

# A methodology for analyzing and designing GUI plug-ins

John Doe  
Technical University of Denmark  
DK-2800 Lyngby, Denmark  
Email: <http://www.example.com>

John Doe  
Technical University of Denmark  
DK-2800 Lyngby, Denmark  
Email: <http://www.example.com>

John Doe  
Technical University of Denmark  
DK-2800 Lyngby, Denmark  
Email: <http://www.example.com>

**Abstract**—Since basic graphical user interface elements can be created very easily with only a few lines of code, most of the developers start their GUI plug-ins' lives directly with the implementation phase. As more GUI elements are being added, their functionality start to interfere maybe even overlap leading to a less stable and reliable product. The current paper addresses these problems by providing a methodology for analyzing and designing GUI plug-in systems. The solution is described and exemplified using an ongoing project as a study case.

## I. INTRODUCTION

What is the paper about.

### A. Background

What problems do we run into when starting building an Eclipse plug-in.

### B. Related work

What solutions have other papers brought

## II. ANALYSIS AND DESIGN METHOD

The steps used to design and analyze the plug-in.

### A. User interface

UI mock-ups.

Requirements identification. Captured in BON *sce-nario\_chart*.

### B. Events

*Incoming* events representing user actions and *outgoing* events meant to inform the user.

### C. Components

Major components captured in BON *static\_diagrams* using *cluster\_chart* and *class*.

### D. Components communication

Component interfaces added to the interface diagram using *feature*, *require* and *ensure*. This will later result in plug-in extensions and extension points.

Update scenarios with events.

### E. Code generation

Beetlz generates the Java code from BON specification.

## III. CONCLUSION

In conclusion

## ACKNOWLEDGMENT

The authors would like to thank...

## REFERENCES

- [1] H. Kopka and P. W. Daly, *A Guide to L<sup>A</sup>T<sub>E</sub>X*, 3rd ed. Harlow, England: Addison-Wesley, 1999.