

# Dragonfly Ad Management Tool

Author: [Mike Gallant](#) - Product Manager

## Revision History

<a href="#">Author</a>	<a href="#">Date</a>	<a href="#">Rev. No.</a>	<a href="#">Change Description</a>
Mike Gallant	January 4, 2007	0.1	First Draft

 Edit

## Table of Contents

- ↓ [Summary](#)
  - ↓ [Terminology](#)
  - ↓ [Anatomy of a Typical Template](#)
  - ↓ [Workflow](#)
  - ↓ [Workflow Detatils](#)
  - ↓ [User Management \(ACL\)](#)
- ↓ [Target Date:](#)
- ↓ [Dependencies on other Properties/Services:](#)
- ↓ [Measures of Success](#)
- ↓ [Metrics](#)
- ↓ [Target Market/User Benefits](#)
- ↓ [Revenue Impact/Business Rationale:](#)
- ↓ [International](#)
- ↓ [Additional Links](#)

## Summary

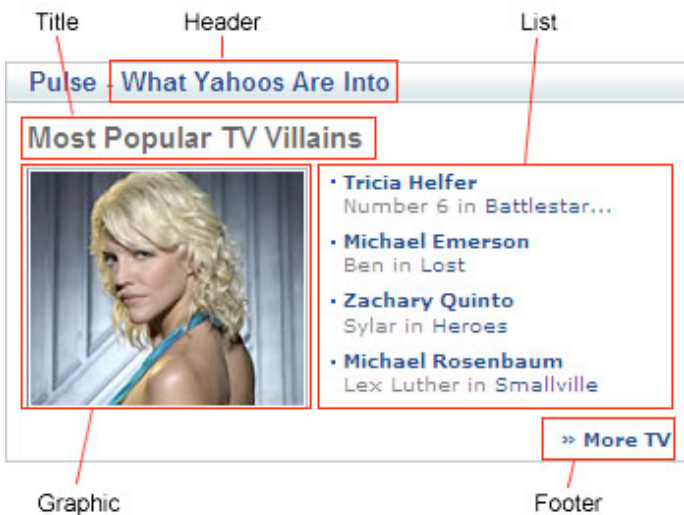
This tool must be accessible and easily usable by various producers inside Yahoo. The interface must be clean and as intuitive as possible, and must present rudimentary help cues where needed. It must enable users to produce ad-type content for any position on any page, of any property, provided the correct templates have been registered in the tool.

## Terminology

- Market – Typically, this is the country in which the content is to be used
- Property – The division within that market's network (Sports, Finance, etc.)
- Page – a section within a property (ex., the Mortgage page within Finance)
- Position – A named section of a page (mantle, Pulse, masthead, etc.)
- Template – An XSLT file that defines the presentation for a specified position
- Content – The XML file that contains the package content
- Package – The combination of an XSLT and its XML content

## Anatomy of a Typical Template

Templates can be described in terms of the content sections:



1. Header – The header is currently static, but may become dynamic (programmable) in the near future. The system should be designed to accommodate this likelihood.
2. Title – The main text of the Pulse package

3. Graphic – Typically a photo
4. List – Text items associated with the graphic
  - o May be bulleted
  - o May include linked or unlinked subtext items for each list item
  - o May be arranged in 1 or 2 columns
5. Footer – A link to further related content

## Workflow

---

The editorial workflow breaks down as follows:

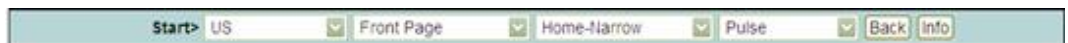
1. Asset Location
2. Content Entry and Proofing
3. Save and Export
4. Administrative functions

## Workflow Details

---

### 1. Asset Location

1. Drill-down to the specific template using the hierarchical select lists – Market, Property, Page, and Position



2. When a position is determined a list of templates for that position is presented, as shown here (note, the user's last drill-down result is displayed on start-up):



3. The user clicks on a template to create a new package for it. This opens the editorial workspace with preview and editing fields.

## 2. Content Entry and Proofing

1. When a template has been selected, the user is presented with the default package for that template in the preview. All user-managed input fields for that template are presented in the editing space below the preview:

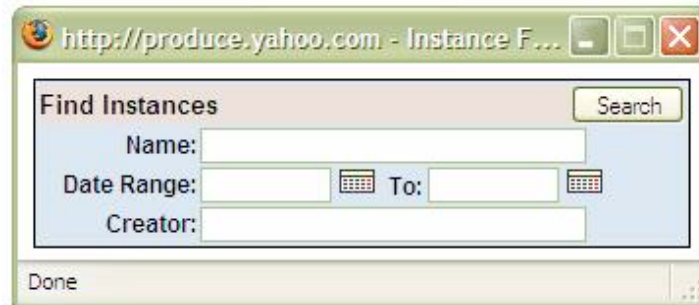


2. Input fields are populated, when editing an existing package.
3. Content entered into the input fields will be validated to ensure all required fields are populated.
4. The user may click “Display HTML” at any time to see the markup for the package as it appears in the preview:



5. Recent Package List – The 10 most recently saved packages are listed, and may be selected to load in the editor (DFly v2)

6. Search – A tool to search for existing packages by date, description, producer, or property.
  1. Search must be a sub-string search – return results based on a provided string, found anywhere within the specified field(s).



### 3. Save and Export

1. File Naming – XML files created with Dragonfly will be associated with an XSL file using the following naming convention: PackageName?.xml
2. Save – Clicking the “Save Package” link opens a dialog featuring the save options:

- Package name – chosen by the user
  - Tags – Any terms that help to find the package later
  - escription – Input by user
  - Tracking – Inserts tracking tags used by Ruby (to be moved to the export function area)
  - Make XML reusable – If yes, the package is listed in the template list to be removed – All packages are reusable
  - Package Status – ???
3. Export Mode – The package code may be exported from the system in several ways, using the controls presented in the “Display HTML” window:
    - Save Code – Saves the markup to the user’s local hard drive

- Email Code – Sends the markup to an email recipient
- FTP – Allows the user to FTP directly to a preset destination (DFly v2).  
If the FTP option is selected, the user will select a particular FTP account from a select list.
  - Each user should be able to access an FTP setup control (mock below), where they can create named FTP connections.

The mockup shows a form titled "FTP Connection" with a purple header. It contains four input fields for "Account Name:", "Host:", "ID (login):", and "Password:". Below this is a section titled "Directories" with a purple header, containing two input fields for "Local :" and "Host:". A green "Submit" button is at the bottom.

- Export Scope – Users can specify whether to send the package XML, XSL, or HTML to that destination (DFly v2).

#### 4. Administrative Functions

1. User Management (ACL) – Create user accounts and set access privileges (see following section)
2. Market, Property, and Page Management – Add/remove markets, properties, and pages to the select lists
  - When a new page is registered, the admin provides Market, Property, and Page names as well as the URL of the page on the network.
  - Controls for Market and Property should be select-or-add type, and Page control always requires new text input.
3. Asset Import and management – Import package files (XML + XSLT), link to pages
  - An imported asset is linked to a particular page using the same drill-down controls that editors use to navigate to a particular page.
  - The admin supplies details on the newly imported asset such as a common name, height and width, and the id of the page div where it “lives”.
4. Export Configuration – Set up needed export parameters, i.e., FTP accounts.  
Note: This might be necessary on a per-user basis, where each user sets up their own export accounts.

## User Management (ACL)

---

ACL provides access to the tool based on Objects and Actions. A hierarchical object tree specifies the object(s) that can be accessed, while an action list specifies the actions that can be taken.

- The object hierarchy starts at the Market level, and progresses to more granular levels, down to the template (or module) level.
- For each object, the permitted activities are listed.
- It may be desirable to establish “roles” that aggregate a set of activities (i.e., the “editor” role may only need Read, and Write permissions, while a “manager” may need these actions plus Push, and an “admin” would have all privileges).
- Such rights may be organized as grid-wise as shown here:

User	Market	Property	Page	Template	Action
User 1	Market 1	Property 1	Page B	ALL	RL
User 1	Market 1	Property 2	Page A	Template X	Manager
User 2	Market 1	Property 2	Page A	Template Y	R
User 3	Market 1	Property 2	Page A	Template Z	R
User 3	Market 2	Property 1	ALL	-	Editor
User 3	Market 2	Property 1	Page C	Template Z	RHPL

## Target Date:

---

- Dependent on budget for contractor - tentatively 30 Nov 07

## Dependencies on other Properties/Services:

---

- No external dependencies

## Measures of Success

---

- Rapid adoption by third parties with improved productivity



## Metrics

---

N/A

## Target Market/User Benefits

---

- Initial users will include: US Pulse production, US Mail managers, LATAM PMs, SEA PMs.
- Users will have a user-friendly WYSIWYG tool to set up and proof templated ad content before scheduling it in various systems.

## Revenue Impact/Business Rationale:

---

- Many ad managers are now producing content by directly editing HTML, which creates avoidable risk and takes more time than necessary. Those who have dedicated tools complain that the tools are outdated, slow, buggy, and unsupported. Tools that might be usable for ad production have fairly steep learning curves (i.e. YAC), so are not suitable for managers who produce content only occasionally. Dragonfly will address all these issues.

## International

---

- Dragonfly was designed from the outset to be used globally and meets our international requirements.

## Additional Links

---

- [Working log for engineering](#)
  - [Bug Table](#)
- 
-

Copyright © 2001-2009 Yahoo! Inc. All Rights Reserved. [Yahoo! Internal Use Only](#)  
Ideas, requests, HowTos regarding YTWiki? [Send feedback](#) [TWikiFAQ](#)  
[Report a problem in Bugzilla about YTWiki](#)