Emergent Social Networking using Atom

James Tyson

July 6, 2006

Contents

List of Figures

1   About this document

1.1   Copyright

This document is copyright 2006 James Tyson and licensed under the terms of the Creative Commons Attribution license.

The authoritive source of this document can be located at helicopter.geek.nz/xesn

2   Emergent Social Networking Overview

XML Emergent Social Networking (XESN) is a method for providing "friends list" for without using a centralised webservice such as MySpace, Facebook, Orkut, etc.

XESN uses features latent within the Atom standard of XML based web content syndication to allow for content management systems implementing "blogging" systems to easily implement social networking features in an emergent[1] fashion.

While it is not the authors intention for XESN to explicitly compete with microformats such as XFN[2] there is bound to be some overlap in capabilities. XESN takes advantage of XFN where present as an additional strategy for automatic discovery of peers.

The main properties of XESN are as follows:

Atom is used as the main format for syndication, however RSS can be accomodated with a reduced featureset.

Peers are people or sites to which you are related in some way.

Friends are first-order peers whom you are especially related.

Emergence meaning that discovery of Peers is primarily automatic, although implementations should allow users to perform functions such as add, remove and block Peers.

Implementation agnostic XESN is an open specification and non-reference implementations are encouraged.

---

[1] emergent in this context is taken to mean using a combination of automatic discovery and optional user configuration.

[2] XHTML Friends Network

3   XESN and Atom

XESN co-mingles with Atom as several additional tags in the XESN namespace ("http://helicopter.geek.nz/xesn"). So the first sign that an Atom feed contains additional XESN information is the addition of an xmlns declaration for XESN in the root element, like so:

```
<feed xmlns="http://www.w3.org/2005/Atom"
  xmlns:xesn="http://helicopter.geek.nz/xesn">
```

All additional XESN elements are placed within Atom's author and contributor elements, which can be set per feed or per entry. It is worth noting that if an author or contributor has a large number of friends it might be wise to make sure that the XESN information is not repeated in every author or contributor element.

XESN defines the following elements:

link A link to a Peer's Atom or RSS feed or a link to the Author's avatar (personal icon).

alias A short nick-name to be used instead of the Author's name in friends lists.

description A sentence or two about the Author.

3.1   Link Element

XESN recognises two types of Link elements, both modelled on XHTML's Link element. The first is a link to a Peer's Atom or RSS feed.

3.1.1   Linking a Peer relationship

```
<xesn:link
  rel="peer friend"
  discovery="configured"
  href="http://first.peer.org/feed"
  type="application/xml+atom" />
```

As you can see from the example above a Peer link uses several attributes:

rel The must contain at least "peer", but also can provide any of the rel values present in XFN (such as; "friend", "met", "neighbor" or "muse"). Only "peer", "friend" and "me" have any special significance for XESN, however implementations may make use of this additional information.

discovery How this Peer relationship was realised, either "configured" if the relationship was manually entered by the Author, or "automatic" if the Peer was discovered using one of XESN's automatic discovery methods.

href The URI of the Peer's feed.

type The MIME type of the Peer's feed. Currently only "application/xml+atom" and "application/xml+rss" are sensible.

An Atom author element can contain as many Peer links as required and order is not important.

5

3.1.2   Linking an avatar icon

```
<xesn:link
  rel="avatar"
  href="http://helicopter.geek.nz/files/movieunix.gif"
  type="image/gif" />
```

As the example above displays linking to an avatar image is done in much the same way as linking to a Peer's feed, except that the "rel" attribute must be set to "avatar", "href" should point to the location of a valid image in GIF, JPEG, PNG or SVG format and "type" should be the corresponding MIME type.

An Atom author element should only contain a single Avatar link. Implementations are encouraged to download and resize Avatar images in order to match local style rules, therefore images should be licensed appropriately.

3.1.3   User Aliases

```
<xesn:alias>jamesotron</xesn:alias>
```

XESN's alias element should contain a short nickname for the author. Any number of alias elements may be provided, however implementations should only choose one to display.

3.1.4   Mottos and descriptions

```
<xesn:description>I don't eat humble pie... it's not vegan.</xesn:description>
```

The description element should provide a short description or motto about the Author. Any number of description elements may be provided, however implementations should probably only choose one to display.

3.2   Example Atom Feed

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
  xmlns:xesn="http://helicopter.geek.nz/xesn"
  xmlns:html="http://www.w3.org/1999/xhtml">

  <title>helicopter.geek.nz</title>
  <link href="http://helicopter.geek.nz/"/>
  <link
    rel="self"
    href="http://helicopter.geek.nz/feed"
    type="application/xml+atom" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>James Tyson</name>
    <!-- begin XESN extensions -->
    <xesn:alias>jamesotron</xesn:alias>
    <xesn:alias>jnt</xesn:alias>
    <xesn:description>I don't eat humble pie... it's not vegan.</xesn:description>
```

```
    <xesn:link
      rel="avatar"
      href="http://helcopter.geek.nz/files/movieunix.gif"
      type="image/gif" />
    <xesn:link
      rel="peer friend met colleague"
      discovery="configured"
      href="http://hww3.riverweb.com/space/start/feed"
      type="application/xml+rss" />
    <xesn:link
      rel="peer"
      discovery="automatic"
      href="http://second.peer.org/feed"
      type="application/xml+atom" />
    <!-- end XESN extensions -->
  </author>
  <contributor>
    <name>Bill Welliver</name>
    <!-- begin XESN extensions -->
    <xesn:alias>hww3</xesn:alias>
    <xesn:description>electronic.alchemy</xesn:description>
    <xesn:link
      rel="peer me"
      discovery="configured"
      href="http://hww3.riverweb.com/space/start/feed"
      type="application/xml+rss" />
    <!-- end XESN extensions -->
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>

  <entry>
    <title>Atomic powered social networking!</title>
    <link href="http://helicopter.geek.nz/xesn" />
    <id>http://helicopter.geek.nz/xesn</id>
    <updated>2003-12-13T18:30:02Z</updated>
    <summary type="html">
      <html:div>
XESN is <html:em>amazing</html:em>!
      </html:div>
    </summary>
  </entry>

</feed>
```

4   Implementation

4.1   Reference Implementation

The XESN reference implementation will be added to the open-source bliki application FinScribe.

4.2   XESN Participation

In order to participate in an XESN cloud the minimum any implementation must provide is a mechanism for users to provide "configured" Friends and Peers which are published in ther ATOM feeds. However whenever was implementing the minimum requirement any fun?

What makes XESN different to other standards is it's ability to automatically discover peers using existing technologies. Suggested methods include:

HTML Anchors Parsing content at publish time looking for links to remote sites.

HTTP Referrers Checking referring sites.

TrackBacks Checking TrackBack (and Pingback) URIs.

Each of these methods is documented in the *Discovery* section of this document.

4.3   Discovery

This section describes how automatic discovery of Peers takes place in XESN. One of the problems with autmatic systems such as XESN that allow remote users to essentially post content to a site is the potential for abuse. One need only look at ones TrackBack logs to highlight this. This document will suggest techniques for minimising potential problems caused by abuse of the XESN system.

4.3.1   HTML Anchors

XESN suggests a method similar to the following to locate Peers in locally published content:

1. Parse out all XHTML link and anchor elements from content at posting time. This can be achieved several ways, but the simples of which is to use an XPath query such as "`//a|link`".

2. Send an HTTP HEAD request for the linked URI.

3. It the request is successful and the remote MIME type is "text/html", "application/xml+xhtml", "application/xml+atom" or "application/xml+rss" then request the URI's contents.

4. If the MIME type is "text/html" or "application/xml+xhtml" (ie an XHTML document) then parse out any link elements in the XHTML head which link to an Atom or RSS feed for the site (XPath: "`/html/head/link[@rel='alternate' and (@type='application/xml+atom' or @type='appl`
   If so send a new request for the Atom or RSS feed.

5. Add the URI for the Atom or RSS feed to the list of Peers. If the original link contains the XFN "rel" attribute "friend" then automatically promote the Peer to Friend status.

4.3.2   HTTP Referrers

4.3.3   TrackBacks

4.4   Using XESN