Fpt University

FPT POLYTECHNIC

ANDROID NÂNG CAO

BÀI 5: XML PARSER

www.poly.edu.vn

Mục tiêu

- Kết thúc bài học này bạn có khả năng
  - Hiểu rõ về Parser
  - Sử dụng XMLPullParser

NỘI DUNG

- Phần I: Parser
  - Giới thiệu Parser
  - Phân tích Feed
- Phần II: XMLPullParser
  - Khởi tạo Parser
  - Đọc Feed
  - Phân tích XML
  - Bỏ qua các tag không cần thiết
  - Sử dụng XML Data

Fpt University

FPT POLYTECHNIC

BÀI 5: XML PARSER

PHẦN I: PARSER

www.poly.edu.vn

Parser

- Google khuyến cáo bạn nên sử dụng XmlPullParser
- XmlPullParser cung cấp cơ chế hiệu quả và dễ bảo trì khi phân tích XML trên Android

Phân tích Feed

- Bước đầu tiên là phân tích Feed để quyết định trường (field) nào mà bạn quan tâm
- Parser sẽ trích rút dữ liệu các trường cần thiết và bỏ qua cac trường khác
- Chúng ta sẽ tìm hiểu về ví dụ NetworkActivity
- Trong ví dụ NetworkActivity, mỗi post của StackOverflow.com sẽ xuất hiện trong một feed như là một entry tag có chứa một số tag con

Cấu trúc feed của StackOverflow

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
xmlns:creativeCommons="http://backend.userland.com/creativeCommonsRssModule" ...">
<title type="text">newest questions tagged android - Stack Overflow</title>
...
   <entry>
   ...
   </entry
   <entry>
      <id>http://stackoverflow.com/q/9439999</id>
      <re:rank scheme="http://stackoverflow.com">0</re:rank>
      <title type="text">Where is my data file?</title>
      <category scheme="http://stackoverflow.com/feeds/tag?tagnames=android&sort=newest/tags"
term="android"/>
      <category scheme="http://stackoverflow.com/feeds/tag?tagnames=android&sort=newest/tags"
term="file"/>
      <author>
         <name>cliff2310</name>
         <uri>http://stackoverflow.com/users/1128925</uri>
      </author>
      <link rel="alternate" href="http://stackoverflow.com/questions/9439999/where-is-my-data-file" />
      <published>2012-02-25T00:30:54Z</published>
      <updated>2012-02-25T00:30:54Z</updated>
      <summary type="html">
         <p>I have an Application that requires a data file...</p>

      </summary>
   </entry>
   <entry>
   ...
   </entry>
...
</feed>
```

Fpt University

FPT POLYTECHNIC

BÀI 5: XML PARSER

PHẦN II: XMLPULLPARSER

www.poly.edu.vn

Khởi tạo Parser

- Bước tiếp theo là khởi tạo Parser và bắt đầu quá trình phân tích
- Trong đoạn code ví dụ sau, parser được khởi tạo mà không xử lý namespace, và sử dụng InputStream là đầu vào.
- Parser bắt đầu tiến trình phân tích bằng lời gọi nextTag() và gọi phương thức readFeed (dùng để trích xuất và xử lý dữ liệu mà ứng dụng quan tâm)

XMLPULLPARSER

```
/**
 * This class parses XML feeds from stackoverflow.com.
 * Given an InputStream representation of a feed, it returns a List of entries,
 * where each list element represents a single entry (post) in the XML feed.
 */
public class StackOverflowXmlParser {
    private static final String ns = null;

    // We don't use namespaces

    public List<Entry> parse(InputStream in) throws XmlPullParserException, IOException {
        try {
            XmlPullParser parser = Xml.newPullParser();
            parser.setFeature(XmlPullParser.FEATURE_PROCESS_NAMESPACES, false);
            parser.setInput(in, null);
            parser.nextTag();
            return readFeed(parser);
        } finally {
            in.close();
        }
    }
```

Đọc Feed

- Phương thức readFeed thực hiện công việc xử lý feed
- Tìm kiếm các element có tag là "entry" như là đầu vào của quá trình xử lý đệ quy feed
- Nếu tag không có **entry** tag, nó sẽ bỏ qua
- Khi toàn bộ feed đã được xử lý đệ quy, trả lại List chứa các entry (bao gồm cả các thành phần dữ liệu) được trích xuất từ feed
- Cuối cùng List được trả lại cho parser

XMLPULLPARSER

```
private List<Entry> readFeed(XmlPullParser parser) throws XmlPullParserException, IOException {
    List<Entry> entries = new ArrayList<Entry>();

    parser.require(XmlPullParser.START_TAG, ns, "feed");
    while (parser.next() != XmlPullParser.END_TAG) {
        if (parser.getEventType() != XmlPullParser.START_TAG) {
            continue;
        }
        String name = parser.getName();
        // Starts by looking for the entry tag
        if (name.equals("entry")) {
            entries.add(readEntry(parser));
        } else {
            skip(parser);
        }
    }
    return entries;
}
```

Phân tích XML

- Như miêu tả trong phần phân tích Feed, bạn phải xác định các tag bạn muốn phân tích. Trong ví dụ của chúng ta, bạn phải trích xuất dữ liệu cho các tag entry và các tag con như title, link, và summary
- Tạo các phương thức sau:
  - Phương thức read cho các tag mà bạn quan tâm. Ví dụ readEntry(), readTitle()
  - Parser đọc tag từ input stream. Khi parser gặp tag tên là entry, titlte, summary, nó sẽ gọi các phương thức tương ứng. Nếu không sẽ bỏ qua tag này

Phân tích XML

- Đối với tag tittle và summary, parser gọi readText().
- Phương thức này trích xuất dữ liệu của các tag này bằng lời gọi parser.getText()

```
// Processes title tags in the feed.
private String readTitle(XmlPullParser parser) throws IOException, XmlPullParserException {
    parser.require(XmlPullParser.START_TAG, ns, "title");
    String title = readText(parser);
    parser.require(XmlPullParser.END_TAG, ns, "title");
    return title;
}
```

Phân tích XML

- Đối với tag link, parser trích xuất dữ liệu của link bằng cách kiểm tra xem link có phải link cần quan tâm không
- Sử dụng parser.getAttributeValue() để trích xuất dữ liệu của link

```
// Processes link tags in the feed.
private String readLink(XmlPullParser parser) throws IOException, XmlPullParserException {
    String link = "";
    parser.require(XmlPullParser.START_TAG, ns, "link");
    String tag = parser.getName();
    String relType = parser.getAttributeValue(null, "rel");
    if (tag.equals("link")) {
        if (relType.equals("alternate")) {
            link = parser.getAttributeValue(null, "href");
            parser.nextTag();
        }
    }
    parser.require(XmlPullParser.END_TAG, ns, "link");
    return link;
}
```

- Đối với tag entry, parser gọi readEntry(). Phương thức này sẽ phân tích các tag con của entry và trả lại đối tượng Entry với các dữ liệu thành viên là tittle, link, summary

```
// Parses the contents of an entry. If it encounters a title, summary, or link tag, hands them
// off
// to their respective &quot;read&quot; methods for processing. Otherwise, skips the tag.
private Entry readEntry(XmlPullParser parser) throws XmlPullParserException, IOException {
    parser.require(XmlPullParser.START_TAG, ns, "entry");
    String title = null;
    String summary = null;
    String link = null;
    while (parser.next() != XmlPullParser.END_TAG) {
        if (parser.getEventType() != XmlPullParser.START_TAG) {
            continue;
        }
        String name = parser.getName();
        if (name.equals("title")) {
            title = readTitle(parser);
        } else if (name.equals("summary")) {
            summary = readSummary(parser);
        } else if (name.equals("link")) {
            link = readLink(parser);
        } else {
            skip(parser);
        }
    }
    return new Entry(title, summary, link);
}
```

Phân tích dữ liệu XML

- Ví dụ phân tích dữ liệu XML sử dụng AsyncTask
- AsyncTask không xử lý dữ liệu ở main thread
- Khi kết thúc xử lý, ứng dụng cập nhật UI của main activity

Sử dụng dữ liệu XML

```
public class NetworkActivity extends Activity {
    public static final String WIFI = "Wi-Fi";
    public static final String ANY = "Any";
    private static final String URL =
            "http://stackoverflow.com/feeds/tag?tagnames=android&sort=newest";
    // Whether there is a Wi-Fi connection.
    private static boolean wifiConnected = false;
    // Whether there is a mobile connection.
    private static boolean mobileConnected = false;
    // Whether the display should be refreshed.
    public static boolean refreshDisplay = true;
    // The user's current network preference setting.
    public static String sPref = null;
    // The BroadcastReceiver that tracks network connectivity changes.
    private NetworkReceiver receiver = new NetworkReceiver();
    // Uses AsyncTask subclass to download the XML feed from stackoverflow.com.
    // This avoids UI lock up. To prevent network operations from
    // causing a delay that results in a poor user experience, always perform
    // network operations on a separate thread from the UI.
    private void loadPage() {
        if (((sPref.equals(ANY)) && (wifiConnected || mobileConnected))
                || ((sPref.equals(WIFI)) && (wifiConnected))) {
            // AsyncTask subclass
            new DownloadXmlTask().execute(URL);
        } else {
            showErrorPage();
        }
    }
```

Lớp DownloadXmlTask

- Lớp DownloadXmlTask là lớp con của Asynctask
- Lớp này gồm có hai phương thức sau:
  - doInBackground():thi hành phương thức loadXmlFromNetwork(). Nó truyền tham số feed URL. Phương thức loadXmlFromNetwork() xử lý dữ liệu feed. Khi kết thúc trả lại xâu kết quả
  - onPostExecute() trả lại xâu kết quả và hiển thị trên UI

XMLPULLPARSER

```
// Implementation of AsyncTask used to download XML feed from stackoverflow.com.
private class DownloadXmlTask extends AsyncTask<String, Void, String> {

    @Override
    protected String doInBackground(String... urls) {
        try {
            return loadXmlFromNetwork(urls[0]);
        } catch (IOException e) {
            return getResources().getString(R.string.connection_error);
        } catch (XmlPullParserException e) {
            return getResources().getString(R.string.xml_error);
        }
    }

    @Override
    protected void onPostExecute(String result) {
        setContentView(R.layout.main);
        // Displays the HTML string in the UI via a WebView
        WebView myWebView = (WebView) findViewById(R.id.webview);
        myWebView.loadData(result, "text/html", null);
    }
}
```


DEMO
Parser XML

BÀI TẬP NHÓM

- Xem xét XML của trang RSS vnexpress.net
- Tiến hành parser XML

TỔNG KẾT NỘI DUNG BÀI HỌC

- Phần I: Parser
  - Giới thiệu Parser
  - Phân tích Feed
- Phần II: XMLPullParser
  - Khởi tạo Parser
  - Đọc Feed
  - Phân tích XML
  - Bỏ qua các tag không cần thiết
  - Sử dụng XML Data


TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
LUẬT KINH TẾ
LUẬT KINH TẾ
KỸ THUẬT SOẠN THẢO VÀ TRÌNH BÀY VĂN BẢN
TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
Cảm ơn