How to Build a Persuasive Web Application
Three Fundamental Feedback Loops

Evan Rosenfeld
Stanford University, Department of Electrical Engineering,
350 Serra Mall
Stanford, CA 94305-9505
evans.rosenfeld@stanford.edu

Abstract. Successful web applications are able to influence the behaviors of their users. The top 40 Facebook applications were analyzed to determine the most persuasive ways of achieving such influence. This paper presents a framework of feedback loops for this purpose and then applies it within the context of Facebook.

Keywords: Persuasive technology, Facebook, social networking, influence, behavior change, online communities.

1 Introduction

Persuasive technologies are able to influence user behavior by applying principles recognized in the field of Social Psychology [1]. They are pervasive on the internet, where an application’s success is tied to its ability to influence the behaviors and actions of its users. For example, most applications grow by persuading users to recruit their friends as new users.

Social Networking. The ascent of social networks including Facebook – with over 65 million users [2] – has provided researchers with an unprecedented opportunity to investigate the cues that effect behavior change. In May, 2007, the Facebook Platform was launched, allowing developers to create web applications that utilize the social graph. This has resulted in an explosion in the variety of options presented to the user and an increased ability to track user actions. For these reasons, Facebook was chosen as the medium for this research.

Theory. User-application interaction is often modeled by control flow diagrams in which users move between states by performing specific actions. Often, these flows linearly transport a user from an initial state to a final state. While this is suitable when the goal is to influence a sporadic behavior, it has been demonstrated that to influence a regular behavior requires repetitive reinforcement [5]. Therefore, flows that effect regular behaviors tend to be recursive loops which are reinforced (feedback) at every repetition. Many papers have shown how web services can influence user behavior [3,4]. This paper continues in that vein by contributing a framework of three feedback loops that the most persuasive applications use specifically to acquire new users and retain engagement.
2 Methodology

The 40 Facebook applications with the most active users as listed by the Adonomics leaderboard\(^\text{[6]}\) were sorted by percent active users and then grouped by theme. For each group, control flow diagrams were drawn to describe the user experience. Patterns in these diagrams uncovered three categories of feedback loops, as shown in figures 1-3.

3 Fundamental Feedback Loops

**Growth Loop.** The growth loop (Fig.\(^\text{[1]}\)) encourages the invitation of new users. In the *action* variant, the actions of current users automatically serve as invitations to new users. This is natural within social networks because one must register with an application before interacting with it.

   This variant is commonly seen in applications which allow users to ‘hug’ their friends. The loop works as follows: 1) users ‘hug’ a friend; 2) the friend is notified of this ‘hug’ by an invitation to register with the application; and 3) the friend accepts the ‘hug’ by registering with the application. Such an application can grow simply by engaging its current users.

   Because users are compliant\(^\text{[7]}\), psychological tweaks can enhance this loop’s persuasiveness. For example, telling users that they should ‘hug’ 20 friends can result in this desired behavior. Another strategy is to encourage users to be in the top quartile of the ‘most hugging’ users.

   In the *viral* variant, current users must invite additional, new users to reach the application’s functionality. *Viral* flows tend to grow faster than *action* flows because they ensure that the application has sent out a number of invitations proportional to its current user base. The request that a user recruit friends before interacting with the application is sure to discourage some users. Therefore, it is important to frame the required invitations as actions and to make the functionality sufficiently enticing to persuade users to send invitations.

   This loop is seen in instances where users send points to other users. For example, ‘Send HOTNESS’ requires a user to invite 15 friends before being allowed to see their own ‘hotness score’. This incentive has proven powerful enough that many people do invite 15 friends, far more than making up for the users who leave immediately. Another application, ‘Perfect Match’ has used a variant of this loop. For every friend a user invites, a new ‘perfect match’ is revealed. In this sense, each invite can be seen as an action for which points are rewarded and automatically redeemed for additional functionality.

**Frequency Loop.** The frequency loop (Fig.\(^\text{[2]}\)) rewards frequency of usage and visitation. Since the monetary value of an application is related to page views, this loop is critical. There are two types of frequency to reward: frequency of visit and frequency of action.

   Frequency of visit lends itself naturally to a points-based framework. Users are rewarded points for every visit within a set time period. These points can be redeemed for new functionality.