

Mobile Media Asset Management

Executive Summary

Camera phone users face challenges in managing photos on their phones and in navigating the growing array of services and applications that consume these photos. We have built a prototype for a simple phone-based annotation and organization system for photos. Our prototype introduces the concept of a feed to route photos to a variety of destinations over a variety of network protocols, and provides a user interface for sending and tracking photos. The system we propose will allow a user to send over carrier networks or via the user's personal computer at no cost; the feed abstraction will help users understand the variety of delivery options and understand the various economic costs involved.

Motivation

Current trends in adoption of mobile camera phones make mobile media an important area for study:

- Camera-equipped mobile phones rapidly becoming the primary device for digital photography (now out-selling stand-alone digital cameras – see Figure 1 below).
- Important features that may be barriers to the adoption of the camera phone as a primary photography device for users are becoming standard features. New camera phone models are available with three to five megapixel resolution, advanced lenses with optical zoom, and flash.
- Two aspects of mobile camera phones make them particularly interesting for digital media research: they are (or will be) ubiquitous in many nations, and users generally carry them at all times, as opposed to special occasions when they plan on taking photos.

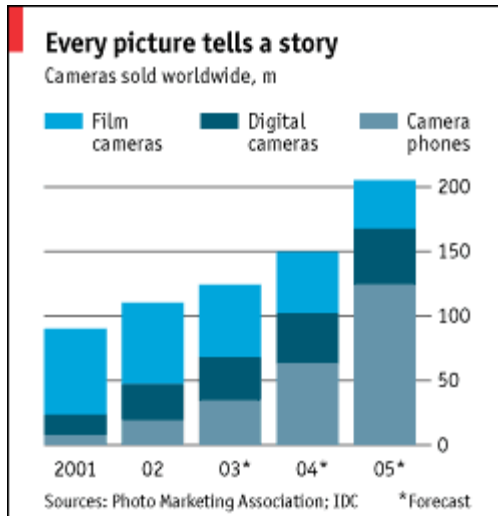


Figure 1: Camera phone trends. Source:

http://www.economist.com/business/displayStory.cfm?story_id=1901134

Problems Addressed

Users of camera phones today face the problem of an ever-growing collection of digital photos spread across various platforms - on their phone, their desktop computers, and online in blogs and other types of web sites. They would like to be able to easily browse, search and share some (but not necessarily all!) of these photos. However, users currently encounter a number of hurdles in accomplishing this, including technical difficulties getting the photos off of the phone to a desired location, financial costs to send photos via network providers, fears about privacy and who may view their photos, lack of interest in annotating photos, and copyright issues.

To help alleviate these problems, users could be greatly benefited by systems which aid users in annotating photos with appropriate metadata, including not only descriptive content, but intentional and usage data and then use this metadata to perform desired actions, enforce usage and privacy policies, and enable richer retrieval. We will design and prototype a mobile phone interface that leverages intentional, descriptive, and contextual metadata to help users appropriately annotate photos and provide the advanced functionalities discussed above.

Task Analysis of Photo Archiving and Sharing

Users engage in the following types of actions and behaviors relating to mobile digital photos:

- **Communication:** sending photos to friends as part of an interactive communication process, via MMS for example. MMS allows mobile phone users to send a photo directly to a friend's phone with a text message.
- **Archiving:** uploading photos from phone to PC for use in a desktop photo album application such as Adobe Photoshop Album or Nokia LifeBlog.
- **Blogging:** posting photos to a public web site.
- **Printing:** sending photos to a print service

Virtually all categories of consumers use cameras -- film, stand-alone digital and camera phones. Camera phone may be in an early-adopter phase today, but is fast becoming a mainstream consumer device that all consumer categories will use. Consumers use camera phones for all the same reasons they use any type of camera – for communication, social connection, memory, or public recognition.

Current System/Approach Description

To inform our prototype design, we looked at several leading photo management systems: Adobe Photoshop Album, Flickr, LifeBlog, and the Mobile Media Metadata (MMM) project at UC Berkeley.

Adobe Photoshop Album

Among the many user-friendly and intuitive features of Adobe Photoshop Album, we were particularly inspired by their use of user-defined tags. Tags such as “Dad,” “Arizona” and “Favorites” can be easily added to a photo or group of photos (see Figure 2) and then those tags can be used to rapidly filter a large photo collection to a few photos of interest (see Figure 3).

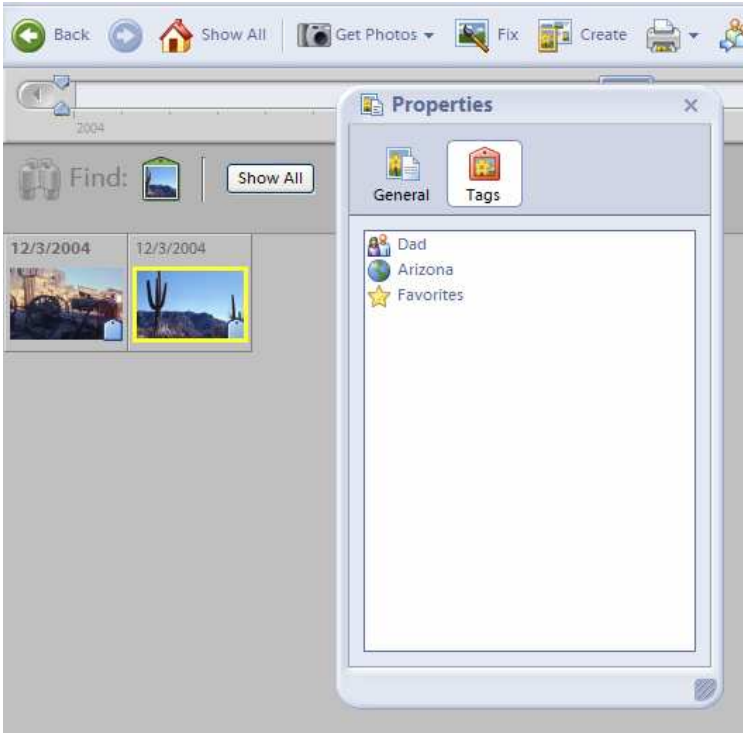


Figure 2: Adobe Photoshop Album – Add Tag Interface

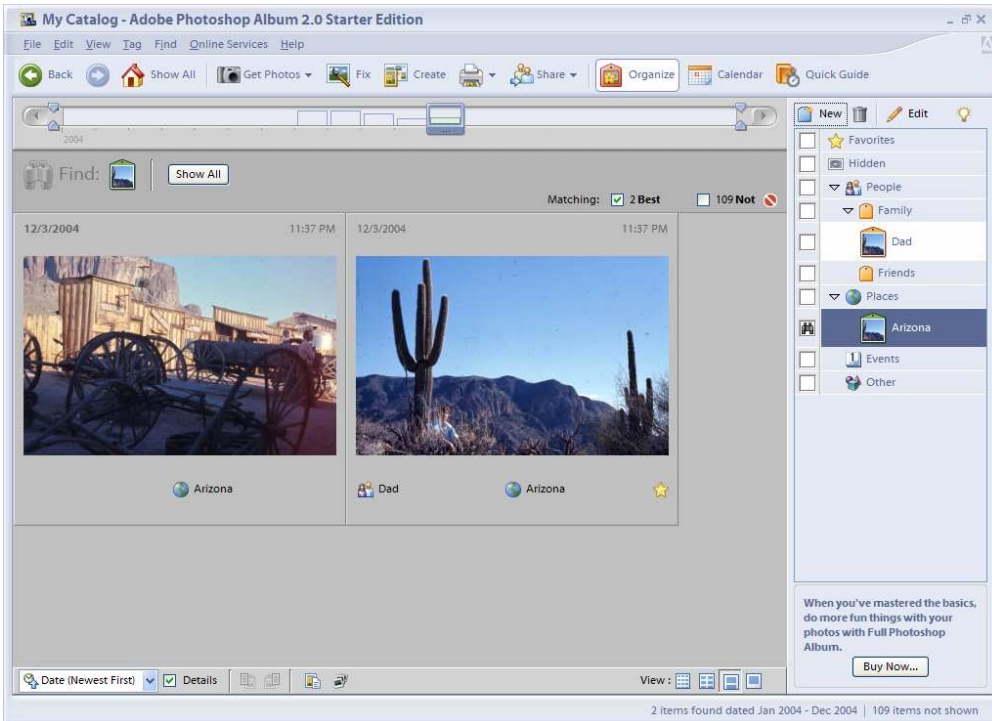


Figure 3: Adobe Photoshop Album – Search by Tag Interface

Flickr

Flickr is a web-based photo sharing service known for its innovative features. We particularly liked its simple privacy control interface (Figure 4) and its use of tags for sharing among all Flickr users (Figures 5, 6 and 7).

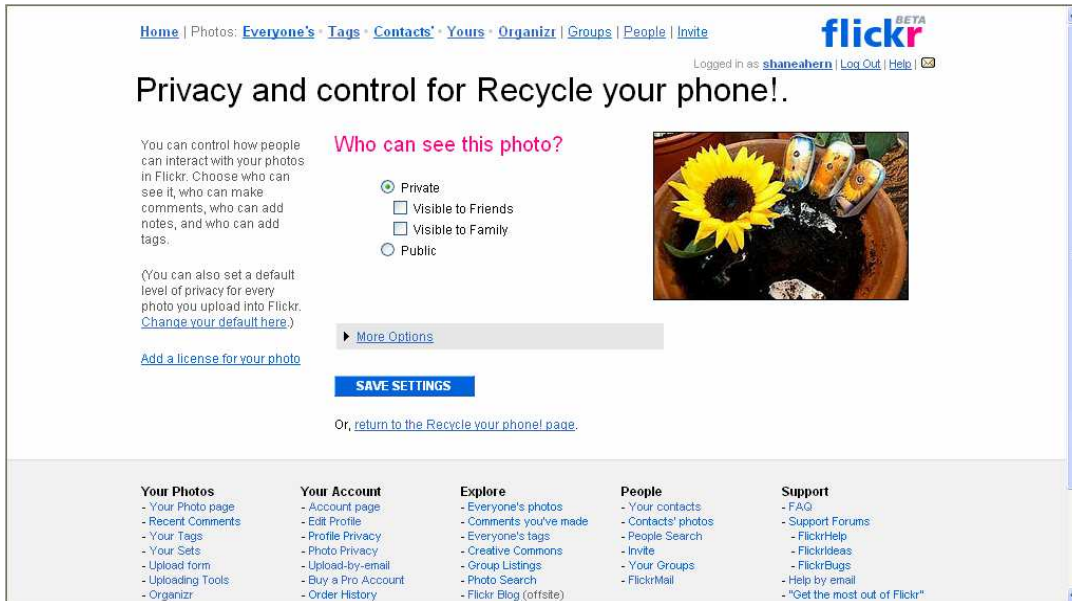


Figure 4: Flickr – Privacy Settings Interface

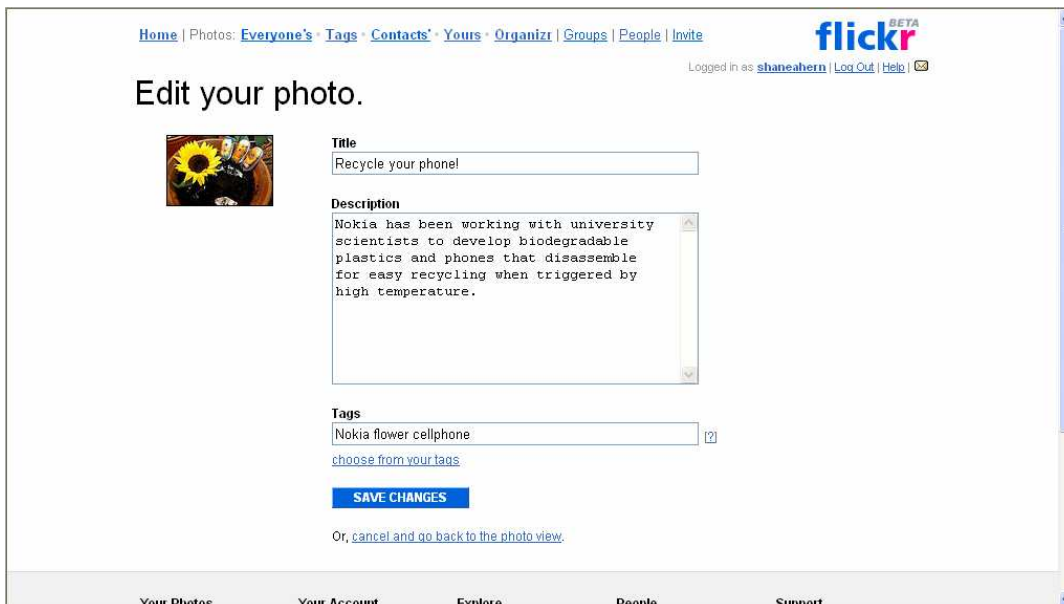


Figure 5: Flickr – Add Tag Interface

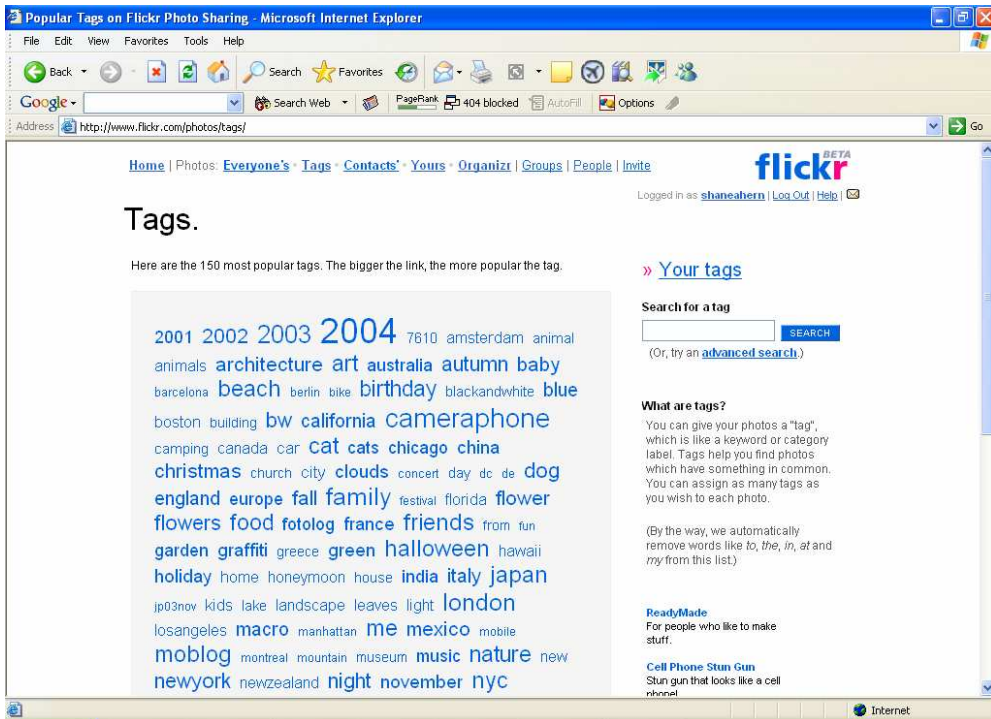


Figure 6: Flickr – Shared Tags Interface

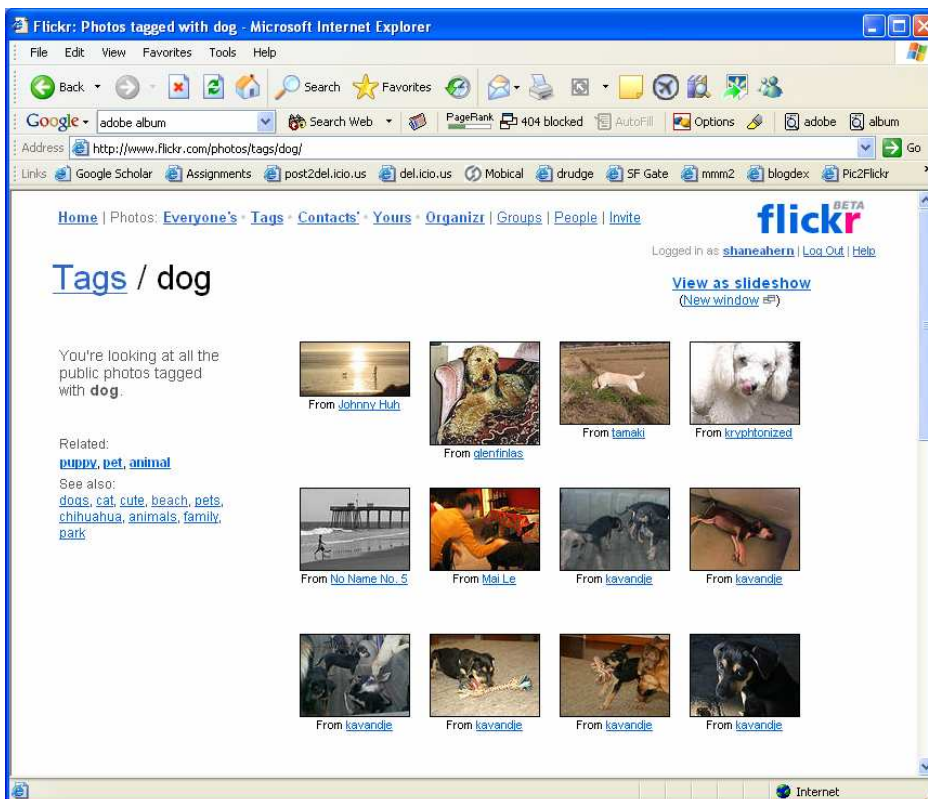


Figure 7: Flickr – Search By Tags Interface

Nokia LifeBlog

Lifeblog from Nokia is both a mobile phone application and desktop PC application. The mobile application has a good photo browsing interface that organizes photos by time on the phone and allows users to easily post photos to a blog (see Figure 8) and synchronize photos to their desktop PC (see Figure 9).

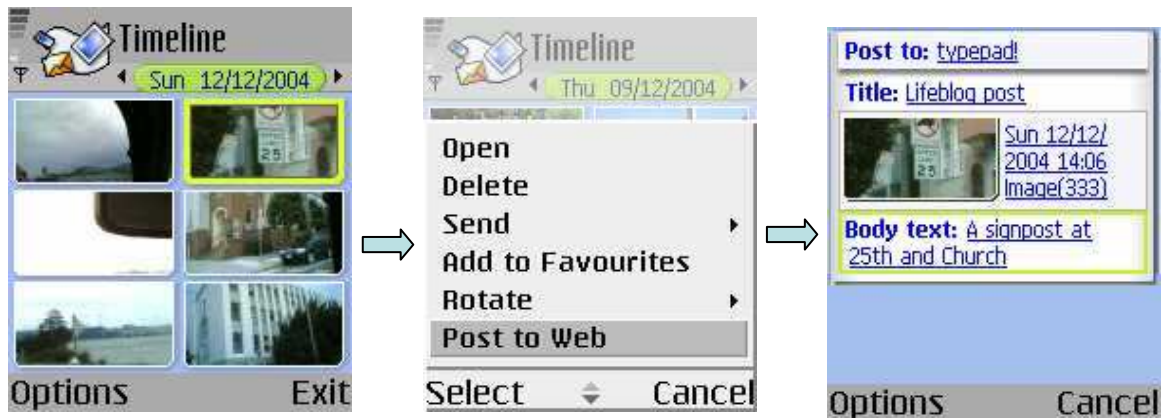


Figure 8: Nokia Lifeblog Mobile App – Post to Blog Interface

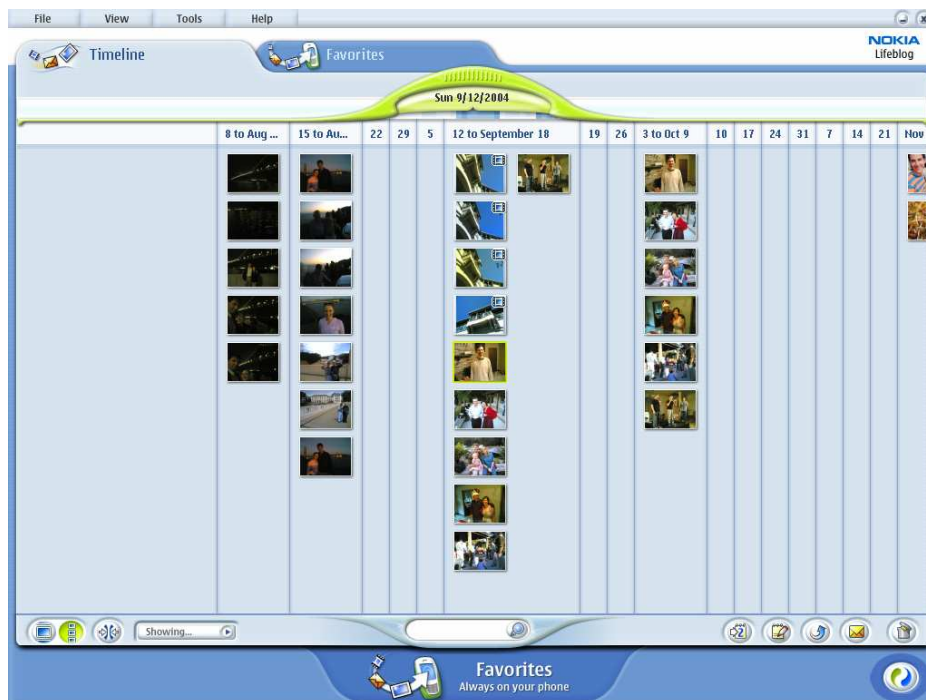


Figure 9 Nokia LifeBlog Desktop PC App – Timeline View

MMM1

The Mobile Media Metadata (MMM) project at UC Berkeley's School of Information Management and Systems has developed two applications of interest. In MMM1, a user could capture a photo, and receive assistance in annotating the photo at time of capture by a system that processes automatically-generated contextual metadata and allows the user to verify and/or modify system-generated annotation guesses.

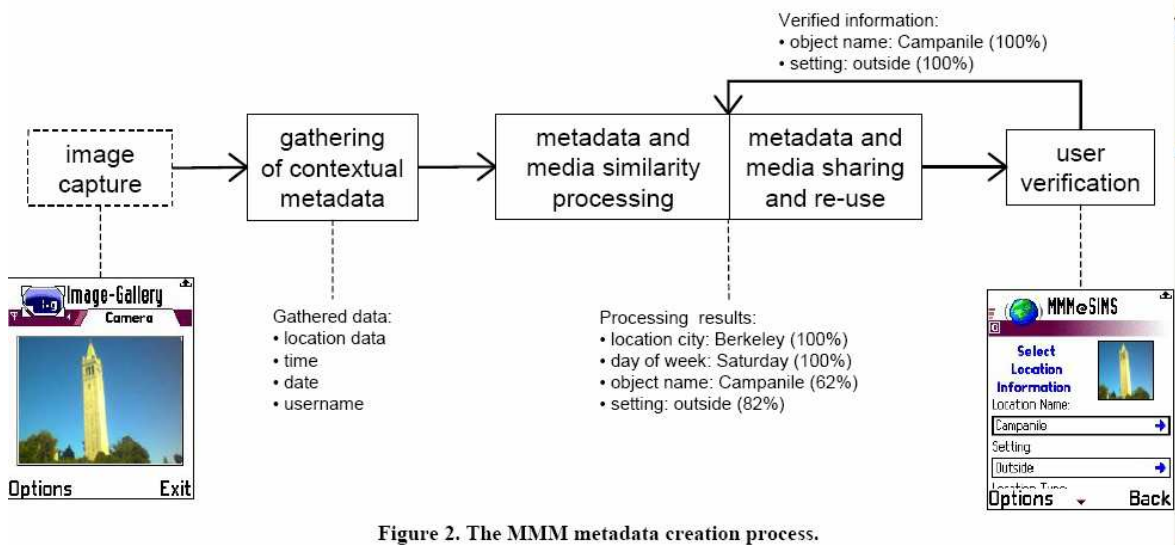


Figure 2. The MMM metadata creation process.

Figure 10: MMM1– Time-of-capture Annotation Interaction Flow

MMM2

MMM2 solved a number of difficulties faced by users in MMM1. All photos captured by users of MMM2 are automatically uploaded to a web application that allows users to manage, annotate and share photos with other users. Users of MMM2 may also share photos a time-of-capture.

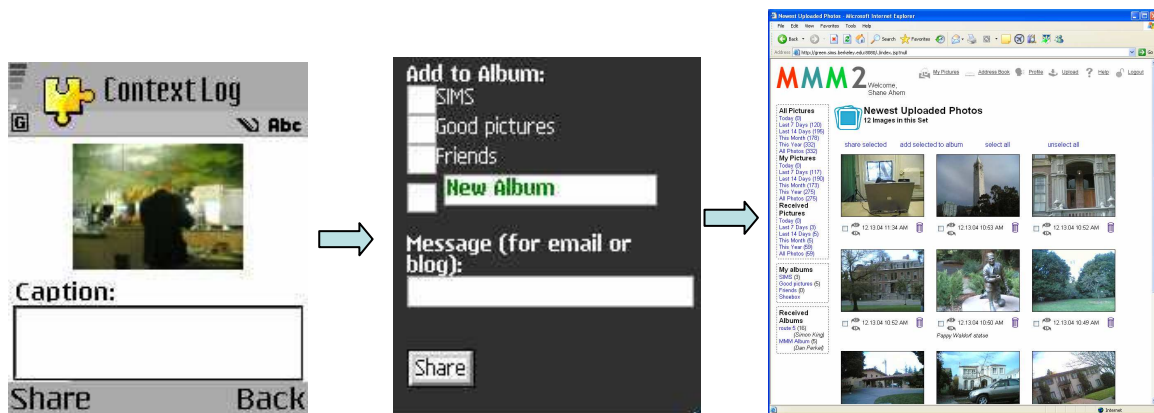


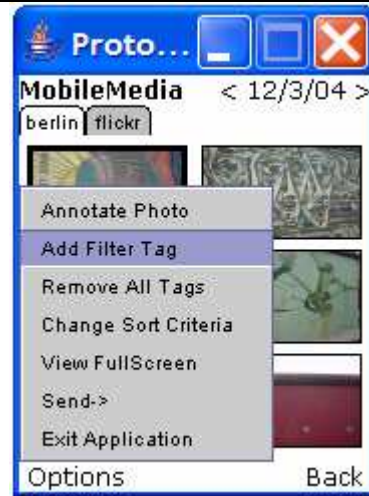
Figure 11: MMM2– Interaction Flow from Mobile App to Web App.

Prototype Screenshot Overview

(do we want to submit prototype?)



Screen 1, Gallery View – This screen allows a user to browse through all photos stored on the phone. The tags at the top of the screen (“berlin” and “flickr”) indicate tags that the gallery is currently filtering on – in this example, only photos tagged as “berlin” that have been sent to flickr (feed tags appear gray, description tags are white) appear in the gallery. The gallery window scrolls left to right, with photos matching the selected tags in chronological order.



Screen 2, Popup Menu – The user can press the “Options” soft key to display a context-sensitive menu. If a photo is selected in the gallery, photo-specific options such as “Annotate Photo”, “View Full Screen”, and “Send” will appear. If no photo is selected, the menu items are restricted to functions that affect gallery mode. In this example the “Add Filter Tag” option will allow the user to control the way gallery filters existing photos.



Screen 3, Filter Tags – When managing tags in gallery mode, the left side of the screen displays a list of available tags. Tags in **bold** are currently selected as



Screen 4, Photo Details – This screen allows a user to view information about a photo, including filename, size, capture date, and tags currently associated with the

gallery filters. The tags with gray background are feed tags (determined by where photos have been sent), those with white backgrounds are description tags (set by the user.) The numbers appearing to the right of the tag name are the number of photos matching a given query (selecting “desktop” on the example screen would result in a match on 5 photos.) The tag list scrolls vertically, allowing for more than one page of tags.

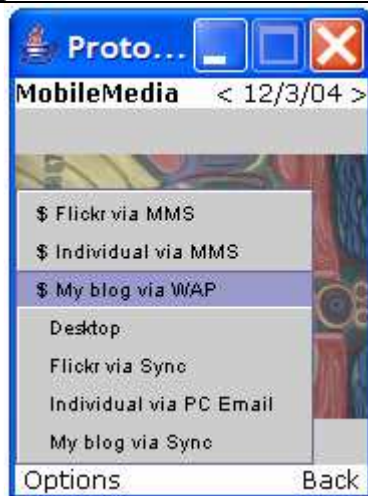
photo.



Screen 5, Popup Menu – When selected from the photo details page, the popup menu contains photo-specific functions. Selecting “Manage Tags” will allow the user to set the descriptive tags associated with a photo.



Screen 6, Manage Tags – Similar to gallery mode’s filter tag management (screen 3, above), this screen allows a user to associate tags with a specific photo (shown in the right pane.)



Screen 7, Feeds Menu – Selecting “Send” from the popup menu (visible in screens 2



Screen 8, Send Confirmation -- This screen informs the user of feed-specific

<p>and 5) creates a new menu listing all available methods for sending a photo. Delivery methods that are associated with some expense appear preceded by “\$” in the menu.</p>	<p>delivery details, including cost and estimated delivery time (based on image file size.) This screen would also allow the user to select, for example, email recipient address, or a text message to associated with the sent photo. After sending, the photo is associated with the feed tag corresponding to this delivery destination (“my blog”, in this case.)</p>
---	--

User Scenarios Using Prototype

Persona – Jeanette

Jeanette, a 26 year old web designer in San Francisco, is an avid user of her camera phone. Sporting an active online social life, she likes posting many of her pictures on her blog and on flickr, where she can share it with her friends and family. In addition, she also occasionally orders prints of her favorite photos using ofoto. She also archives all her photos together in the photo album on her laptop. Despite her enthusiasm for photos, managing all these different repositories and manually routing her photos was tiring drudgework... until she got mobile media asset management software.

Scenario 1 – Photo Capture and Sharing

Walking down Valencia Street, she spies a gorgeous table in a vintage furniture shop. While she can't really justify the furniture purchase, she has to get a picture of it. Besides, her friend Rachel would absolutely love it! She pulls out her camera phone and selects the image capture application. A screen appears showing a preview image of the photo. Jeanette points the camera at the table, lines up a nice shot, and with the press of a button snaps a picture of it.

The capture application then pops up a dialog asking “Would you like to view/annotate your photo?” Jeanette selects the ‘Yes’ button, and the application takes her to a screen showing an image of the photo and various descriptive information, including the file name, date and time of capture, image resolution and size (screen 4, above). In addition, there are some tags indicating where the photo has been sent, and descriptive tags about its content. “flickr” is already listed as a destination, as she sends most of her photos to this services and so made these options the default when she first configured the software. Below that is a space for descriptive annotation tags and free-text captions, which do not appear until she enters them.

She first wants to make sure that the photo is archived correctly, so she pushes the ‘Options’ button and selects ‘Manage Tags’ from the resulting menu (screen 5). This gives her a list of annotation tags she has created before (screen 6). The tags organize data as non-hierarchical labels, much like flickr or g-mail. As she has been saving photos from around the Mission District for a possible photo project, she selects the ‘Mission’ tag she created before, causing that tag name to appear **bold** in the menu. She then selects

‘Create New Tag...’ and enters in the text ‘Table’ to give her self an additional cue. She then pushes the ‘Back’ button and is returned to the viewing and annotation screen, where she now sees her new annotations reflected.

Jeanette wants to make sure that her friend Rachel sees this picture, so she also wants it to go to her blog, which she knows Rachel reads. To do so, she pushes the ‘Options’ button on her phone and selects ‘Send’ from the resulting menu. She then selects ‘My Blog’ from the resulting menu (screen 7.) She has the option to send it immediately using WAP (the ‘\$’ in the menu indicates this will cost her a little bit of money), or to just wait until the next time she synchs with her laptop and let the complimentary software on her laptop manage it (for free). She wants Rachel to see it sooner, and doesn’t care about the cost so she opts to send it now. A send confirmation page appears with details about cost and delivery time (screen 8). The small cost is fine with Jeanette, so she then presses the ‘Send’ button and is returned to the image view and annotation screen.

Scenario 2 – Managing Existing Photos

Sharing a photo with Rachel reminded Jeanette of a few other photos she took about 15 minutes ago but did not annotate, as she wanted to take a bunch of shots in succession without pausing to annotate each. She now hits the ‘Options’ button and selects ‘Go To Image Gallery’ where she sees the various pictures still on her phone, now indexed by time (screen 1). She knows that the photos have been posted to flickr, since this has been set as a default behavior all day, so she selects the “flickr” tag via the Filter Tags screen (screen 3). She quickly finds a group of 7 photos she took earlier of the murals on the Women’s building on 18th Street. She selects them as a group, pushing a selection button as she moves the cursor over each image. She then pushes the ‘Options’ button and selects ‘Annotate Photo’ to return to the image viewer and annotation screen. It is slightly different then before, for example, choosing to view the images in full screen now results in a scrollable slideshow, and the metadata shown is that common to all photos. She verifies that the photos are all going to flickr (those defaults sure are handy), also routes them to her desktop for archival at synchronization time, and as before, adds the ‘Mission’ annotation tag to the group.

She then hits the ‘Options’ button, selects ‘Exit Application’, and continues her walk down Valencia. Now everything is taken care of, as the table photo is on its way to her blog (she could verify this later should she want to) and the mural photos will eventually be routed to the right place next time she synchs up. And no doubt Rachel is going to die when she sees that table.

Strength and Weaknesses of Prototype

Java Infrastructure

Using Java as a prototyping tool gives us the ability to quickly build and modify the prototype while also allowing us to design some application infrastructure. Building a lo-

fi prototype using HTML or PowerPoint would not have resulted in a fully interactive prototype and would not have allowed us to test the viability of some application design ideas. In particular, the FeedStore and TagStore modules allowed us to build a fully functional query system and to test price and delivery time calculations...

Still need to implement

There are still a number of areas which our prototype does not address. In particular, we did not have an opportunity to implement UI to test the following:

- **Delivery details** – Clicking on feed tags in photo details (screen 4) should show delivery details about when and how a photo was sent. Though we have no immediate plans to implement actual delivery, this UI could have been simulated.
- **Interactive tabs in gallery** – The selected filter tags appearing at the top of gallery view (screen 1), should allow the user to interact (either remove selected tag, or remove all others), allowing user to remove query constraints without going to the filter tags view (screen 3)
- **Additional input on send page** – The send page (screen 8) should allow the user to select an email address or phone number for individual recipients as well as allowing free text entry for messages to accompany sent photos. While the PC-based prototype does not have access to a mobile device's contact list, this could have been simulated.
- **Adding tags** – There is currently no way to add new tags. Though the application logic exists, this is an important user interface to explore.
- **Finding tags/browsing a large list of tags** – The current prototype contains only 15 or 20 tags, which makes list-based tag selection feasible. If the system contained hundreds of tags, we would need to provide the user with better mechanisms for finding their desired tags. Having the list sort by frequency of use, or sort alphabetically with the user able to jump to specific start letters both seem like reasonable options.
- **Send when no cost** – Currently the prototype enumerates all feed delivery mechanisms (e.g. WAP, email, USB). A more simplified interface might allow delivery to occur over any method when there is no cost associated. We could probably implement this UI immediately, but the underlying application logic is somewhat complex...
- **Group annotation** – Currently the prototype does not allow the user to select multiple photos and annotate or send them as a group. This would be nice to test, since displaying a large set of selected photos could be tricky given limited screen space.
- **Macros/keyboard shortcuts** – Our prototype doesn't currently include a capture interface, but we have discussed creating keyboard shortcuts or user-definable macros that would allow a user to annotate or delete a photo at the time of capture with a single keystroke. For example, the 0-9 keys on a phone keypad could be used as a rating system: 0 indicating that a photo should be discarded, 9 indicating that it should be saved and that the user should be reminded to annotate it later (perhaps at sync time.) Alternatively, a subset of the number keys could be used for rating, while other keys are mapped to more complex macros (e.g. send to a particular destination or associate with some group of tags).

Bigger Problems for Thought

There are a few issues related to shared metadata that we cannot address in our prototype without some additional thought:

- **Shared tags** – In any real world application, a user will receive as well as send photos. There must be some way for incoming metadata to be incorporated into our system so that, for instance, a user can search through tags for not only the photos he created, but also the photos he received from friends.
- **Metadata phone home** – One of the nice things (from a programmer's standpoint) about using a push/publish model for feeds, is that once photos are sent away they become somebody else's problem. In reality however, it is possible that these photos will return to the device that created them, containing metadata that was created elsewhere. While it might not be completely practical to expect automatic metadata updates from other sources, it is necessary to at least be able to process incoming metadata when it is available.

System Architecture Description

A high-level view of the system architecture for routing photos based on the user-input depicted in the prototype is shown in Figure 12 below. The system is based on the concept of a "feed". A feed is the combination of a destination (e.g. desktop, blog, a friend's mobile phone) and a transport mechanism (Bluetooth synch, email, MMS). A feed descriptor tells the system which feed and transport module to use, what privacy levels, and what metadata to allow through.

A feed descriptor might look something like:

ID: <some ID>

Description: <human-readable string>

Cost per Kb: <some \$>

Estimated delivery time: <X seconds>

Privacy: <at least [public|friends|private]>

Filter out metadata: <[location|tags|captions|etc.]>

Feed Module: <ClassName>

Transport Module: <ClassName>

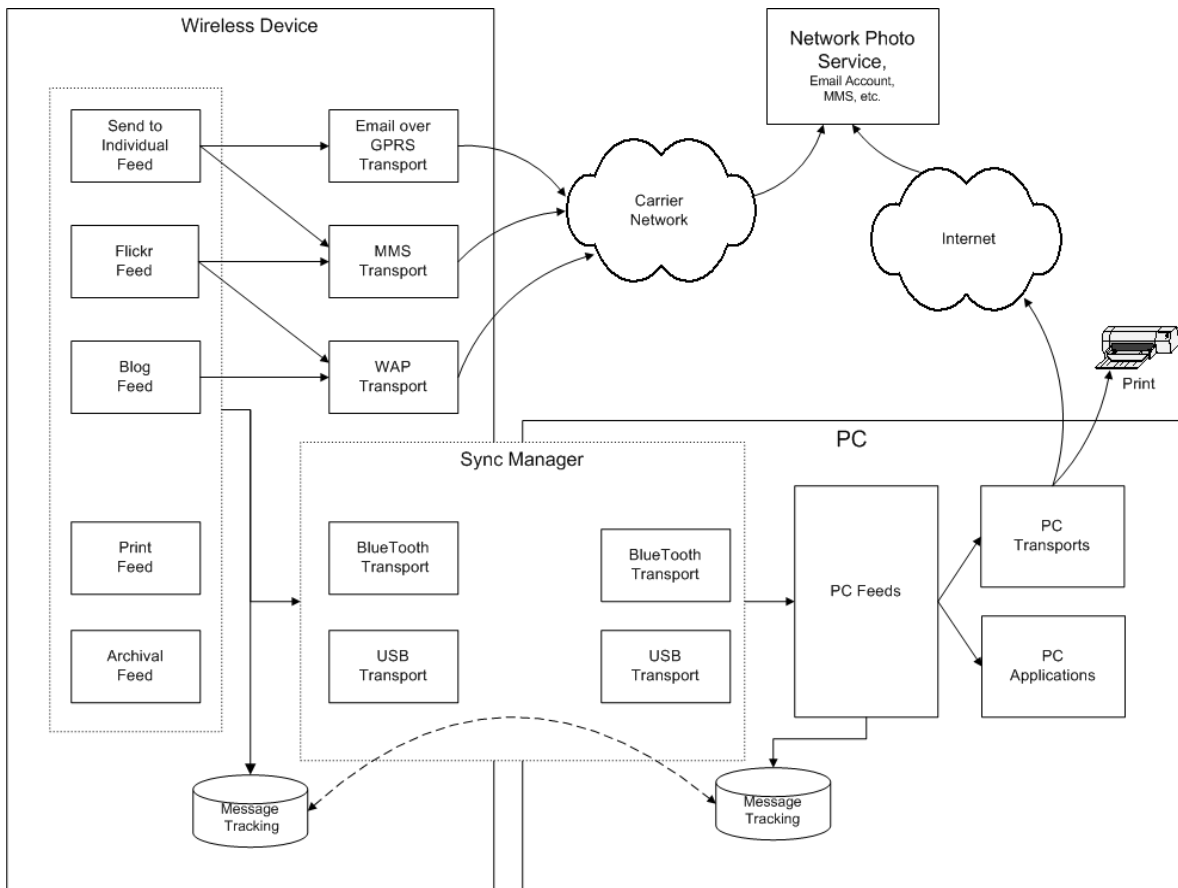


Figure 12 – MMAM System Architecture