Information & Management 48 (2011) 328–335

Contents lists available at SciVerse ScienceDirect

Information & Management

journal homepage: www.elsevier.com/locate/im

How organizations motivate users to participate in support upgrades of customized packaged software

Huoy Min Khoo [a,*], Cecil Eng Huang Chua [b], Daniel Robey [c]

[a] Department of Information Systems and Technology Management at the University of Texas, San Antonio, USA
[b] University of Auckland, New Zealand
[c] Georgia State University, USA

ARTICLE INFO

Article history:
Received 5 May 2010
Received in revised form 14 March 2011
Accepted 4 August 2011
Available online 7 September 2011

Keywords:
Packaged software upgrade
Maintenance strategies
IT implementation
Framing
Enterprise resource planning
Enterprise systems

ABSTRACT

Support upgrades are undertaken to correct errors, improve speed, and otherwise improve an existing version of customized packaged software. Motivating such projects is especially challenging, because users typically anticipate little benefit. We investigated ways of motivating user participation in maintenance upgrading projects via an in-depth case study using the method of communicative framing. This argues that (1) the positivity or negativity of a frame, and (2) the credibility, salience, and consistency of the diagnostic, prognostic and motivational elements of the frame influence others' willingness to believe and respond to a communication. Our case study explored user motivation and participation in an upgrade of SAP software in an organization where no upgrade had been performed in the past three years. We discovered that: (1) a negatively valenced communicative frame characterizing an external party as a threat is most likely to motivate users, and (2) framing the support upgrade simultaneously affected user motivation and reinforced the position of the IT support group.

Published by Elsevier B.V.

1. Introduction

Today, organizations buy or lease customized packaged software in lieu of writing their own to improve the productivity of their business operations. Packaged software is developed by a vendor and sold or leased to many users. *Customized packaged software* allows adopting users to modify the base software to fit their specific needs within certain limits imposed by the vendor. In contrast, *traditional packaged software* allows little or no customization; the user simply installs the software whose features have been completely determined by the vendor.

This distinction between these packaged software, of course, depends on the organizational context. For example, in some organizations, Microsoft Office is treated as traditional packaged software because the IT department only checks that new versions of desktop tools work with the existing operating system. However, organizations that employ Excel-based macros consider the *same* Office package as customized packaged software, because the installation of a new version would only be complete after existing macros had been tested and found to be compatible with the new version of Microsoft Excel. Today, customized packaged software is used in a wide variety of enterprise applications, including accounting, demand forecasting, MRP, CRM, financial trading (e.g., Murex), and decision support.

The maintenance of customized packaged software differs substantially from that of custom software. It requires the initial installation of the software, followed by the installation of upgrades or service packs provided by the vendor as well as the maintenance of custom code. Upgrading customized packaged software is a non-trivial activity, often consuming a year or more and expending resources equal to 20–30 percent of the cost of the original software. Software upgrades may not work as planned and can create serious organizational problems by interrupting service for a long time. For example, the Lockheed Martin Corp. had to shut down its SAP servers for 71 h when upgrading from SAP 4.7 to 6.0, after which they experienced slowdowns in database queries and user lockout problems. Software upgrade projects are therefore expensive, high risk propositions that must be carefully managed.

In many cases, upgrades are made though they have no direct business benefits. Indeed, the need to upgrade often arises because software vendors establish "sunset" dates beyond which they state that they will not support earlier versions of packaged software without additional fees: thus firms are forced to upgrade, accrue additional charges, or maintain the packaged software themselves. Because organizations often do not have access to the packaged software source code, upgrading to a new version becomes the most realistic choice [9]. Such an upgrade can be considered a *support upgrade*, which costs substantially more than an upgrade of traditional package software. With customized packaged software,

* Corresponding author. Fax: +1 210 458 6305.
E-mail address: huoy.minkhoo@utsa.edu (H.M. Khoo).

0378-7206/$ – see front matter. Published by Elsevier B.V.
doi:10.1016/j.im.2011.09.001

one must test and adapt the library of existing code and infrastructure to the new version of the software.

Due to changes introduced in a new version of software, support upgrades may involve changes to business processes that users perform on a regular basis; thus users may resist the imposition of modifications that they did not request. User reluctance thus creates an additional project risk.

The objective of our study was therefore to examine this issue and identify effective strategies for eliciting user participation in a support upgrade. We report here on an in-depth analysis of a single support upgrade case using the concept of communicative framing [3].

2. Related literature

2.1. Packaged software implementation and upgrades

Organizations choose to use packaged rather than custom software to reduce development costs, shorten implementation times, acquire state-of-the-art best practice, reduce maintenance, and obtain extended functionality [11]. However, software problems may arise after packaged software is installed. One ongoing challenge in its management is in coping with its evolution, as vendors often change software capabilities when introducing new versions and terminating support for older ones [10]. Although client organizations may influence vendors, they lack direct control over software support. Upgrades initiated by vendors can introduce unwanted additional functionality or eliminate existing desired functions. In addition, upgrades are costly. An ERP upgrade may cost the implementing firm a million US dollars or more.

Academic research on packaged software issues has predominantly focused on initial implementation (e.g. [2]). Of the few studies on customized packaged software maintenance, the majority address descriptive research questions instead of building a theory to explain how or why organizations cope with packaged software maintenance [13]. The research that has attempted to build a theory focused on the decision to upgrade rather than the process of upgrading.

To our knowledge, only Nah and Delgado [12] have explicitly studied the upgrade process by comparing the CSFs for package implementation with those for package upgrades. They reported that CSFs related to change management, such as user training and coordination across all affected parties, are important for success. Users are important to an upgrade, because new versions are typically designed to suit a generic business, thus users must either adapt to these generic prescriptions or participate in the customization and configuration of the package to fit with their business processes. Users also create idiosyncratic adaptations and workarounds to overcome limitations in any customized software. For example, Excel spreadsheets are often employed in place of database systems to allow users more control over their data. Such unintended work practices, while critical to actual business practice, are often unknown to a maintenance team and/or the customized package software vendor.

Furthermore, lack of user participation can create difficulties. For example, users may refuse to relinquish their IT equipment to allow the support upgrade. Similarly, users may complain about the upgrade or otherwise disrupt work on the support upgrade.

2.2. Communicative framing

Communicative framing theory explains how messages can be structured to motivate other users to action. As such, it makes it possible to understand the mobilization of users to participate in support upgrades of customized packaged software.

Communicative frames have been used in the marketing literature to show how binary characteristics of advertisements influence targeted recipients. For example, researchers have studied the impact of a promotion window as restrictive or expansive in its effect on consumer purchasing decisions [5], and the impact of stating a discount either as a percentage or as a dollar value [6].

Our purpose, however, was not to understand how messages could mobilize purchases, but how they can motivate organizational stakeholders. We therefore turn to communicative framing research in the communications and social movement literature. The literature in these fields revolves around two questions: (1) how does positive and negative framing influence effort? and (2) how does social context influence the construction of communicative frames?

2.2.1. Negative/positive framing

Most research on negative and positive framing stems from prospect theory, the basis of the work that earned Kahneman the 2002 Nobel Prize in Economics. According to prospect theory, a project situation framed in terms of its likelihood of loss may elicit a different effort from project members than one framed in terms of potential gain: the effort is likely to be asymmetric. Also, Kahneman and Tversky had come to prospect theory by noting that people preferred a focus on a positive outcome (you have a 50% probability of surviving) to the equivalent negative statement (you have a 50% chance of dying). Some research has suggested, however, that context determines whether a frame should be constructed positively or negatively; thus there may be no need to frame a customized package support upgrade positively, e.g., by saying it results in a better life: it could be re-framed as necessary to avoid a worse life. When maintenance is performed well, users may fail to see that the effort improves their lives.

IS projects create disruptions to the social order of a firm, and this can create anxiety that can cause performance to fall [7]. Therefore, the change management literature would suggest that positively framed messages should be employed during a customized package support upgrade.

2.2.2. Communication context

Other characteristics of a communicative frame can influence its effectiveness. However, most research has found that these other characteristics depend strongly on the specific communicative frame being examined (e.g. [8]). For example, for discounts, the question arises: Should the discount be stated in absolute monetary terms or as a percentage of the cost? Such questions are irrelevant to the framing of restrictions to civil liberties during a war [4]. More relevant questions include whether such restrictions should be framed in an objective or advocacy tone [1], or whether heroic images should be used.

Research on communicative frames has identified two general factors that influence a frame's resonance with recipients: frame credibility and frame salience. *Frame credibility* is the recipient's willingness to accept the message as true. *Frame salience* is the importance of the message to its recipients.

Frame credibility can be considered from two dimensions: empirical credibility (the believability of evidence presented) and frame consistency (the continuity of the message as it is created and disseminated) across three core framing tasks: diagnostic, prognostic, and motivational. *Diagnostic* frames describe a particular problem and suggest blame or causality resulting from it. *Prognostic* frames propose a solution to the problem and also identify strategies that can be used to solve it. *Motivational* frames develop rationales for action that encourage groups of stakeholders to act. Effective frames must be consistent across their diagnostic, prognostic, and motivational forms.