# Personalized Recommendations in Peer-to-Peer Systems

Loubna Mekouar
University of Waterloo
Waterloo, Canada
lmekouar@bbcr.uwaterloo.ca

Youssef Iraqi
Dhofar University
Salalah, Oman
y_iraqi@du.edu.om

Raouf Boutaba
University of Waterloo
Waterloo, Canada
rboutaba@bbcr.uwaterloo.ca

## Abstract

*In Peer-to-Peer (P2P) file sharing systems, peers have to choose the files of interest from a very large and rich collection of files. This task is difficult and time consuming. To alleviate the peers from the burden of manually looking for relevant files, recommender systems are used to make personalized recommendations to the peers according to their profile. In this paper, we propose a novel recommender scheme based on Peers' Similarity and Weighted Files' Popularity. Simulation results confirm the effectiveness of the Symmetric Peers' Similarity with Weighted File Popularity scheme in providing accurate recommendations, this way, increasing peers' satisfaction and contribution since peers will be motivated to download the recommended files and serve other peers meanwhile.*

## 1 Introduction

### 1.1 Motivation

The most important challenges of online environments is to assure positive interactions and satisfactory transactions. People will usually back up from meeting new strangers and buying new items that they did not know or try before. Therefore, people minimize their interactions and transactions and tend to remain in their comfort zone. Positive interactions can be achieved through reputation and recommendation systems. Reputation and recommender systems are most commonly found in e-Commerce applications. A positive interaction between strangers can be achieved by relying on reputation. Reputation is used to rate people, for example, the reliability of a vendor in ebay, or the authentic behavior in peer-to-peer systems [4]. It is the most valuable information that helps reduce the offensive and deceptive behavior of online users. While reputation systems are used to enforce appropriate behavior by rating people, recommender systems are used to allow satisfactory transactions by rating the quality of items (e.g. products, services). Recommender systems will motivate online users to discover new and interesting items and choose items that they will most probably like. The search for interesting items will be easier and the users will be alleviated from the burden of finding these items.

### 1.2 Recommender Systems

Recommender systems are widely used in e-Commerce applications (e.g. amazon.com, CDnow.com, BizRate.com, CNet.com, Epinions.com, Yahoo.com) [1, 7, 8]. The collaborative filtering recommender techniques are achieving widespread success on the web [2]. User-based collaborative filtering algorithms are the most used in e-Commerce applications. In these techniques, a user-item matrix is used to compute similarity between users and then recommendations are made. On the other hand, item-based collaborative filtering algorithms analyze the relationships between items and then use these relationships to compute recommendations to users (e.g. amazon.com). However, collaborative filtering algorithms suffer from the cold start, popularity effect, data sparseness and trust problems [1, 7, 8].

Although, e-Commerce applications have been using recommender systems for at least a decade, this research field is still new in P2P systems. In these systems, reputation mechanisms have been proposed to detect malicious peers that send inauthentic content and isolate them from the system [4]. This helps in reducing the spread of malicious content. While reputation systems are used to rate peers in P2P systems, recommender systems are used to rate files. Only few research works have been proposed for P2P systems [6, 9, 5].

### 1.3 Contributions

In [5], we proposed *Files' Popularity Based Recommendation (FP)* and *Asymmetric Peers' Similarity Based Recommendation with File Popularity (ASFP)*. The recommendations in *FP* scheme are based on the most popular files within the peers that have the requested file, while, the rec-

978-0-7695-3315-5/08 $25.00 © 2008 IEEE
DOI 10.1109/WETICE.2008.45

ommendations in *ASFP* scheme are based on the most popular files within the most similar peers that have the requested file. In this paper, we propose a novel way of computing similarity and a weighted approach to compute *Files' Popularity*. We propose three new schemes based on *Peers' Similarity* combined with *Files' Popularity* and *Weighted Files' Popularity*. However, we focus on *Symmetric Peers' Similarity with Weighted File Popularity* scheme. The goal is to achieve the followings:

- Increase peers' satisfaction by informing them about files of interest.
- Increase peers' contribution [3] since peers will be motivated to stay connected to download the recommended files and hence serve other peers meanwhile. Free riders may be motivated to share their files to get a profile that reflects their preferences in order to receive accurate recommendations.
- Attract more users by making the search process easier, and more efficient.
- Preserve network resources since peers will not have to download a large number of files that they do not like and will just discard.

The paper is organized as follows. Section 2 discusses related works. Section 3 describes the basics of the proposed recommender schemes. Section 4 describes the proposed recommender schemes. Section 5 presents the performance evaluation. Finally, Section 6 concludes the paper.

## 2 Related Works

In [9], the authors propose a distributed collaborative filtering method that is self-organizing and operates in a distributed way. Similarity ranks between items are computed by log-based user profiles and are stored locally in buddy tables. To perform a recommendation for a given user, the buddy tables for all the items in user's profile are downloaded and the relevance ranks are computed based on the user-content relevance model. In [6], the authors propose a decentralized recommendation system that takes advantage of the high clustering coefficient of Preference Networks. The nodes of these networks are users of a file sharing system and the links are connections between pairs of nodes that share one or more identical files. The authors experimentally prove that the preference networks are small worlds. They propose a recommendation scheme based on the fact that nodes can be naturally gathered together on the basis of common interests.

The proposed recommender schemes in [9, 6] are suitable for decentralized P2P systems but not for partially decentralized systems. However, partially decentralized systems are the most popular. In this paper, we propose a novel recommender scheme for these systems.

## 3 The Basics of the Recommender Scheme

In E-commerce, the collaborative filtering technique is based on the ratings by the customers of the products and/or the purchases they made. A rating scheme from 1 (not interesting) to 5 (very interesting) can be used to collect customers opinions. To use this *explicit* rating scheme in peer-to-peer systems, the user has to explicitly provide a rating for each file he/she downloads according to its content (i.e. matches the user's preferences or not). There is no guarantee that this additional effort will be made by the users. We adopt an *implicit* rating approach since it is automatically done. We propose to assign a rating of 1 (*I like it*) to the files owned by the user. All other files are assigned a rating of 0 (*I do not know*).

In this paper, we take advantage from the partial search process used in partially decentralized systems. Once a peer sends a request for a file to its supernode and assuming the file is not found locally, the supernode will forward this request to other supernodes. Once the search result is received at the supernode level, a list of all the shared files of the peers that have the requested file is also given. Based on this information and the requestor peer's profile, the supernode uses the recommender scheme to make recommendations. We adopt a user-based collaborative technique rather than an item-based collaborative technique since recommendations should be made in real time and it is usually time consuming when we are dealing with a large collection of files. Exploring relationships between peers is better since the search process limits the number of peers that are taken into consideration.

## 4 The Proposed Recommender Scheme

### 4.1 Formal Notations

In the remaining of the paper, we will use the following formal notations:

Let $P$ be the set of all peers in the system.

Let $F$ be the set of all files shared by the peers.

Let $p_i$ be the requestor peer looking for a file $f_x$. $p_i$ is the user to whom the recommendation will be made.

Let $P_{f_x}$ be the set of peers that possess the file $f_x$.

Let $F_{P_{f_x}}$ be the set of files that these peers possess in addition to $f_x$. This is the set of files that peers share.

Let $f : P \rightarrow \Omega(F)$, such that $f(p_j)$ is the set of files held by peer $p_j$ for every $j$ and $\Omega(F)$ is the power set of $F$. Then we have:

$$F_{P_{f_x}} = \bigcup_{p_k \in P_{f_x}} f(p_k)$$

Let $G_{p_i} = F_{P_{f_x}} - f(p_i) - \{f_x\}$ be the set of files that $p_i$ does not have from the set $F_{P_{f_x}}$ not including the file $f_x$. These are all the files owned by the peers in $P_{f_x}$ that the peer $p_i$ does not have.

The *Files' Popularity Based Recommendation* (FP) and *Asymmetric Peers' Similarity Based Recommendation with File Popularity* (ASFP) schemes have been presented in [5]. The FP scheme will allow a peer to discover the files that are more popular among the peers that have the requested file.

## 4.2 Asymmetric Peers' Similarity with Weighted File Popularity (ASWFP)

Peers' similarity is an important factor in this technique. To be able to make accurate recommendations, the active user's files are compared against those of other users. The goal of this process is to find peers with similar preferences as the active peer $p_i$ and make recommendations on the files that they have. In fact, the files' popularity approach is applied within these peers.

For every peer $p_j$ in $P_{f_x}$, the similarity relationship is:

$$ASim_{p_i}(p_j) = \frac{|f(p_i) \cap f(p_j)|}{|f(p_i)|}$$

$|f(p_i)|$ is assumed not null, which means that the peer $p_i$ owns at least one file. If the peer does not own any file, the *FP* scheme is used.

The value of $ASim_{p_i}(p_j)$ is a numerical score that shows how similar the peer $p_j$ is to the peer $p_i$. Note that this similarity relationship is not symmetric, i.e. $ASim_{p_i}(p_j)$ may not be equal to $ASim_{p_j}(p_i)$

This scheme will choose only peers that have $ASim_{p_i}(p_j) \geq t_2$. Where $t_2$ is a threshold.

Let $S_{p_i}^{t_2} = \{p_j, p_j \in P_{f_x} \; and \; ASim_{p_i}(p_j) \geq t_2\}$

We apply the *Weighted File Popularity* within the set $S_{p_i}^{t_2}$ of peers most similar to peer $p_i$.

In this technique, we weight the files owned by the peers within the set $S_{p_i}^{t_2}$ of peers most similar to peer $p_i$ according to peers' similarity. For every file, we add the similarity value for each peer $P_j$ that owns this file and then we divide by the sum of all peers' similarities for peers that belong to the set $S_{p_i}^{t_2}$.

For every file, we compute

$$WPop_{ASim}(f_k) = \frac{\sum_{P_j \in S_{p_i}^{t_2} \cap P_{f_k}} ASim_{p_i}(p_j)}{\sum_{P_j \in S_{p_i}^{t_2}} ASim_{p_i}(p_j)}$$

The recommendation list is sorted according to the weighted popularity of the files $WPop_{ASim}(f_k)$ with the files that have a higher weight at the top of the list.

## 4.3 Symmetric Peers' Similarity Based Recommendation (SS)

Peers' similarity is also an important factor in this technique. We compare the active user's files against those of other users.

For every peer $p_j$ in $P_{f_x}$ we define the similarity relationship as:

$$SSim_{p_i}(p_j) = \frac{|f(p_i) \cap f(p_j)|}{|f(p_i) \cup f(p_j)|}$$

Note that the denominator $|f(p_i) \cup f(p_j)|$ can not be null.

The value of $SSim_{p_i}(p_j)$ is a numerical score that shows how similar the peer $p_j$ is to the peer $p_i$. Note that this similarity relationship is symmetric, i.e. $SSim_{p_i}(p_j) = SSim_{p_j}(p_i)$

This scheme will choose only peers that have $SSim_{p_i}(p_j) \geq t_3$. Where $t_3$ is a threshold.

Let $S_{p_i}^{t_3} = \{p_j, p_j \in P_{f_x} \; and \; SSim_{p_i}(p_j) \geq t_3\}$.

### 4.3.1 Symmetric Peers' Similarity with File Popularity

We apply the *FP* scheme within the set $S_{p_i}^{t_3}$ of peers most similar to peer $p_i$. For every file, we compute

$$Pop_{SSim}(f_k) = \frac{|P_{f_k} \cap P_{f_x} \cap S_{p_i}^{t_3}|}{|P_{f_x} \cap S_{p_i}^{t_3}|}$$

Note that if $t_3 = 0$ then $Pop_{SSim}(f_k) = Pop(f_k)$.

This scheme will recommend only files $f_k$ such that $Pop_{SSim}(f_k) \geq t_1$, where $t_1$ is a threshold. This recommendation list is sorted according to the popularity of the files $Pop_{SSim}(f_k)$ with the files that are most popular at the top of the list.

### 4.3.2 Symmetric Peers' Similarity with Weighted File Popularity (SSWFP)

We apply the *Weighted File Popularity* within the set $S_{p_i}^{t_3}$ of peers most similar to peer $p_i$. For every file, we compute:

$$WPop_{SSim}(f_k) = \frac{\sum_{P_j \in S_{p_i}^{t_3} \cap P_{f_k}} SSim_{p_i}(p_j)}{\sum_{P_j \in S_{p_i}^{t_3}} SSim_{p_i}(p_j)}$$

The recommendation list is sorted according to the weighted popularity of the files $WPop_{SSim}(f_k)$ with the files that have a higher weight at the top of the list.

Figure 1 shows the proposed recommender schemes. Each scheme is based on a similarity metric in order to choose similar peers to the active peer. Each scheme also relies on either the *Files' Popularity* technique or the *Weighted Files' Popularity* technique to rate files and recommend only files with high popularity.

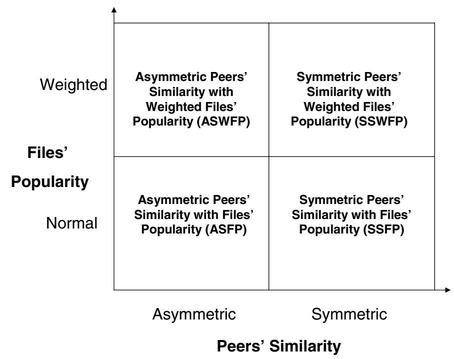


**Figure 1. The proposed schemes**

The goal from proposing all these schemes is to explore all the possible alternatives that can be taken into consideration in computing the peers' similarity metric and finding the most relevant files to recommend. As shown in the performance evaluation section in [5], *Asymmetric Peers' Similarity* based recommendation scheme provides more accurate recommendations than the *Files' Popularity* based recommendation scheme. *Asymmetric Peers' Similarity* is used to find the neighbors of the active peer. These neighbors are the users that have similar taste and preferences as the active user. In this paper, in addition to *Asymmetric Peers' Similarity* as defined in [5], we use also the *Weighted Files' Popularity*. By using this technique, we give a priority to most popular files owned by the neighbors and according to the degree of similarity between these neighbors and the active peer. In this paper, we narrow further the list of the neighbors, making the recommendations even more accurate, especially when the *Weighted Files' Popularity* is used (SSWFP). In this scheme, the focus is on users that are likeminded as the active user, with almost similar interests and only files with a high weighted popularity value are recommended. In this paper, we focus on *Symmetric Peers' Similarity with Weighted File Popularity* scheme.

## 5 Performance Evaluation

### 5.1 Simulated Schemes

We have simulated the following techniques:

- *Symmetric Peers' Similarity with Weighted File Popularity (SSWFP)* (see section 4.3.2).
- *Cosine measure* technique (COS): similarity metric is based on the *cosine measure* with a *predicted rating* as described in the literature [7, 8, 9].

It is important to note that since implicit rating is used for files' recommendations, the use of Pearson correlation [7, 8, 2, 9] is not applicable since the average rating given by a peer $P_i$ to its files is always 1. In this case, the Pearson correlation measure is not well defined since the denominator is null. For this reason, we compare our proposed scheme to the *user-based collaborative filtering* using the *Cosine measure* as a similarity metric. In this technique, the active user and any other user are represented by two vectors and the similarity between them is measured by computing the cosine of the angle between the two vectors.

### 5.2 Simulation Parameters

The simulation parameters are the following:

- We simulate a system with 1,000 peers and 1,000 files.
- At the beginning of the simulation, each peer has at most 50 files and each file has at least one owner. This allows for a maximum of 50,000 files.
- Peers are divided into four interest categories (C1: Action, C2: Romance, C3: Drama and, C4: Comedy) and files are also divided into the same four categories. Each peer belongs to one category. For this reason, peers prefer to have most of the files from their category and only few files from other categories.
- The percentage of peers in each category is 25% and the percentage of files in each category is 25%.
- We simulate 100,000 requests for each simulation.

Following the simulation parameters, peers with indices from 1 to 250 belong to the category C1 (Action), peers with indices from 251 to 500 belong to the category C2 (Romance). Peers with indices from 501 to 725 belong to C3 (Drama) and peers with indices from 726 to 1,000 are peers that belong to C4 (Comedy).

### 5.3 Performance Metrics

For each scheme, we computed the following metrics:

- For each peer's category, the ratio of the number of downloaded files that belong to this category over the files downloaded by the peers from this category ((a) bars).
- For each peer's category, the ratio of the number of downloaded files that belong to other categories over the files downloaded by the peers from this category ((b) bars).

The proposed recommender scheme help the peers discover new, interesting and relevant files. We define *Peers Satisfaction* as the average number of files from peers' category

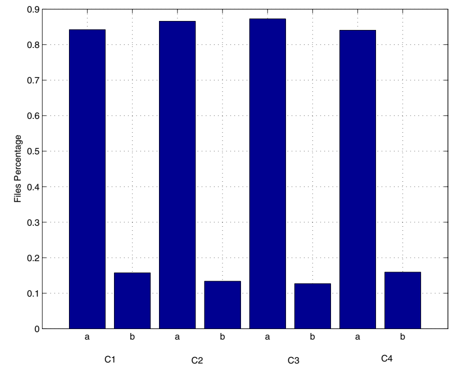


**Figure 2. The cosine measure: First scenario**

over all the files owned by the peers. This will affect the trust given by the peers to the recommender system. The higher this value is, the more confident the peers are that the recommended files match their preferences.

## 5.4 Simulation Results

We compare the COS and the SSWFP scheme under the same conditions. The simulations were conducted in three different scenarios.

***First Scenario***

A the beginning of simulations, peers get files from the category that they prefer with a probability of 0.9 and files from other categories with a probability of 0.1. Figure 2 depicts the percentage of files for each peer category based on the cosine measure. From this figure, it is clear that peers have more files from the category that they prefer. This is shown by the fact that (a) bars are higher than (b) bars for the four categories. Figure 3 depicts the results of the SSWFP scheme. This figure shows clearly the effectiveness of this recommender system. In this set of simulations, an average of 99% of files that were selected by peers have been recommended to them. In this case, both COS and SSWFP have similar results.

***Second Scenario***

To show the effectiveness of this scheme compared to the cosine measure, we performed another set of simulations with the same parameters as described above except that at the beginning of the simulations peers get files from the category that they prefer with a probability of 0.8 and files from other categories with a probability of 0.2.

Figure 4 depicts the percentage of files for each peer category based on the cosine measure. From this figure, it is clear that peers have more files from the category that they prefer. However, the results are not as good as in the previous set of simulations. Figure 5 depicts the percentage of files for each peer category based on the SSWFP scheme.

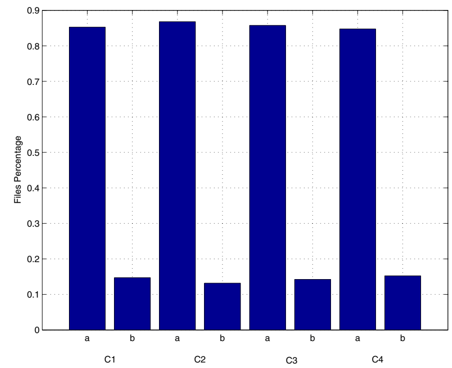


**Figure 3. The SSWFP Scheme: First scenario**

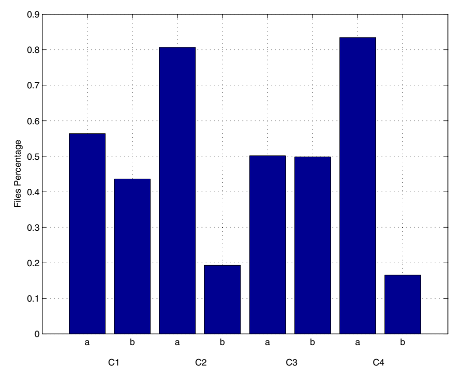


**Figure 4. Cosine measure: Second scenario**

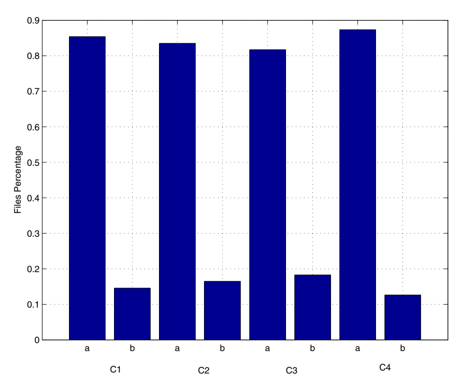


**Figure 5. SSWFP Scheme: Second scenario**

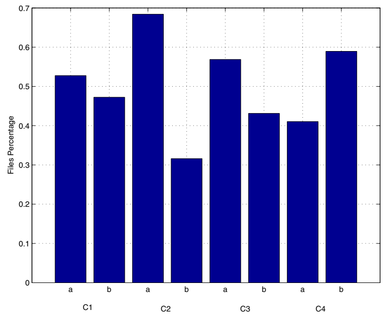


**Figure 6. Cosine measure: Third scenario**

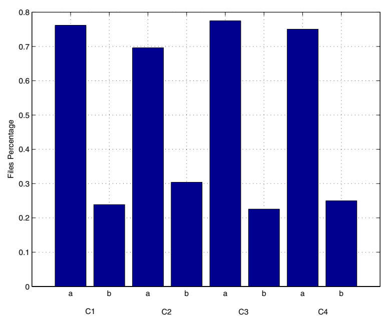


**Figure 7. SSWFP Scheme: Third Scenario**

It clearly shows that the performance of this recommender scheme surpasses the cosine measure. The proposed recommender scheme makes a good distinction between files categories and recommend the appropriate files based on peers' profiles.

***Third Scenario***

To show that the SSWFP scheme makes appropriate recommendations compared to the COS scheme, we performed a new set of simulations where at the beginning peers get files from the category that they prefer with a probability of 0.7 and files from other categories with a probability of 0.3. Decreasing the former probability will make it hard for a recommender scheme to recognize peers' profile. However by comparing figures 6 and 7 that represent the results obtained for the COS and SSWFP schemes respectively, SSWFP provides better recommendations' accuracy.

Table 1 presents a summary of the previous results comparing the COS and the SSWFP schemes in terms of peer's satisfaction. It is clear that as the initial distribution of files becomes fuzzy, the COS scheme is not able to clearly find the exact profile of peers and hence will lead to poor peer's satisfaction. On the other hand the SSWFP scheme is able to make good recommendations with improvements of up to 25% in terms of peer's satisfaction.

| Scenario | Probabilities | COS | SSWFP |
|---|---|---|---|
| First | 0.9/0.1 | 85.55% | 85.66% |
| Second | 0.8/0.2 | 67.67% | 84.50% |
| Third | 0.7/0.3 | 54.77% | 74.55% |

**Table 1. Summary of peer's satisfaction**

## 6 Conclusion

In this paper, we proposed a new recommender scheme for partially decentralized peer-to-peer systems. The proposed recommender scheme makes personalized recommendations to the peers according to their profile. Simulation results show the effectiveness of the *Symmetric Peers' Similarity with Weighted File Popularity* scheme in providing accurate personalized recommendations compared to the *cosine measure*. The proposed recommender scheme does not require any additional effort from the users since implicit rating is used.

## References


[1] G. Linden, B. Smith, and J. York. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. In *IEEE Internet Computing*, pages 76–80, Jan/Feb 2003.

[2] P. Massa and P. avesani. Trust-aware Collaborative Filtering for Recommender Systems. In *International Conference on Cooperative Information Systems (CoopIS)*, 2004.

[3] L. Mekouar, Y. Iraqi, and R. Boutaba. Free Riders under Control through Service Differentiation in Peer-to-Peer Systems. In *IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2005)*, December 2005.

[4] L. Mekouar, Y. Iraqi, and R. Boutaba. Peer-to-Peer Most Wanted: Malicious Peers. *The Computer Networks Journal, Special Issue on Management in Peer-to-Peer Systems: Trust, Reputation and Security*, 50(4):545–562, 2006.

[5] L. Mekouar, Y. Iraqi, and R. Boutaba. Recommender Schemes in Peer-to-Peer Systems. In *International Workshop on Dependable and Sustainable Peer-to-Peer Systems (DAS-P2P 2008)*, August 2008.

[6] G. Ruffo, R. Schifanella, and E. Ghiringhello. A Decentralized Recommendation System based on Self-Organizing Partnerships. In *IFIP-Networking 2006*, pages 618–629, May 2006.

[7] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Analysis of Recommendation algorithms for E-commerce. In *EC*, October 2000.

[8] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-Based Collaborative Filtering Recommendation Algorithms. In *WWW*, May 2001.

[9] J. Wang, J. Pouwelse, R. L. Lagendijk, and M. J. T. Reinders. Distributive Collaborative Filtering for Peer-to-Peer File Sharing Systems. In *SAC*, April 2006.