
1062	10.49		-	+	-	-		0	0	4	0	0
1063	10.5		-	+	-	-		0	0	4	0	0
1064	10.51		-	+	-	-		0	0	4	0	0
1065	10.52		-	+	-	-		0	0	4	0	0
1066	10.53		-	+	-	-		0	0	5	0	0
1067	10.54		-	+	-	-		0	0	5	0	0
1068	10.55		-	+	-	-		0	0	5	0	0
1069	10.56		-	>	-	-		0	0	5	1	0
1070	10.57		>	-	-	-		0	0	5	0	0
1071	10.58		>	-	-	-		0	1	6	0	0
1072	10.59		-	-	-	>>		1	0	5	0	0
1073	10.6		-	-	>	-		0	0	5	0	0
1074	10.61		-	-	>	-		0	1	6	0	1
1075	10.62		-	-	-	>>		1	0	5	0	0
1076	10.63		-	-	>	-		0	0	5	0	0
1077	10.64		-	-	>	-		0	1	5	0	1
1078	10.65		-	-	-	>>		1	0	5	0	0
1079	10.66		-	-	>	-		0	0	5	0	0
1080	10.67		-	-	+	-		0	0	5	0	0
1081	10.68		-	-	-	-		0	0	5	0	0

Рис. 8. Журнал подій імітації моделі GSPN-мережі, комутації пакетів, відкритий для аналізу у табличному редакторі

Інтероперабельність запропонованого середовища дозволяє використовувати моделі, розроблені не тільки для конкретного програмно-інструментального засобу імітаційного моделювання, а й для всіх засобів, які підтримують запропоноване середовище.

Використання узагальнених мереж Петрі дозволяє проводити верифікацію розроблених моделей компонентів КМ, використовуючи формальні методи дослідження мереж Петрі, що додає впевненості в коректності моделі.

Спрощення побудови моделей компонентів КМ сприяє суттєвому зниженню витрат у дослідженнях КМ, коли використовуються сучасне мережеве обладнання та протоколи. З іншого боку, інтероперабельне середовище допомагає у розробці нових протоколів та обладнання, оскільки дозволяє провести імітацію роботи моделей нового обладнання або протоколу під навантаженням, близьким до реальної КМ.

1. *Точилін В.В.* Програмно-інструментальні системи імітаційного моделювання комп'ютерних мереж та можливість їх розширення мережами Петрі / Матеріали міжнар. наук.-техн. конф. «Комп'ютерні системи та мережні технології» // Проблеми інформатизації та управління. – 2008. – № 1(23). – С. 46–53.

2. *Гриценко В.И., Гладун А.Я.* О применении временных и стохастических сетей Петри в задачах оценки производительности высокоскоростных сетей связи // Кибернетика и вычислительная техника. – 1999. – **103**. – С. 98–111.
3. *Мурата Т.* Сети Петри: Свойства, анализ, приложения // ТИИЭР. – Апр. 1989. – **77**, № 4. – С. 41–85.
4. *Питерсон Дж.* Теория сетей Петри и моделирование систем. – М.: Мир, 1984. – 460 с.
5. *Анисимов Н.А., Поступальский П.А.* Верификация коммуникационных протоколов с использованием сетей Петри. – <http://www.fiztech-usa.net/anisimov/papers/MATH-1998.pdf>
6. *Гладун А.Я., Проценко Е.И.* Математические методы анализа для определения показателей производительности корпоративных сетей с использованием стохастических сетей Петри // Проблемы управления и информатики. – 2001. – № 2. – С. 12–30.
7. *Гладун А.Я., Проценко О.И.* Аналіз моделей процедур хендвера в бездротових мережах, базованих на технології АТМ // Наук. зап. Сер. Комп'ютерні науки. Т. 21. – К.: Києво-Могилянська Академія, 2003. – С. 85–95.
8. *Гладун А.Я., Точилін В.В.* Разработка моделей стохастических сетей Петри для анализа и проектирования беспроводных коммуникационных систем // Там же. Т. 27. – 2004. – С. 56–62.
9. *Зайцев Д.А.* Верификация телекоммуникационных протоколов с помощью декомпозиции сетей Петри // Зв'язок. – 2005. – № 1(53). – С. 41–47.
10. *Бандман О.Л.* Проверка корректности сетевых протоколов с помощью сетей Петри // Автоматика и вычислительная техника. – 1986. – № 6. – С. 82–91.
11. *Точилін В.В.* Інтероперабельне середовище проектування та моделювання конструктивних елементів імітаційних моделей // Інформаційні технології в економіці, менеджменті і бізнесі. Проблеми науки, практики та освіти: Зб. наук. пр. – 2010. – Т. 2. – С. 105–108.
12. *OmNet++ Network Simulation Framework.* – <http://www.omnetpp.org/>
13. *GreatSPN home page* – Dipartimento di Informatica – Universit di Torino. – <http://www.di.unito.it/~greatspn/index.html>
14. *Petri Nets Tools and Software.* – <http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/>
15. *SSE: TimeNET.* – <http://www.tu-ilmenau.de/fakia/TimeNET.timenet.0.html>
16. *Reinhard German.* Performance Analysis of Comunication Systems. – New York: John Wiley & Sons, Ltd., 2003. – 538 p.

Поступила 10.06.2013
Тел. для справок: +38 067 135-1872 (Київ)
E-mail: vtochilin@gmail.com
© В.В. Точилін, 2013

Разработка модели проектирования компонентной базы программных средств исследования компьютерной сети на основе интероперабельной среды

Введение. Задача исследования будущей компьютерной сети (КС) на этапе ее проектирования, помощь администраторам КС при штатном и, особенно, в нештатном (аварийном) режимах работы КС есть, без сомнения, насущной и актуальной.

Наиболее распространенным методом решения этой задачи есть метод построения модели КС и дальнейшего исследования ее программно-инструментальными средствами имитационного моделирования [1], такими как *OmNet++* (<http://www.omnetpp.org/>), *ns2* (<http://www.isi.edu/nsnam/ns/>), *OPNET Network Planner* (<http://www.opnet.com/>) и т.д.

Указанные программно-инструментальные средства дают возможность построить модель КС, используя встроенную библиотеку моделей компонентов сети (модели сетевого оборудования, протоколов, каналов связи и др.) Для создания отсутствующих моделей компонентов КС программно-инструментальные средства имеют уникальный программный интерфейс (*API*).

Качество результатов моделирования в значительной мере зависит от точности и качества моделей компонентов КС. На результаты также влияет наличие моделей компонентов КС, моделирующих оборудование и протоколы, применяемые в КС.

Таким образом, для качественного моделирования КС программно-инструментальное обеспечение должно иметь качественную и полную библиотеку моделей компонентов КС.

Однако следует отметить, что создание моделей компонентов КС требует значительных усилий. Процесс создания модели компонента КС, ответственную за работу нового оборудования или протокола, требует глубокого знания принципов построения моделей компонентов КС для конкретного программно-инструментального средства, принципов работы и эксплуатационных параметров описываемого оборудования или протокола, а также навыков программирования.

Ввиду названных трудностей программированием библиотеки компонентов КС, в основном, занимаются корпорации – разработчики программно-инструментального комплекса. В свою очередь это провоцирует значительные задержки с появлением моделей новых протоколов и оборудования, а также высокую цену пользования таким средством.

Возникает актуальная задача быстрого и качественного создания моделей компонентов КС для конкретного программно-инструментального средства.

Автором настоящей статьи предложена среда проектирования, с помощью которой возможно построить необходимые (недостающие) компоненты КС без навыков программирования и изучения взаимосвязи компо-

нентов и программно-инструментального средства имитационного моделирования.

Внедрение данной среды позволит пользователям программно-инструментальных средств имитационного моделирования (администраторам КС, инженерам и исследователям) избежать ожидания, когда разработчик внедрит необходимые компоненты в свои программы и существенно сократить затраты на самостоятельную разработку.

Разработка архитектуры среды проектирования компонентов компьютерных сетей

Исследование принципов построения моделей компонентов КС в разных программно-инструментальных средствах позволило предложить следующие требования к этим средствам имитационного моделирования КС:

- простое и широко известное средство проектирования моделей;
- доступность средств и методов формальной верификации построенной модели;
- интероперабельность построенной модели (способность модели быть использованной как компонент КС во всех программно-инструментальных средствах, соответствующих заданным условиям).

Изучение специальной литературы позволяет сделать обобщающий вывод, что наиболее эффективно для проектирования моделей в рамках предложенных требований может быть применена математическая модель, которая базируется на обобщенных математических стохастических сетях Петри (*GSPN*) [2].

Впервые сети Петри были предложены в 1962 г. для анализа асинхронных параллельных процессов [3, 4].

Как средство моделирования сети Петри имеют множество преимуществ, а именно:

- математический аппарат сетей широко применим и прост для изучения;
- как графическое средство сети Петри можно применять для наглядного отображения процессов, наблюдаемых в модели системы;
- относительно легко можно сделать корректировку модели, построенной на основе сетей Петри;
- для описания параллельных процессов, происходящих в компьютерных сетях, обобщенные стохастические модели Петри наиболее подходят, поскольку модель *GSPN* наиболее адекватна реальной функционирующей системе. Результат – более точные прогнозируемые характеристики поведения проектируемой системы [2, 5];

• существует большое количество методов анализа и верификации модели [6–8], построенной при помощи *GSPN*, базируемых на формальных методах анализа сети Петри. Анализируя такие качества, как достижимость,

ограниченность, активность и другие, а также используя методы формального исследования, можно быть более уверенным в корректности данной модели [5, 9, 10].

Таким образом, применение обобщенных стохастических сетей Петри для построения моделей компонентов КС соответствует первым двум требованиям, предложенным для решения задачи быстрого и качественно создания моделей компонентов КС.

Для достижения требований третьего пункта, автор предлагает создание среды, которая базируется на программном интерфейсе (*API*) для построения компонентов КС программно-инструментального средства и обеспечивает универсальную, независимую от программно-инструментального средства, платформу построения моделей компонентов КС [1, 11].

Разработанная среда получила название *Интероперабельная среда проектирования компонентов компьютерной сети* (рис. 1).

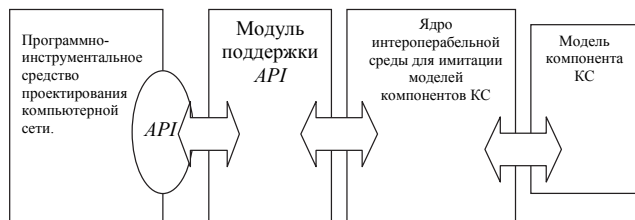


Рис. 1. Функциональная схема среды проектирования

Основным модулем указанной среды построения моделей компонентов КС есть *Ядро интероперабельной среды*. Этот модуль представляет собой Петри-машину, которая во время моделирования, задаваемое программно-инструментальным средством, имитирует модель компонента КС, построенную при помощи сетей Петри.

При транслировании запросов программно-инструментального средства в *Ядро интероперабельной среды* отвечает *Модуль поддержки API* (см. рис. 1), который зависит от конкретного программно-инструментального средства.

Таким образом, после разработки и внедрения в программно-инструментальное средство *Модуля поддержки API* с помощью данного программно-инструментального средства возможно будет проектирование КС при поддержке любых моделей ее компонентов, разработанных по данной технологии.

Внедрение интероперабельной среды в программно-инструментальное средство *OmNet++*

Экспериментальная апробация интероперабельной среды проектирования конструктивных элементов имитационных моделей была проведена для программно-инструментального средства имитационного моделирования *OmNet++* [12].

OmNet++ – модульный программный комплекс моделирования дискретных событий в сети, который можно применять при решении следующих проблем:

- моделировании
 - проводной и беспроводной компьютерной сети;

- сетевого протокола;
- сетей организации очередей;
- мультипроцессоров и других распределенных аппаратных систем;

- валидации архитектуры аппаратной системы;
- оценке аспектов работы сложных систем программ многообеспечения;
- моделировании любой дискретной системы, в которой объекты общаются посредством сообщений.

OmNet++ предлагает инфраструктуру и инструменты для моделирования, где имитационная модель строится на основе ее топологии и моделей ее элементов (например, для компьютерной сети такими элементами есть: каналы связи, оборудование передачи данных, методы доступа, протоколы и пр.).

Руководить моделированием в *OmNet++* можно под разными интерфейсами пользователя. Существует развитый графический интерфейс с динамическим отображением процесса моделирования, который можно использовать для демонстрации работы модели и ее настройки. При помощи интерфейса командной строки можно автоматизировать некоторые аспекты моделирования (последовательное моделирование нескольких моделей, подбор параметров, модулей и пр.).

В *OmNet++* модель состоит из модулей, которые общаются между собой при помощи сообщений (рис. 2). *Простыми модулями* называют модули, написанные языком программирования *C++* с использованием библиотеки классов моделирования. *Простые модули* могут быть сгруппированы для получения *составных модулей*, которые в свою очередь могут быть сгруппированы.

Количество уровней иерархии группировки модулей задается произвольно, что способствует более корректному отображению логических связей фактического объекта исследования модели.

Модули, как правило, имеют параметры, настраивающие их поведение. Составные модули могут передавать параметры для своих подмодулей.

Модель в *OmNet++* называется *сетью* и есть самостоятельным составным модулем.

При проектировании нет разницы между составными и простыми модулями, что позволяет пользователю изменять модули, менять простые модули на составные (и наоборот), не затрагивая модулей более высокого уровня иерархии.

Общение модулей между собой происходит при помощи сообщений, а они имеют как обусловленные данные, например, время моделирования, так и произвольные.

Отправка и получение сообщений в модуле происходит через *ворота (gates)*, которых существует два типа: ворота входа и выхода.

Ворота могут объединяться для передачи сообщений *линией связи* (см. рис. 2).

Составные модули только передают сообщения между своими подмодулями и внешними модулями. В то же вре-

мя простые модули непосредственно получают и преобразуют сообщения.

Сеть

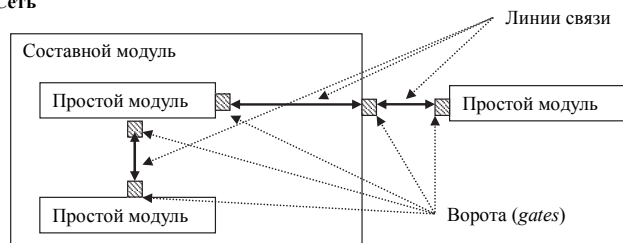


Рис. 2. Простые и составные модули программно-инструментального средства *OmNet++*

Внедрение интероперабельной среды в программно-инструментальное средство *OmNet++* выполнено как отдельный простой модуль под названием *PetryNet*. При помощи параметров этого модуля можно указать модель компонента КС, построенную посредством обобщенных сетей Петри, а также настроить ее поведение.

Последовательность действий при создании и исследовании модели компонента КС с использованием интероперабельной среды

Для построения модели, исследования ее поведения, а также реакции не нее модели КС, необходимо выполнить следующие шаги:

1. Создание модели компонента КС при помощи математического аппарата обобщенных стохастических сетей Петри, применяя для этого любой программный комплекс построения моделей на основе сетей Петри (например, *TimeNet*, *GreatSPN* и др. [13–15]).

2. Исследование и верификация построенной модели с использованием формальных методов изучения сетей Петри.

3. Описание простого модуля *OmNet++* с указанием файла, содержащего описание модели сетей Петри.

4. Построение модели КС в инструментально-программном комплексе *OmNet++* с использованием созданного модуля.

5. Исследование поведения КС и указанных параметров сети Петри при помощи аналитического инструментария *OmNet++*.

6. Исследование поведения модуля компонента КС с использованием файла журнала событий в сети Петри.

Создание и верификация модели сети коммутации пакетов. Рассмотрим упрощенную модель сети коммутации пакетов с установлением соединения [16]. Для построения указанной модели автор использовал программное средство анализа обобщенных сетей Петри *TimeNet* [15] (рис. 3).

TimeNet – графическое интерактивное инструментально-программное средство для моделирования при помощи стохастических сетей Петри. Оно было разработано в *Institut für Technische Informatik* при *Technische Universität Berlin* студентами, программистами и магистрами под руководством профессора *Günter Hommel*.

Разработка началась в 1991 г. Прототипом этого инструмента послужил пакет *DSPNexpress*. *TimeNet* написана при помощи языка программирования C++, а графический интерфейс, начиная с V.4, построен при помощи языка программирования *Java*.

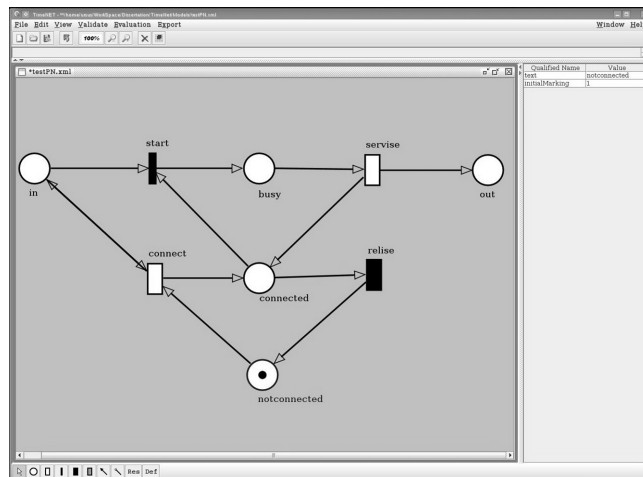


Рис. 3. Модель *GSPN*, ориентированной на связь сети коммутации пакетов

Рассмотрим данную модель.

В модели сети коммутации пакетов с установлением соединения связь между источником и приемником должна состояться перед передачей первого пакета (срабатывает переход *connect*). Для установления связи требуется определенное время. Неэффективно устанавливать связь для передачи одного пакета, поэтому система поддерживает связь до тех пор, пока необходимо передавать пакеты. За отсутствием пакетов на протяжении определенного времени связь прерывается (срабатывает переход *relise*).

Установление связи происходит во время прихода первого пакета для передачи (позиция *in*); в этой позиции находятся маркеры, которые ассоциируются с пакетами, передаваемыми по сети связи. Передача пакетов происходит последовательно, один за другим.

Значение всех позиций и переходов данной модели показаны в таблице.

Значения позиций и переходов в модели, ориентированной на связь сети коммутации пакетов

Позиции	
<i>In</i>	Буфер пакетов, которые должны передаваться по сети
<i>Connected</i>	Связь установлена
<i>Notconnected</i>	Связь не установлена
<i>Busy</i>	Передача одного пакета данных
<i>Out</i>	Буфер пакетов, которые были переданы по сети
Переходы	
<i>Connect</i>	Установление связи
<i>Relise</i>	Разрыв связи
<i>Start</i>	Начало передачи пакета данных
<i>Serve</i>	Передача одного пакета данных

В позицию *in* попадают маркеры, когда программно-инструментальное средство передает пакет *GSPN*-модели. А с позиции *out* маркеры, превращаясь в пакеты, возвращаются в программно-инструментальное средство.

Для верификации построенной модели можно использовать формальные методы исследования сетей Петри. Существует большое количество работ, которые исследуют этот вопрос [4, 5, 10, 16].

Описание модуля ориентированной на связь сети коммутации пакетов. Для использования модели сети коммутации пакетов с установлением связи в программно-инструментальном средстве *OmNet++* построен дочерний модуль *demo PetryNet*. Этот модуль наследует простой модуль интероперабельной среды *PetryNet*. Используя принятые в *OmNet++* языки описания (*NED*, *Network Description*) заданы параметры поведения построенного модуля.

На рис. 4. показано описание модели *demoPetryNet*.

```

simple DemoPetryNet extends PetryNet
{
    parameters:
        // @file метки
        PetryNetFile = "/home/IOS/TimeNet/Models/testPN.xml";
        // Точки входа и выхода модели
        RuleIn_in = "in";
        RuleOut_out = "out";
        // Параметры журнала событий в интероперабельной среде
        eventPetryNet = "stdout"; // (stdout, none, event, none)
        // Позиции метки и как записывать статистику данных
        statisticCollectionPlace = "in out connected";

        @signal[in](type="long");
        @signal[p_out](type="long");
        @signal[p_connected](type="long");

        @statistic[in](title="in"; source="p_in"; record=vector,stats; interpolationmode=none);
        @statistic[out](title="out"; source="p_out"; record=vector,stats; interpolationmode=none);
        @statistic[connected](title="connected"; source="p_connected"; record=vector,stats; interpolationmode=none);

    gates:
        input in;
        output out;
}

```

Рис. 4. Описание модели, ориентированной на связь сети коммутации пакетов в системе *OmNet++*

Во-первых, указан путь к файлу, описывающему модель сети коммутации пакетов (параметр *PetryNetFile*).

Для связи модуля *demoPetryNet* с другими модулями *OmNet++* описаны следующие ворота (секция *gates*):

in – входные, *out* – выходные.

Для соединения данных ворот с позициями сети Петри применяются следующие правила:

RuleIn (название входных ворот) = список входных позиций – когда во входные ворота по линии связи поступает пакет данных – в указанных позициях количество маркеров увеличивается на один;

RuleOut (название выходных ворот) = список выходных позиций – когда во всех указанных позициях появляются маркеры, то из каждой позиции удаляется по одному маркеру, а через указанные выходные ворота выходит пакет данных.

Далее задается способ записи журнала событий, происходящих в сети Петри (параметр *EventPetryNet*).

Возможны следующие варианты:

- *event* – запись журнала в окне эмуляции *OmNet++*;
- *stdout* – вывод на консоль;
- *none* – без вывода журнала событий;

• путь к файлу – запись журнала событий в избранный файл.

Возможно также задать позиции, количество маркеров в которых будет отслежено аналитической частью программно-инструментального средства имитационного моделирования *OmNet++*.

Для этого укажем интероперабельной среде, какие позиции необходимо контролировать (параметр *statisticCollectionPlace*), а также зададим *OmNet++*, каким образом отслеживать эти параметры (директивы *@signal* и *@statistic*).

Построение модели КС в инструментально-программном комплексе *OmNet++* с использованием созданного модуля и ее имитация. После того, как модуль *demoPetryNet* построен, его можно использовать при моделировании КС в программно-инструментальном средстве *OmNet++* (рис. 5).

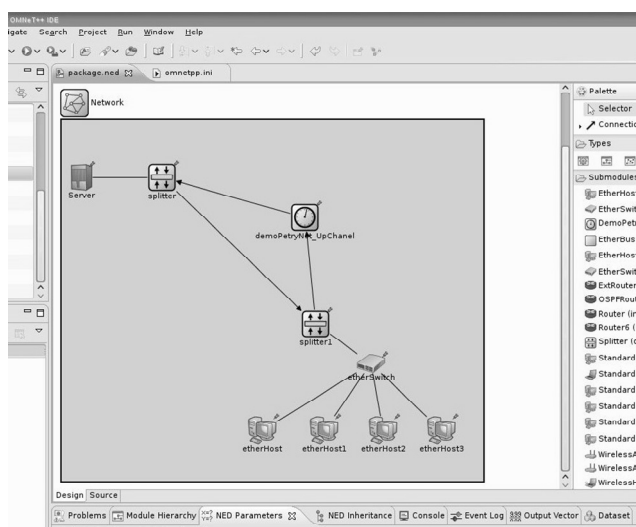


Рис. 5. Топология КС с использованием модуля *demoPetryNet*

Моделируемая КС представляет собой четыре клиентских компьютера (*etherHost* – *etherHost3*), которые при помощи свича (*etherSwitch*) связаны с сервером (*Server*). На канале от свича к серверу (восходящем) установлен модуль сети коммутации пакетов построенной при помощи сетей Петри (*demoPetryNet_UpChannel*). Модули *splitter* и *splitter1* необходимы для разделения восходящего и нисходящего каналов.

Для исследования построенной КС запустим имитацию.

Исследование поведения КС и указанных параметров сети Петри при помощи аналитического инструментария *OmNET++*. Программно-инструментальное средство имитационного моделирования *OmNet++* позволяет исследовать параметры и поведение модели КС (рис. 6).

Интероперабельная среда также позволяет отслеживать поведение маркеров в произвольной (указанной при описании модели) позиции сети Петри. Динамика маркеров в позиции входного буфера *in* показана на рис. 7.

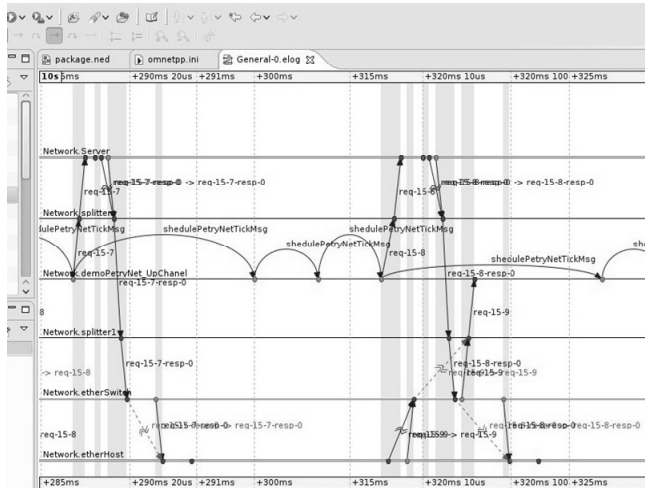


Рис. 6. График поведения моделированной КС

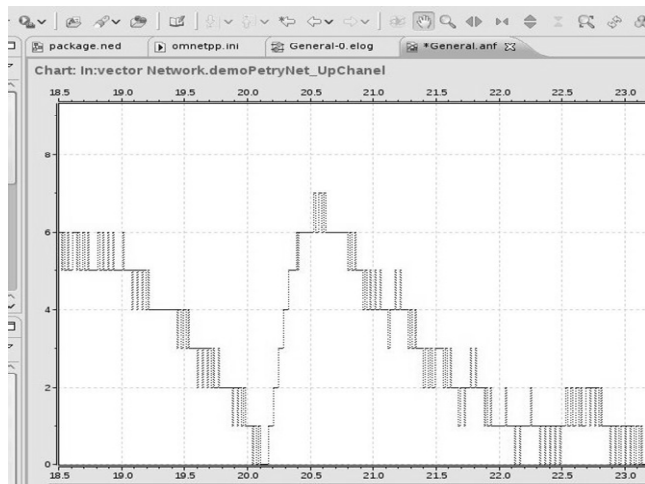


Рис. 7. Динамика изменения количества маркеров в позиции входного буфера *in*

Исследование поведения модуля компонента КС с использованием файла журнала событий в сети Петри. Ядро интероперабельной среды во время имитации модели позволяет создавать журнал событий, происходящих в среде сети Петри. Это обеспечивает контроль поведения сети во время имитации для выявления ошибок в проектировании или параметрах запуска сети Петри. Журнал событий позволяет также анализировать поведение модели на основе сети Петри при влиянии на нее реальной компьютерной сети (рис. 8).

Закключение. Внедрение программно-инструментальных средств имитационного моделирования для проектирования новых и исследования существующих компьютерных сетей непосредственно зависит от качества и актуальности компонентной базы данных программно-инструментальных средств. Поддержка актуальности в

настоящее время требует значительных финансовых затрат от разработчиков таких средств, что также обуславливает их цену.

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	Note:											
3	PL - place											
4	- Number of markers indicated are in the position after the tick											
5	TR - transition											
6	- use the following notation:											
7	(-) passive transition											
8	(>) open transition											
9	(+) active transition											
10	(>) close transition											
11	(>>) open & close transition											
12	simTime TR: connect relise servise start PL: busy connected in notconnected out											
13	(init)							0	0	1	1	0
14	0.01	>	0	0	0	0	0
1049	10.36	.	>	0	0	0	0	0
1062	10.49	.	+	0	0	4	0	0
1063	10.5	.	+	0	0	4	0	0
1064	10.51	.	+	0	0	4	0	0
1065	10.52	.	+	0	0	4	0	0
1066	10.53	.	+	0	0	5	0	0
1067	10.54	.	+	0	0	5	0	0
1068	10.55	.	+	0	0	5	0	0
1069	10.56	.	>	0	0	5	1	0
1070	10.57	.	>	0	0	5	0	0
1071	10.58	.	>	0	1	6	0	0
1072	10.59	>>	.	1	0	5	0	0
1073	10.6	.	.	>	.	.	.	0	0	5	0	0
1074	10.61	.	.	>	.	.	.	0	1	6	0	1
1075	10.62	>>	.	1	0	5	0	0
1076	10.63	.	.	>	.	.	.	0	0	5	0	0
1077	10.64	.	.	>	.	.	.	0	1	5	0	1
1078	10.65	>>	.	1	0	5	0	0
1079	10.66	.	.	>	.	.	.	0	0	5	0	0
1080	10.67	.	.	+	.	.	.	0	0	5	0	0
1081	10.68	0	0	5	0	0

Рис. 8. Журнал событий имитации модели GSPN сети, коммутационный пакетов, открытый для анализа в табличном редакторе

Предложенная интероперабельная среда позволяет создать модели конструктивных элементов КС без углубленных знаний в программировании и знаний особенностей API конкретного программно-инструментального средства имитационного моделирования. Разработчики таких моделей могут осуществлять даже пользовательские программные средства имитационного моделирования.

Интероперабельность предложенной среды позволяет использовать модели, разработанные не только для конкретного программно-инструментального средства имитационного моделирования, но и для всех средств, поддерживающих предложенную среду.

Использование обобщенных сетей Петри позволяет проводить верификацию разработанных моделей компонентов КС, используя формальные методы исследования сетей Петри, что придает уверенности в корректности модели.

Упрощение построения моделей компонентов КС способствует существенному снижению затрат в исследовании КС, когда используется современное сетевое оборудование и протоколы. С другой стороны, интероперабельная среда необходима при разработке новых протоколов и оборудования, поскольку позволяет провести имитацию работы моделей нового оборудования или протокола под нагрузкой, близкой к реальной КС.