

УДК 681.3.06

О.К. Коноплицька, асист., О.В. Коваленко асист.

Кіровоградський національний технічний університет

Взаємодія мобільної платформи J2ME з комунікаційними лініями глобальної мережі

Описано організацію взаємодії мобільної платформи Java 2 Micro Edition (J2ME) комунікаційними лініями глобальної мережі. J2ME надає платформи-залежну функціональність мови Java для портативних пристроїв, дозволяючи широкому класу мобільних бездротових пристроїв використовувати однакові додатки.

комунікаційні лінії, J2ME, протокол, встановлення з'єднання, дейтаграма, сокети, HTTP з'єднання

Комунікаційні лінії забезпечують передачу інформації в глобальну мережу (рис. 1). Для роботи з мережею в J2ME [1-3] використовується спеціальний API Generic Connection Framework (GCF). Підтримка GCF здійснюється на рівні конфігурації (пакет `javax.microedition.io`). Основу GCF складає клас `Connector` і набір інтерфейсів. Клас `Connector` є основою для типів підключення. Інтерфейси визначають типи підтримуваних з'єднань. Розглянемо якими шляхами можна організовувати взаємодію J2ME з лініями глобальної мережі Інтернет.

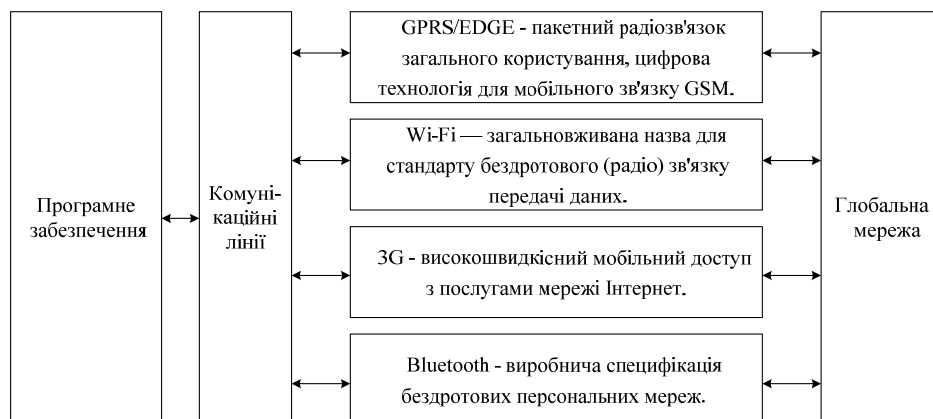


Рисунок 1- Взаємодія комунікаційних ліній з глобальною мережею

1. Наведемо код, що забезпечує найпростіше HTML з'єднання (лістинг 1).

```

HttpConnection MyCon = (HttpConnection)
Connector.open ("http://www.kntu.kr.ua/",
Connector.READ_WRITE, true);

```

Лістинг 1

Клас `Connector` визначає, який тип з'єднання потрібно створити, за допомогою спеціальної URL рядка.

2. В основі технології дейтаграм лежить передача даних без встановлення з'єднання. Дейтаграми передаються в мережу "наосліп", тобто факт доставки листа адресату не перевіряється. У разі якщо зв'язок поганий, дейтаграма цілком може не дійти до адресата. Розглянемо дейтаграму і передамо її за вказаною IP-адресою

(лістинг 2). Дейтаграма з текстом "TEST DATA", яка передається на порт 9001 локального пристрою (телефон передає дейтаграму сам собі).

```
try {
    DatagramConnection dgc = (DatagramConnection)
    Connector.open ("datagram://localhost: 9001");
    try {
        byte [] payload = "TEST DATA". getBytes ();
        Datagram datagram = dgc.newDatagram (payload,
payload.length);
        dgc.send (datagram);
    } Finally { dgc.close ();}
    } Catch (IOException x) {
        x.printStackTrace ();
    }
}
```

Лістинг 2

3. Іншим найпоширенішим типом з'єднання є TCP сонет з'єднання. Передача даних при роботі з сокетом відбувається на основі підключення. Це означає, що відправник і одержувач повинні встановити між собою канал зв'язку для обміну даними, ініціалізувати з'єднання (лістинг 3). Це як дзвінок по телефону. Якщо Ваш друг не зняв трубку, Ви не зможете передати йому інформацію. Використання сокетів увазі гарантовану доставку даних у тому порядку, в якому вони були відправлені.

```
try {
    SocketConnection sc = (SocketConnection)
    Connector.open ("socket://localhost: 9002");
    OutputStream os = null;
    try {
        os = sc.openOutputStream ();
        byte [] data = "TEST DATA". getBytes ();
        os.write (data);
    } Finally {
        sc.close (); os.close ();
    }
} Catch (IOException x) { x.printStackTrace ();}
```

Лістинг 3

SocketConnection встановлює з'єднання з 9002 портом локальної машини. Якщо з'єднання успішно встановлено, OutputStream записує повідомлення в вихідний потік. Для передачі даних сокетам потрібен канал даних. У разі якщо клієнтське ПЗ не запущено і не прослуховує порт, програма поверне помилку.

4. HttpConnection набагато складніше сокетів і дейтаграм. HttpConnection поєднує в собі InputStream і OutputStream. Кожен HttpConnection може відкрити та використовувати лише один InputStream і один OutputStream. Також важливий порядок в якому використовуються потоки. OutputStream повинен використовуватися перед InputStream. Після того як потік даних використаний, з'єднання повинне бути закрито і, якщо це необхідно, створено нове. Ця схема відповідає HTTP парадигмі запит-відповідь.

HTTP з'єднання може перебувати в одному з трьох станів: встановлюється, зв'язано, закрито.

Відразу після відкриття з'єднання HttpConnection знаходиться в стані "встановлюється". На цій стадії можна задати різні параметри, наприклад тип запиту (GET/POST) або налаштування HEAD заголовка (за допомогою методів setRequestMethod () і setRequestProperty ()).

У стан "з'єднане" потік переходить в результаті виклику будь-якого з методів, які призводять до передачі даних на сервер. Після того як потік перейшов у стан "з'єднане" виклик методів setRequestMethod () і setRequestProperty () призведе до виникнення IOException.

Отримання даних за допомогою `HttpConnection` (лістинг 4), встановлюється з'єднання з <http://www.kntu.kr.ua/>. Демонструє зчитування даних `HttpConnection`, оскільки порт не вказаний, підключення відбувається до 80 порту. За замовчуванням для запиту використовується метод `GET`.

```
HttpConnection c = null;
InputStream is = null;
StringBuffer sb = new StringBuffer ();
try {
    c=(HttpConnection)Connector.open("http://www.kntu.kr.ua/",
        Connector.READ_WRITE, true);
    c.setRequestMethod (HttpConnection.GET);
    is = c.openInputStream (); int ch = 0;
    for (int ccnt = 0; ccnt <150; ccnt)
    { ch = is.read ();
      if (ch == -1) {break;}
      sb.append ((char) ch);
    }
}
catch (IOException x){x.printStackTrace ();}
finally {
    try { is.close(); c.close();
        }Catch(IOException x) { x.printStackTrace ();}}
    System.out.println (sb.toString ());
```

Лістинг 4

Як висновок можна сказати, що J2ME це високотехнологічне рішення, яке найкращим чином дозволяє організовувати взаємодію з лініями глобальної мережі Інтернет. Для швидкої перевірки доступу до Інтернет необхідно використовувати найпростіше HTML з'єднання (лістинг 1). При проектуванні складного мережного ПЗ треба використовувати `HttpConnection` (лістинг 4).

Список літератури

1. Вишне夫斯基 В.М Широкополосные беспроводные сети передачи информации / В.М Вишне夫斯基, А.И. Ляхов, С.Л. Портной. – М.: Эко-Трендз, 2005. – 592 с.
2. Горнаков С.Г. Программирование мобильных телефонов на Java 2 Micro Edition / С.Г. Горнаков. – М.: ДМК Пресс, 2004. – 336 с.
3. Пирумян В.В. Платформа программирования J2ME для портативных устройств / В.В. Пирумян Пер. с англ. – М.: КУДИЦ-ОБРАЗ, 2002. – 352 с.

О. Коноплицкая, А. Коваленко

Взаимодействие платформы J2ME с коммуникационными линиями глобальной сети

Описана организация взаимодействия мобильной платформы Java 2 Micro Edition (J2ME) с коммуникационными линиями глобальной сети. J2ME предоставляет платформо-зависимую функциональность языка Java для портативных устройств, позволяя широкому классу мобильных беспроводных устройств использовать одинаковые приложения.

O. Konoplitskaya, A. Kovalenko

Interaction J2ME mobile platform with a global network of communication lines

Describes the organization of interaction of mobile platforms Java 2 Micro Edition (J2ME) with communication lines of the global network. J2ME provides a platform-dependent functionality of the Java language for mobile devices, enabling a broad class of mobile wireless devices use the same applications.

Одержано 04.04.11