Although there have been many advances in end-user programming environments, recent empirical studies report that programming still remains difficult for end users. We hypothesize that one reason may be lack of effective support for helping end-user programmers problem-solve their own way around barriers they encounter. Therefore, in this paper, we describe the Idea Garden, a concept designed to help end-user programmers generate new ideas and problem-solve when they run into barriers. The Idea Garden has its roots in Minimalist Learning Theory and problem-solving theories. Our proof-of-concept prototype of the Idea Garden concept in the CoScripter end-user programming environment currently targets three barriers reported in end-user programming literature. It does so using an integrated, just-in-time combination of scaffolding for problem-solving strategies, for design patterns, and for programming concepts. Our empirical results showed that this approach helped end-user programmers overcome all three types of barriers our prototype targeted.

Key words: end-user programming; mashups; problem solving; Idea Garden; development frameworks and environments; designing software

* An early version of portions of this paper appeared in Cao et al. (2011).