

# A Generative Programming Approach to Interactive Information Retrieval: Insights and Experiences

Saverio Perugini<sup>1</sup> and Naren Ramakrishnan<sup>2</sup>

<sup>1</sup> Department of Computer Science,  
University of Dayton, OH 45469-2160 USA  
saverio@udayton.edu

<http://homepages.udayton.edu/~perugisa>

<sup>2</sup> Department of Computer Science,  
Virginia Tech, VA 24061-0106 USA  
naren@cs.vt.edu

<http://people.cs.vt.edu/~ramakris>

<http://oot.cps.udayton.edu>

**Abstract.** We describe the application of generative programming to a problem in interactive information retrieval. The particular interactive information retrieval problem we study is the support for ‘out of turn interaction’ with a website – how a user can communicate input to a website when the site is not soliciting such information on the current page, but will do so on a subsequent page. Our solution approach makes generous use of program transformations (partial evaluation, currying, and slicing) to delay the site’s current solicitation for input until after the user’s out-of-turn input is processed. We illustrate how studying out-of-turn interaction through a generative lens leads to several valuable insights: (i) the concept of a web dialog, (ii) an improved understanding of web taxonomies, and (iii) new web interaction techniques and interfaces. These notions allow us to cast the design of interactive (and responsive) websites in terms of the underlying dialog structure and, further, suggest a simple implementation strategy with a clean separation of concerns. We also highlight new research directions opened up by the generative programming approach to interactive information retrieval such as the idea of web interaction axioms.

## 1 Introduction

Generative programming has been typically been applied to problems at the crossroads of programming languages and software engineering such as modularizing cross-cutting concerns, synthesizing programs from formal specifications, and automatically generating program documentation. We describe here a novel application of generative programming to a problem in interactive information retrieval [1].

### 1.1 Motivating Example

Everybody has experienced the frustration in interacting with automated information systems where the system does not let the user progress through the dialog without answering a currently posed question. For instance,

- 1 **System:** Welcome to the automated flight reservation system.
- 2 **System:** Please say the date on which you wish to travel.
- 3 **Sallie:** I'd like to fly from New York to Brussels next week.
- 4 **System:** Sorry, I didn't understand. Please specify a date.
- 5 **Sallie:** If you can tell me available dates, I can choose.
- 6 **System:** Please say the date on which you wish to travel.
- 7 **Sallie:** [Hangs up]

The mental mismatch between Sallie's conception of the task and the system's design is manifest in the above interaction. The system is expecting a date in Line 3 whereas Sallie specifies her choice of source and destination cities. Even though this information is going to be relevant further into the interaction, the system insists on specifying date before going further. Similar inconveniences happen while interacting with websites. A site presents hardwired choices of hyperlinks to pursue and even though the user's input is pertinent and probably solicited deeper in the site, there is no way for the user to circumvent the given navigation structure.

Our solution to the above situations, where the user cannot answer a currently posed question, but does have some other information pertinent to the task at hand, is to provide a capability for *out-of-turn* interaction. For instance, we would provide a capability for the user to speak something into the browser, and in this way supply out-of-turn input. Such 'unsolicited reporting' has been recognized [2] as a simple form of *mixed-initiative interaction*, a dialog management strategy where the two participants take turns exchanging the initiative. Using out-of-turn interaction, the user is empowered to complete an information-finding task in the manner that best suits her conception. Moreover, we show that out-of-turn interaction, irrespective of when it happens, can be supported uniformly by a generative programming approach. A website that currently provides a hardwired choice of completion options can be automatically converted into one that supports out-of-turn interaction!

The idea behind our approach is quite simple: we liken out-of-turn interaction to non-sequential evaluation of a computer program, e.g., partial evaluation. We model an information seeking interaction as a computer program so that user inputs correspond to values for program variables (ref. Fig. 1, left). When the user provides input in the order in which they are requested, we are sequentially evaluating the program, i.e., interpreting it. In a web hierarchy, this would correspond to plain browsing. When the user provides out-of-turn input, we jump ahead to nested program segments that involve that input and simplify them out via partial evaluation. By employing sequences of such interpretations and partial evaluations, we can support complex interactions that involve both responsive as well as out-of-turn inputs.