



University of HUDDERSFIELD

University of Huddersfield Repository

Iqbal, Saqib

Representing Aspects In Design Model

Original Citation

Iqbal, Saqib (2009) Representing Aspects In Design Model. In: University of Huddersfield Research Festival, 23rd March - 2nd April 2009, University of Huddersfield. (Unpublished)

This version is available at <http://eprints.hud.ac.uk/id/eprint/5231/>

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

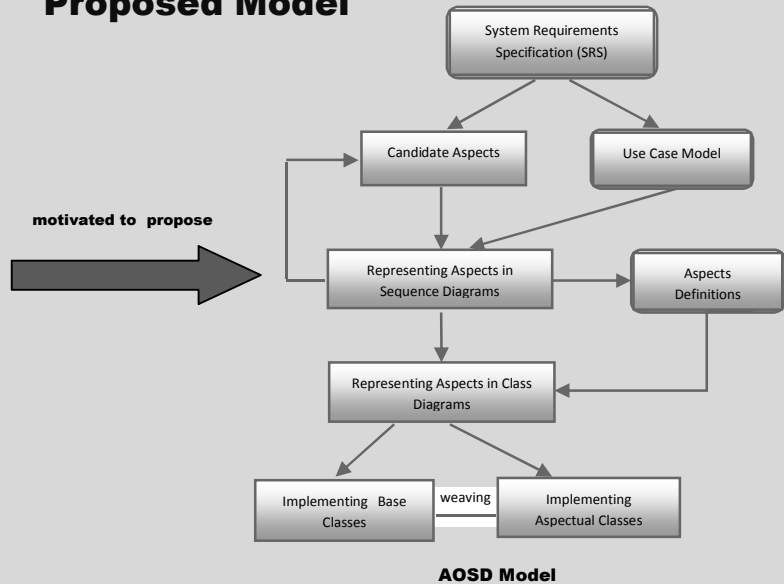
For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

<http://eprints.hud.ac.uk/>

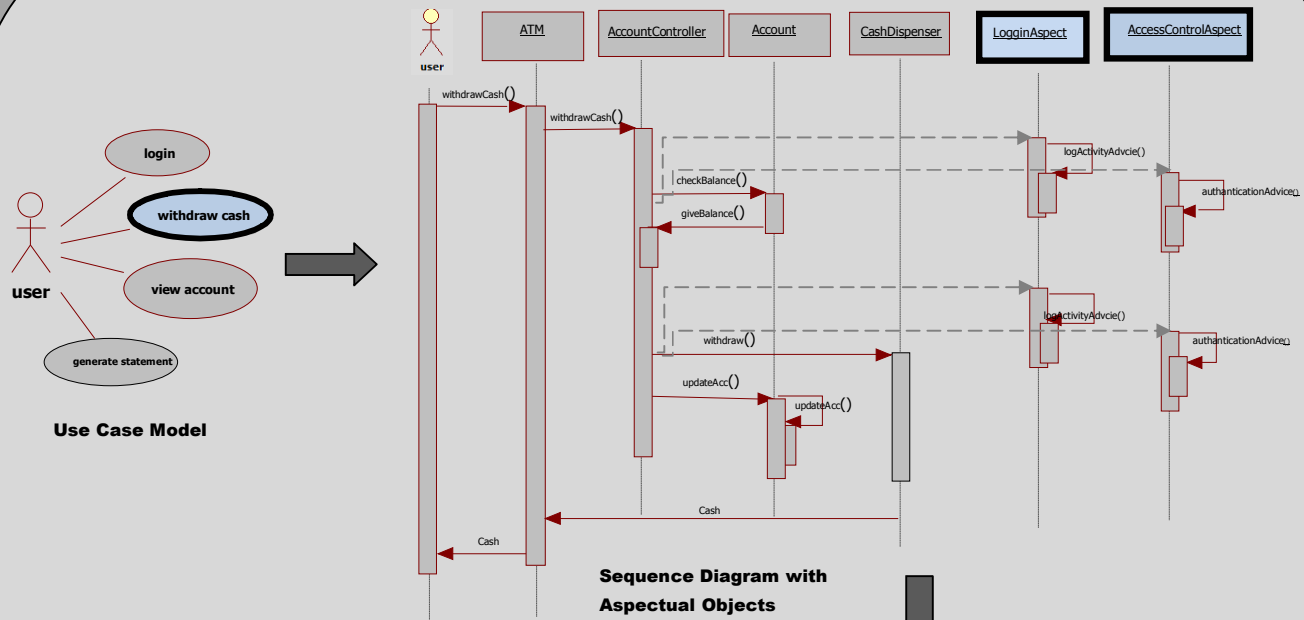
Problems/Motivation

- No design model in existence for effectively representing Aspects
- No effective mechanism in use for identifying and representing Aspects in Analysis and Design phases
- No effective mechanism in use for showing interaction of Aspects with Base Program Model
- No mature design model in existence for Aspects which could be interpreted in Implementation (Code).
- Need for Unified Design model of Aspect-Oriented Software Development

Proposed Model

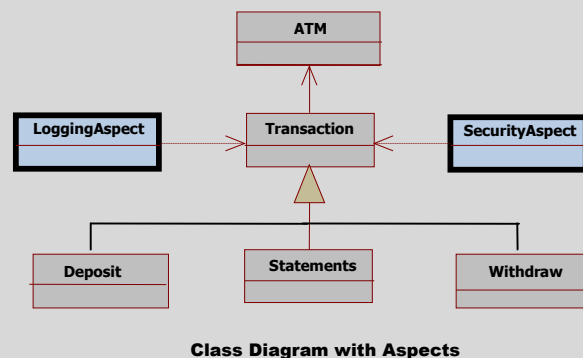


Impact



Results

- Representation of Aspects along with Base Functions in Design Model
- Facilitates designers to identify interaction of Aspects with Base program
- Facilitates designers to identify Aspects, Join Points and ultimately helps writing effective point cuts and advices
- Easy design to interpret in Implementation



Class Diagram with Aspects