In this paper, we explore how the componentware paradigm can be used to re-engineer existing software systems. As a case study, we use the prototype implementation of the CASE tool AutoFocus which has been developed using standard object-oriented design techniques. Although designed for re-usability with respect to certain criteria, AutoFocus did not use a component-based approach yet. The case study concentrates on the repository subsystem, as this part is currently the least modular part of AutoFocus. We outline essential concepts of the componentware paradigm, including a methodology how to carry out a component-based design process, introduce the current state of AutoFocus, and sketch how we derive a component-based redesign, including a migration strategy from purely object-oriented legacy systems to component-based systems.

Componentware; Object-Oriented Software Engineering; Design; Redesign; Java; CASE; Repository
entries: