

MicroContent is Everywhere

Presentation MicroLearning Conference,

Innsbruck, 23 June 2005

version 0.3 (June 22, 2005)

Introduction

With the advent of Internet, publishing has become accessible to everyone. People have been creating and gathering content and made this content available to everyone in the world. This has started with Usenet, FTP and Email. With Usenet users could start a discussion and comment on an existing discussion. Each of these, an Email, a file, a discussion item can be seen as a MicroContent. Then came the web and the users temporarily moved away from MicroContent and started to create home-pages and whole sites. This introduced a lot of complexity. I refer to web-pages and -sites as MacroContent. MacroContent enfolds MicroContent.

Unfortunately creating and maintaining MacroContent is too hard for most people. People seem to be much better in producing MicroContent, such as small thoughts, items in discussions, comments, bookmarks, etc. Luckily the "blog" entered the scene. This allowed people to write and publish such small thoughts episodically. And there was no longer a need for maintenance. Each blog entry, consisting just of a title and a description, is automatically merged into a web-page and this is made available to everyone. This blogging or journaling has become extremely popular. One of the blog-aggregators, Technorati, is now tracking some 11 million blogs (June 10, 2005) and I have seen reports of half a billion blog-items out there.

Slowly we are starting to get into a next phase. People realise that it is not just thoughts that they are publishing, but reviews, comments on other blog-entries, announcement of events, recipes, interesting sites, records of a golf run, books they keep, images they have taken, places they have been to, etc,

etc. Items contain links to other Items, Items have structure. We are moving to Structured MicroContent.

MicroContent Definition

In order to record Structured MicroContent, just a *title* or *description* is no longer enough to describe everything. Other fields are needed. And this can be any field, such as a *link*, a *rating*, a *location*, an *image*, *tags*, etc, etc. Your imagination can be endless. Maybe we should no longer talk about journals, but about simple single record databases.

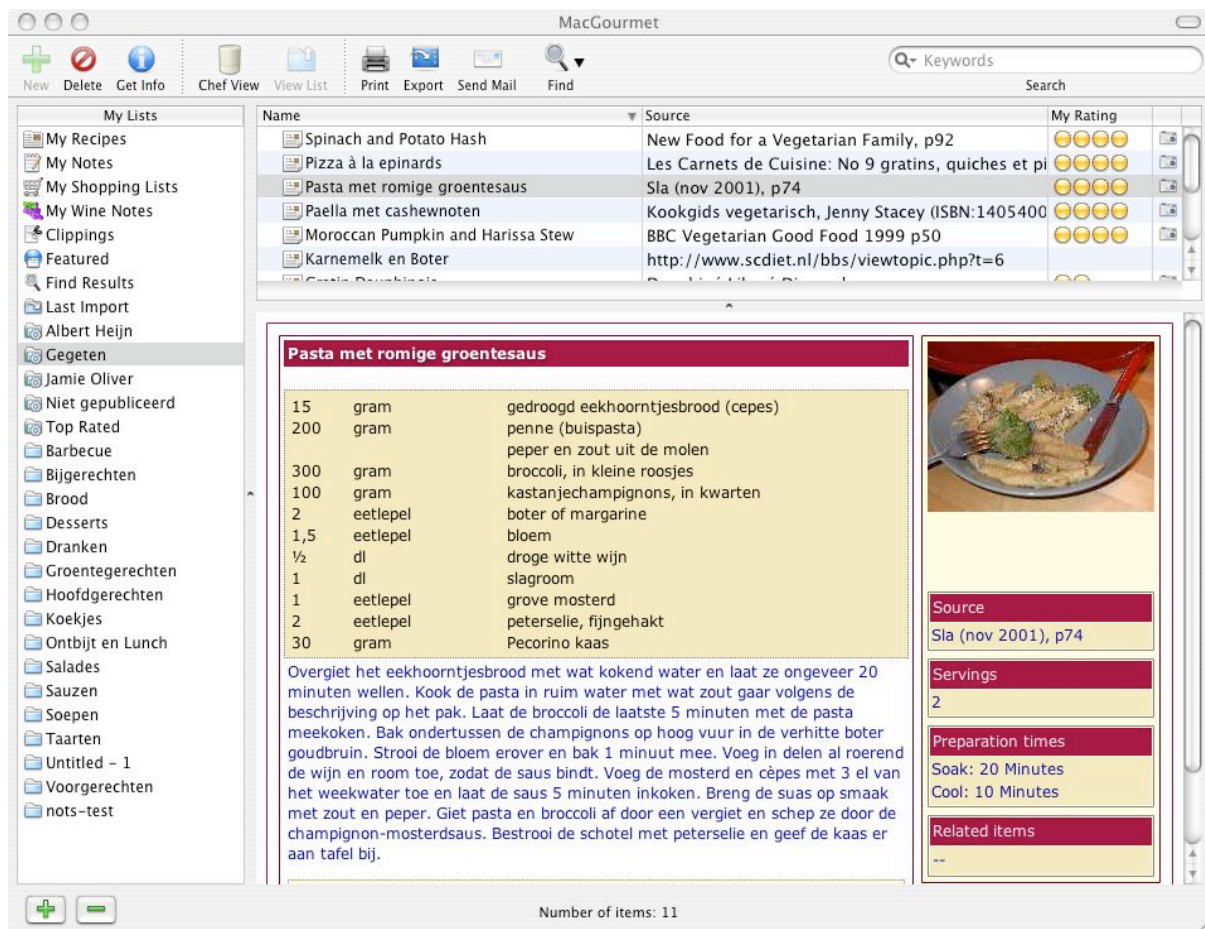
This brings us to question what MicroContent really is. Unfortunately there is not a good definition around. The best way to define MicroContent is by showing examples. However some attributes of MicroContent start to become visible. This list is based on what I see and on what I would like to see:

- **small** - this is the original defining feature of MicroContent. We see however that items get larger and larger in terms of Bytes. Instead of smallness, maybe one should talk about focus. MicroContent (usually) has a single subject, it is focussed;
- **self-contained** - MicroContent can live on its own. It does not need context. This implies that it can be repurposed in all kind of ways. I am not sure whether this attribute excludes any MicroContent type. But in some MicroContent types the relation, the thread, between individual Items is very strong. An example of such threading is found in discussions;
- **addressable** - each piece of MicroContent must have a unique permalink (url or uri). That allows for syndication in all kind of forms. Other authors can thus copy your MicroContent Item and repurpose it in some way. Note that this attribute seems to exclude comments in blog-systems. I think however that this more a relic of current practice, than a desired situation;
- **structured** - MicroContent should have structure. This structure can be extremely minimal (title, permalink, description), but ideally more fields are defined. Structure gives meaning and allows for new kind of aggregation, searching and linking. Structure allows for combining MicroContent on a field-level, which I call the MicroWeb (or the loosely coupled database);
- **flexible** - MicroContent should not be rigid in it's structure. The user should be able to add or leave out fields at will. Flexibility also implies that there shouldn't be any rigid schema's underlying any MicroContent type;
- **single** - it should be possible to make a copy of a MicroContent item with all the structure included. This implies that each MicroContent item should have it's own container, file or uri. Another approach is to add the required mark-up for a MicroContent item in the MacroContent contained. Through smart parsing this allows for extracting the MicroContent item. Personally I prefer the former (single) solution as it is much cleaner;
- **computer data** - maybe this is a sine-qua-non: we talk here about computer data, which also includes audio, video, etc.;

MicroContent Client

It seems that MicroContent is something new. But it is not! People have been defining MicroContent since the Personal Computer came into their homes. People made records of their stamp or video collection, etc with database applications. The major new aspects that Internet now adds is publishing and interconnection. On Internet structured MicroContent is however still in it's infancy. So I looked at applications, which support MicroContent in order to understand MicroContent. I call these MicroContent clients. I found many such MicroContent clients running under MacOS-X, supporting up to 15 different MicroContent types. Each MicroContent type defines a different structure.

A MicroContent client supports a user in creating, managing, viewing and publishing MicroContent items. It turns out that one can define a generic set of functions and a common interface for a MicroContent client. This is best demonstrated by a familiar MicroContent type: a recipe. The application MacGourmet by Advenio (advenio.com) serves as a good example.



Picture 1: MacGourmet

The picture presents the primary interface for this client. On the lower right one sees the View-pane. This is the part of the window that shows the actual recipe, the Microcontent Item itself. One sees a list of ingredients, the method to prepare the recipe, an image, etc. Some MicroContent clients allow the user to set the style through a style-sheet.

Above the View-pane one sees the Items-pane. The Items-pane gives an overview of the recipes, the MicroContent Items. The Items-pane has a tabular format and shows some data of each recipe, such as the title, the source, a rating and whether an image is associated. Just having an Items-pane is normally not enough to manage MicroContent Items. The number of Items in a library can easily grow into the hundreds. Finding a specific Item in such a large collection is hard. One needs a facility to create subsets.

These are shown in the Lists-pane. Each List defines a specific set of Items.

Naturally one of the Lists, *My Recipes*, defines the entire catalogue. The other lists can either be *Hand-Picked Lists*, which content is defined by additions made by the user, or, *Smart Lists*, which content is based on the contents of an Item.

Life-cycle: invocation

Also the functions supported by MicroContent clients reveal commonality. And one will find similar commonality in corresponding hosting services. In order to analyse and present these functions I use the service life-cycle model. For any service one can distinguish three life-cycles: the invocation, the provisioning and the development life-cycle.

The invocation life-cycle describes the phases a user goes through when he uses a service. Or in the case of MicroContent what happens when the viewer accesses a MicroContent Item. Multiple life-cycle's are possible, depending on the location of the MicroContent Item (on PC, network or Internet) and the tools of the viewer. The viewer will either use a tool like an Internet enabled MicroContent client or a generic Web-browser. As MicroContent is still hidden with MacroContent containers, such as web-pages, the actual life-cycle is mainly determined by the MacroContent. And the protocol required to fetch the MacroContent container, such as the HTTP-protocol or others defines the invocation life-cycle..

A client with embedded Internet support will let the user link to a MacroContent container within the Lists-pane of the interface. A good example of this function can be found in the iTunes MusicStore (Figure 2).



Picture 2: iTunes Music Store

For the viewer the fact that the MicroContent Items are stored on the Internet is totally transparent. The difference is hard to tell. Naturally Apple added various functions for buying and downloading the Items.

I especially show this example as one normally thinks of news aggregators, where one subscribes to RSS-feeds. As we have just started with integrating Internet-based information into local applications, I expect other novel functions and life-cycles to appear.

Life-cycle: provisioning

The next life-cycle that needs to be discussed is the Provisioning Life-cycle. This is the life-cycle of the provider, the party that makes the MicroContent Item available to viewers. I assume here for simplicity that the author is the publisher as well. The Provisioning Life-cycle consists of the following steps:

1. *imagine* - the first step is imagining a new MicroContent Item by an

author. This step involves deciding firstly on the MicroContent type to be used for the Item. Is a recipe, an address or a blog-item going to be created. And secondly one can focus on the contents of the Item itself. In this phase the author is not hampered by any restrictions imposed by the tools;

2. *create* - this step translates the imagined Item to actual data. The author must select the tools he will use for creation. Depending on the MicroContent type to be created, the user will chose a different client or service. The client/service will present the user with a standard form consisting of multiple fields, which the user can fill. Note that the creativity of the author is limited by the functions of the tools. In this step that is the structure of the MicroContent Item;
3. *publish* - in this step the author moves the newly created MicroContent Item to public space, either on the local network or on Internet. The author has to decide how to publish the MicroContent Item. Will it be a separate file? Will it be part of a web-page? Will it be part of a web-log? Should it be published as a Atom/RSS-feed? As a web-service? The author has to decide on the MacroContent container for his MicroContent Item. Often the possibilities are set by the client builder or service provider;
4. *provision* - in this phase the published Item is continuously kept available for viewers. This requires hosting providers, which can handle the appropriate protocols needed for the Invocation Life-cycles;
5. *change* - the author of an Item should be able to change an Item at will. Just changing typo's, adding comments, etc. Ideally these changes percolate to new, but also past, viewers;
6. *remove* - finally an author should be able to retract an Item. This is a very badly supported function. Most service providers do not allow a user to remove (or change) an Item. And often an Item will be cached somewhere on the Internet and impossible to delete. Once written often means always written. Good service providers adhere to the DataLibre principle;

The provisioning life-cycle I sketched above is only partly supported by clients and services that cater for MicroContent. I expect the situation to improve however, when the interest for structured MicroContent grows.

Life-cycle: development

The Development Life-cycle involves the creation, definition and sharing of new MicroContent types. How this life-cycle should look like is still unclear. At the moment this process is carried out through committees or personal initiative. This implies that the process usually takes a long time, is troubled by adoption and might account for the sparsity of official MicroContent types. A few Content Management Systems (CMS) allow developers to create extra fields upon a basic blog item. Expression Engine by pMachine.com is an example. The complexity of this process makes it however only available to high-end authors (read geeks).

One would like to see a public repository of MicoContent type definitions. Bob Wyman of Structured Blogging started in this direction with the use of XSD-files. And it seems that Bob Reger of reger.com will incorporate this idea into his service. This will open the reger.com service to any MicroContent type. The next step will be that other blog-hosters and aggregators adopt the same idea.

A big part of the problem is the incorporation of these new MicroContent types into the various MacroContent types. Thus one should have a way to put it into web-pages, RSS-feeds, etc. It seem that the usage of XHTML in combination with appropriate CSS class labels is the way to go.

What the end-situation will be and whether we will get a truly open and flexible life-cycle is unclear.

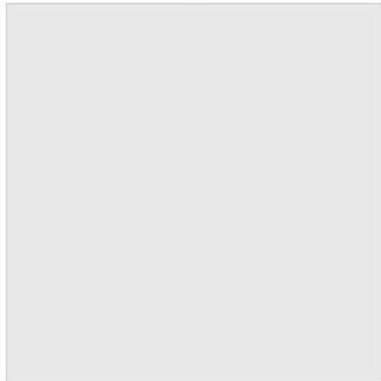
MicroWeb

When talking about the Invocation Life-cycle, I only mentioned viewing by humans. Things will get however more interesting when we think about machine-readable MicroContent. Then services can take MicroContent published by individual users and create something new, something that adds value. Services can aggregate MicroContent of different types and create something new. Alas there are not many examples around. However Technorati and the client DeliciousLibrary offer a glimpse.

Tag: Microcontent

24 posts from 6 blogs match this tag. (What's a Tag?)

Related Tags: [Weblogs](#), [Web/Tech](#), [Memes & Memetics](#), [Social Networks](#), [The Metaweb](#).



Tag my posts with del.icio.us and view tags

6 days ago
I know that this weblog doesn't have too many readers, but I just edited my blog template and...
Stephan Mosel 115 links from 94 sources

"MicroContent is Everywhere" 8 days ago
Dr. Arnaud Leene (MicroContent Musings), who is also going to be at Microlearning 2005, published a first draft of his paper: "MicroContent is..."
Stephan Mosel 115 links from 94 sources

t2 9 days ago
t2
Dave Online 0 links

What's this? This page shows goodies from the web about microcontent. To contribute, just make a post to your blog about microcontent and include the link below. [More Info »](#)
microcontent

Microcontent Photos

from Flickr



[More in Flickr »](#)

Microcontent Posts

from Technorati

Microcontent Musings 21 days ago
Structured Blogging: second look What i mean is that I should have a second look. I read Jon Udell's take on Structured Blogging. He publishes an...
Microcontent Musings 19 links from 12 sources

Microcontent Musings 21 days ago
Replace RSS with Atom Bob Wyman argues that Atom is the next step. RSS is an historical format. Atom has learned form the past (RSS) and improves on...
Microcontent Musings 19 links from 12 sources

Microcontent Musings 21 days ago
Standard Form for MicroContent l.m.orchard replies to

Microcontent Links

from Furl and Delicious

PRYLAR - Nokia 6265 - Mobiltelefon med finesser (svartling@Delicious)

Omniquiti Lathe - 2 Click Video Conversion for Portable Media Devices (svartling@Delicious)

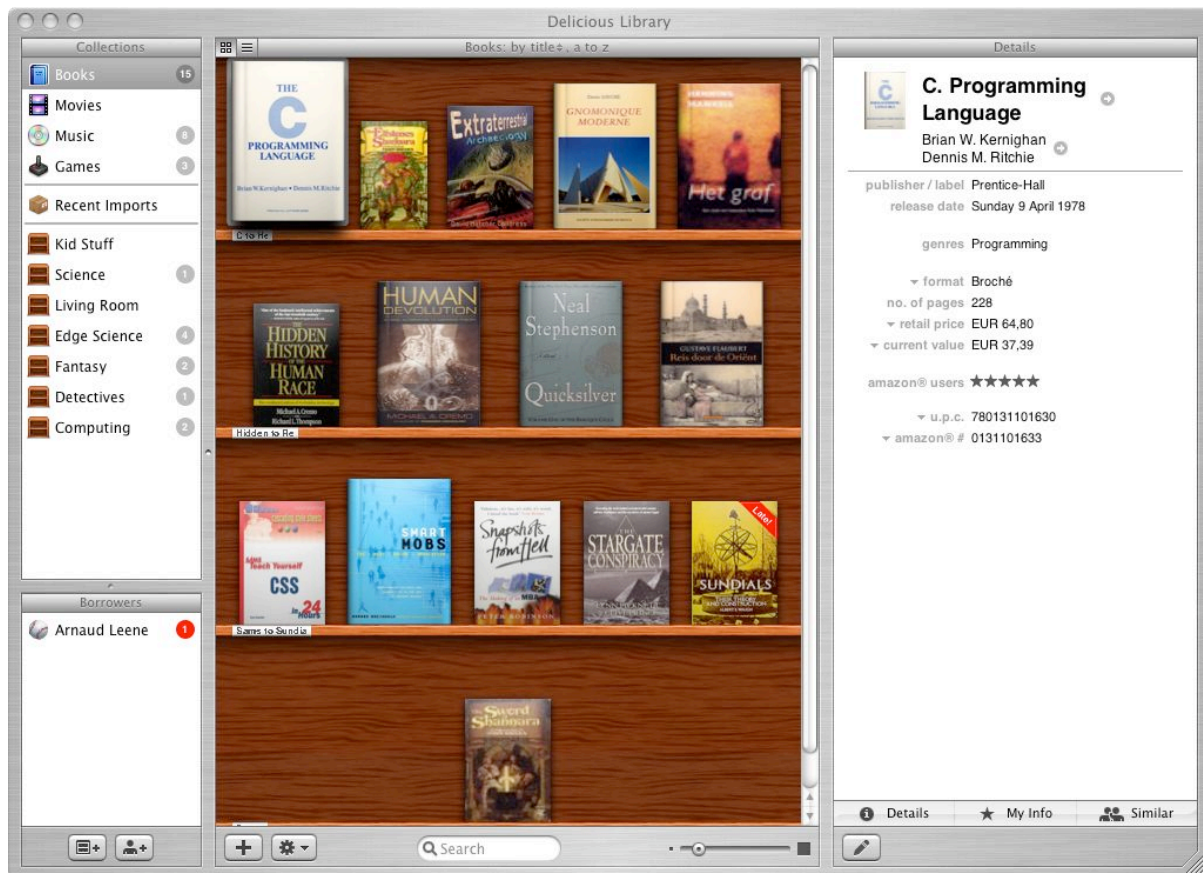
DivX 6 (svartling@Delicious)

InterVideo WinDVD 7 (svartling@Delicious)

Nero Burning ROM 6.6.0.14 (svartling@Delicious)

Picture 3: Technorati screen on the tag MicroContent

A service like Technorati not only has created a search service for weblogs, but now also uses the tag field, i.e. keywords per blog item. Through tags it is able to show the links between MicroContent on Technorati, del.icio.us and Flickr for each tag. Each of these services aggregates and hosts a different MicroContent type. This linking shows another way of connecting Items. And this is just the start. Imagine what happens when we can link other fields in this way. Think of locations, times, products, people, etc. etc.



Picture 4: Delicious Library main screen

The application DeliciousLibrary allows a user to create a database of his books, CD's, Games and DVD's. Most of the data required for this database is taken from Amazon. Thus the user only has to enter the ISBN- or EAN-code and the information on the book is retrieved at Amazon. For me this is an example, which connect two MicroContent items through a single field. I hope we will see more of the connections between MicroContent based on Web-services.

Wrap-up

In this article I indicated that MicroContent is new for Internet, but MicroContent dates already from the invention of the Personal Computer. MicroContent comes in all kind of forms and shapes. It is possible to imagine an infinite number of MicroContent types, only limited by the imagination of the author. Unfortunately the actual support for publishing this limitless imagination is disappointing. Both MicroContent clients and hosting services offer limited functions. Each life-cycle is only partly supported. And what the

next steps, the weaving of the MicroWeb, will be is entirely unclear. However people around the world are dreaming of the new possibilities this MicroWeb offers.

**MicroContent is everywhere and wants to be free,
but be sure you own what you publish.**

Questions for Discussion

- ★ How does MicroContent fit into MacroContent (web-sites, feeds, discussions, etc)?
- ★ Should we define threads as a MicroContent type as well?
- ★ What does a flexible environment for creating, publishing and viewing MicroContent look like?
- ★ Can MicroContent lead to a MicroWeb (Web 2.0)?
- ★ Does MicroContent have a meaning in a Business environment?