Programming Context-aware Applications with Continuations

Engineer Bainomugisha  Jorge Vallejos  Pascal Costanza  Wolfgang De Meuter
Software Languages Lab - Vrije Universiteit Brussel - Belgium

Motivation: a context-aware post-it app

Application behaviour as a user moves about

- @office
  - meetings
  - @home
  - home todo
  - @supermarket
  - @meeting room

- active computation
- suspended computation

Observations

- Computation is mostly driven by context changes
- Dynamic behaviour switching
- Computations arbitrarily stop and resume
- State needs to be preserved between context changes

Problem Statement

- Reacting to unpredictable order of context changes
  - Use of event handlers leads to inversion of control
  - Deciding on which computation behaviour to make available
  - Computation behaviours may conflict each other

- Manual state management
  - Saving application state
  - Recovery: reinitialising computation state

```
(define (shopping location)
  (if (supermarket? location)
      ; initialise application
      ; restore/recover application state
      (shopping-list))
  (if (left-supermarket? location)
      ; save application state
      ; stop application ))
```

- hard & error prone!

Computations as continuations

- Modularising computations as continuations
- A computation encloses behaviour and state
- A computation responds to resume and suspend messages

Reactive programming+continuations

- Reactive context reasoning
- suspends/resumes
- Implementation platform

- iScheme[1]: a Scheme implementation that runs on iPhone/iPod and iPad devices

Observations

- Computation is mostly driven by context changes
- Dynamic behaviour switching
- Computations arbitrarily stop and resume
- State needs to be preserved between context changes


http://soft.vub.ac.be/~ebainomu