

Data-Centric Systems and Applications

Florian Daniel
Maristella Matera

Mashups

Concepts, Models
and Architectures

 Springer

Chapter 9

Mashups and End-User Development

Figures

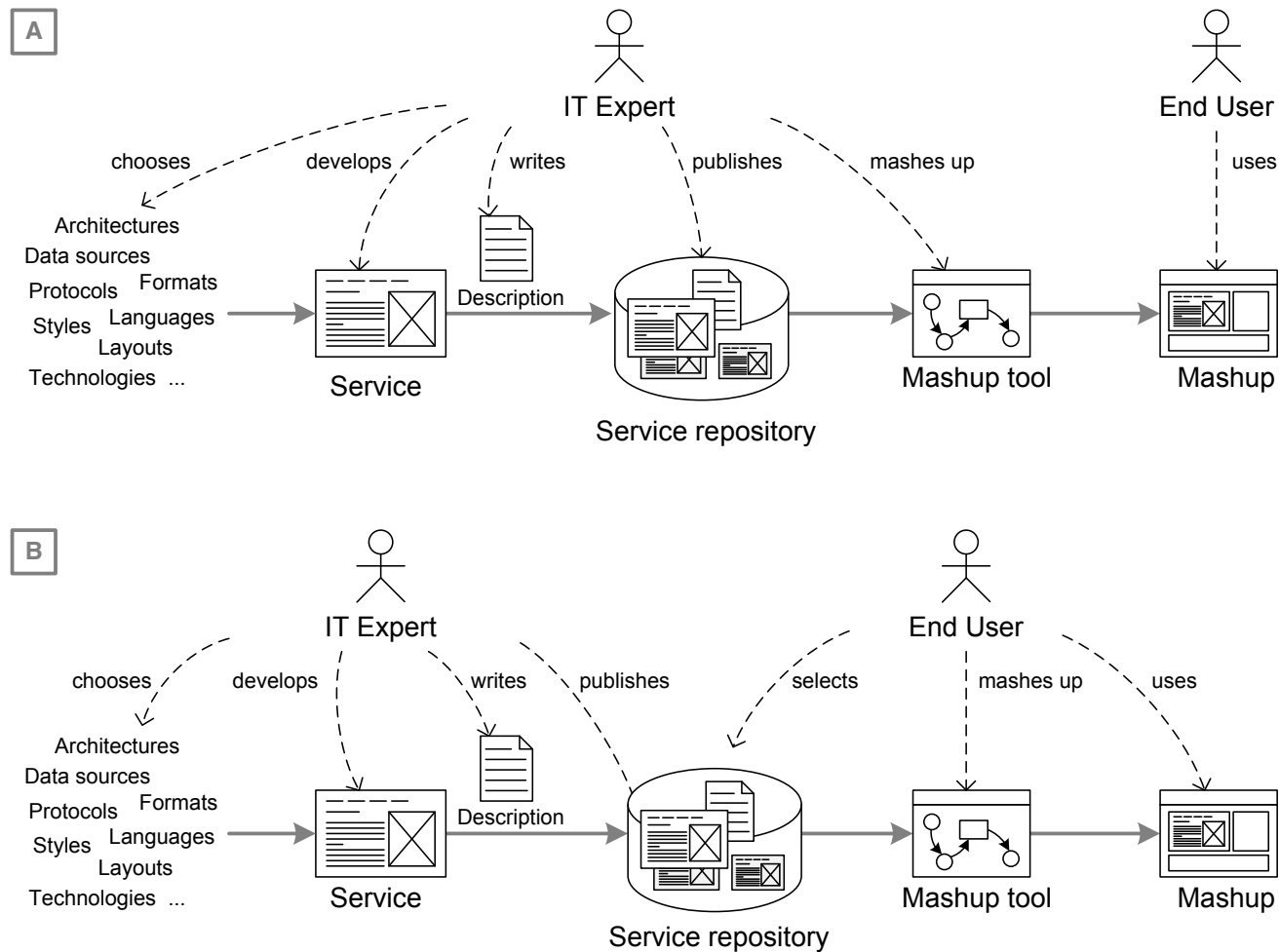


Fig. 9.1 Two main mashup development scenarios. (a) Expert developers exploit mashup tools “centrally” to deliver applications quickly. (b) Users exploit such tools to create mashups in a “distributed” fashion, starting from a set of ready services.

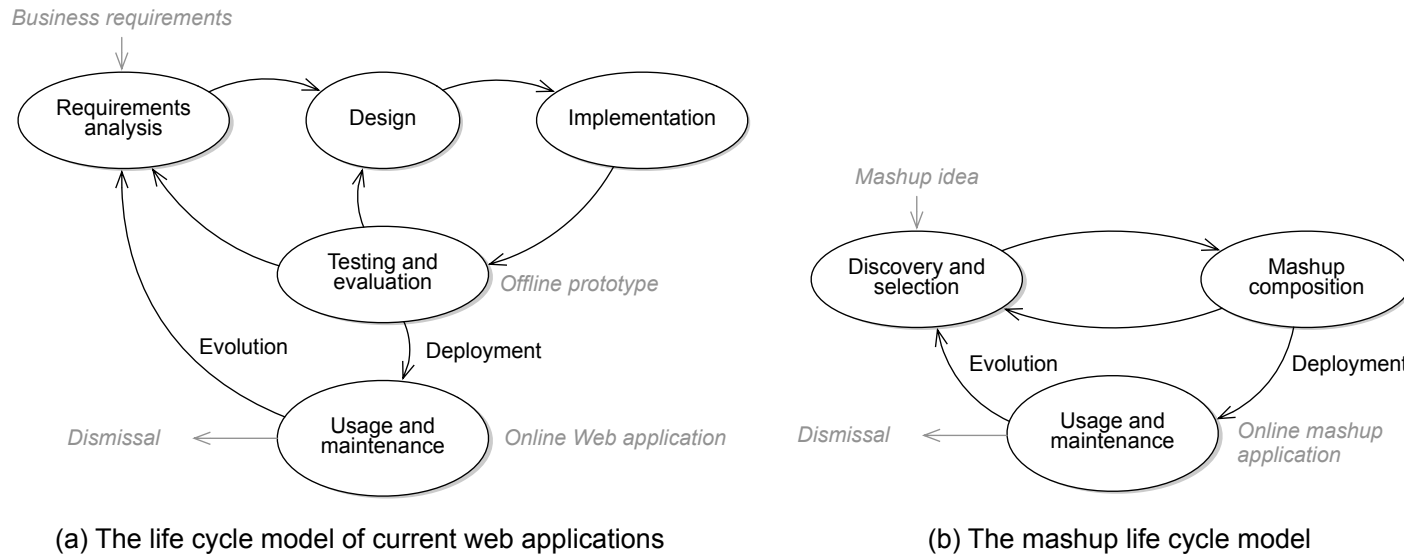


Fig. 9.2 Life-cycle models of (a) current Web applications and (b) mashups. The model for the end-user development of mashups presumes the availability of a dedicated mashup platform and toolkit, along with a set of open Web services that provide functionality and data.



Menu showing the list of available components. Users can add components in the mashups by dragging and dropping them into the interactive workspace

Once a component is added into the workspace, its UI is immediately displayed and its behaviour synchronized with the other components, according to “default bindings” based on component compatibility

The user can enrich the default synchronization behaviour, and define further component couplings by selecting possible behaviours that the two components have to show within the final application

Fig. 9.3 The WYSIWYG composition editor of the PEUDOM platform [58, 183].

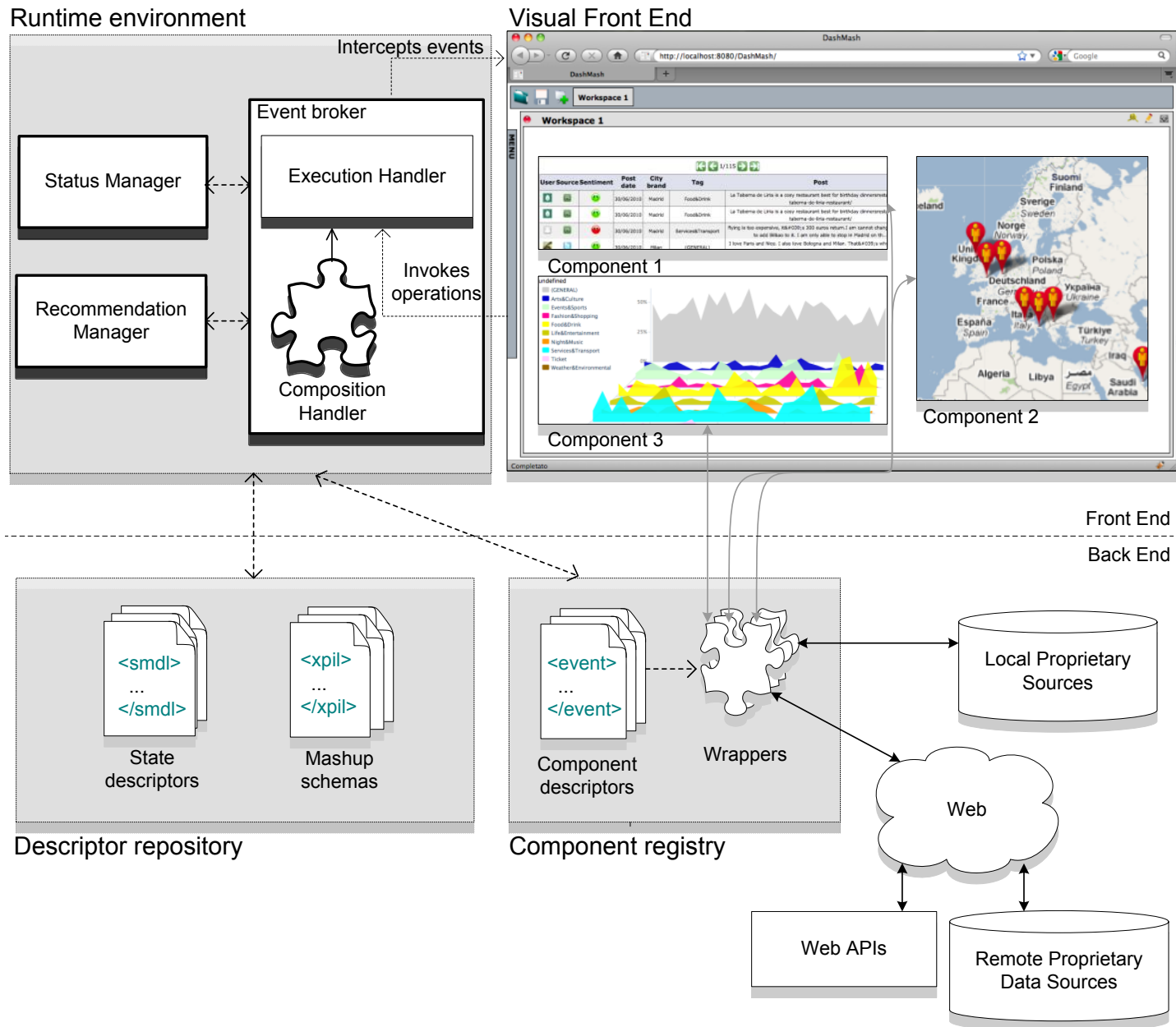


Fig. 9.4 Internal architecture for WYSIWYG composition in PEUDOM [58, 183].

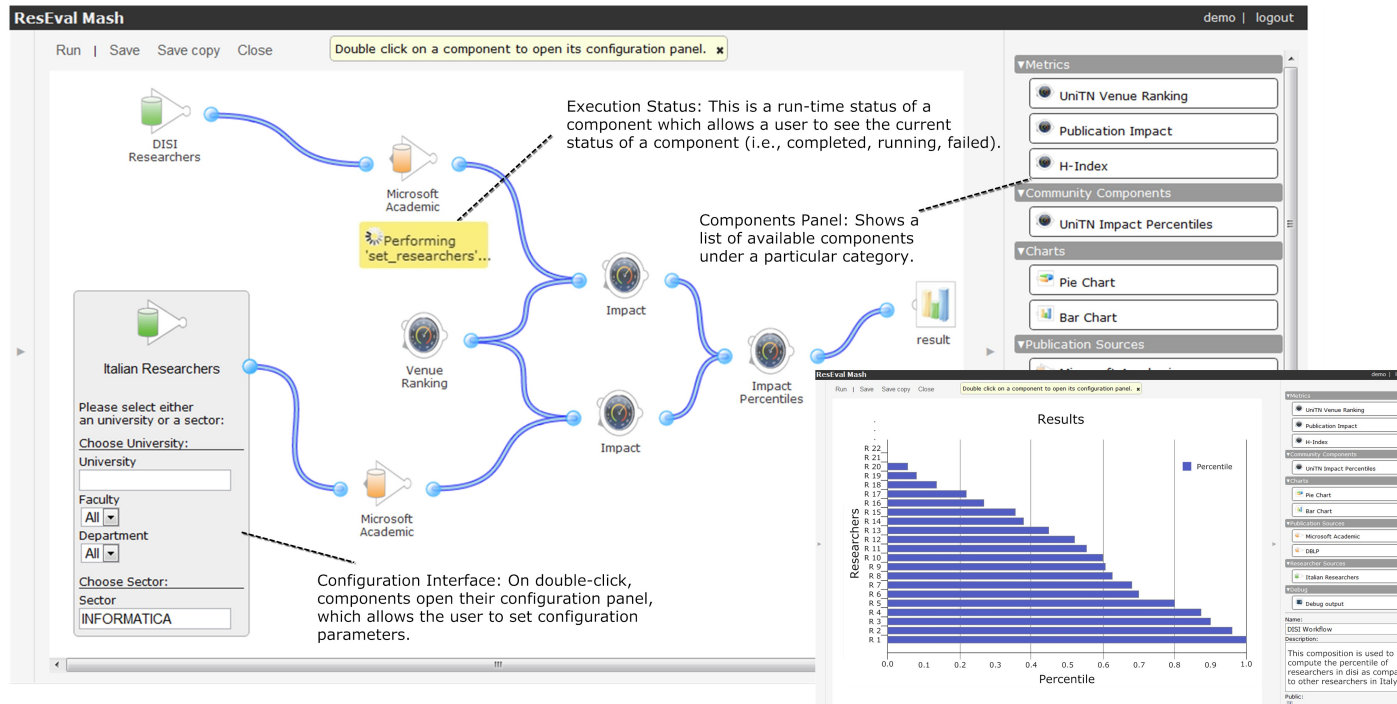
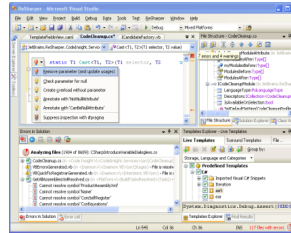


Fig. 9.5 Model of University of Trento's internal department evaluation procedure modeled in ResEval Mashup [155].

Professional developers



Integrated development environments (IDEs) for:

- Implementing software environments for other stakeholders
- Implementing visual templates

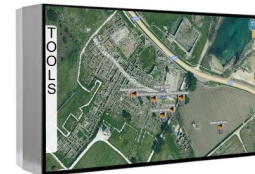
Domain experts in collaboration with service experts



Platform environment for:

- Service selection
- Service registration
- Definition of visual templates

End users



Platform environments for:

- Mashup creation
- Mashup use through different devices
- Mashup update and evolution

Fig. 9.6 A meta-design approach for mashup creation. The bottom layer outlines the environments for end-users, the middle and the top layers the environments for experts developers and domain experts who operate customizing the platform [18].

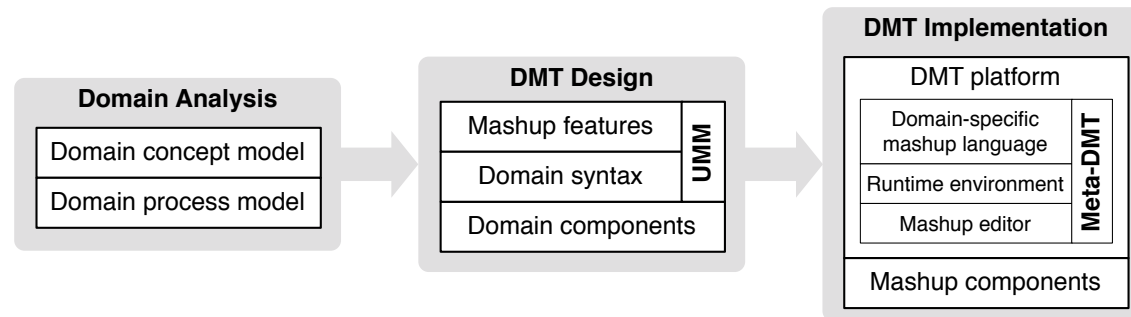


Fig. 9.7 Methodology for the development of domain-specific mashup tools with the Meta-DMT described in [253].

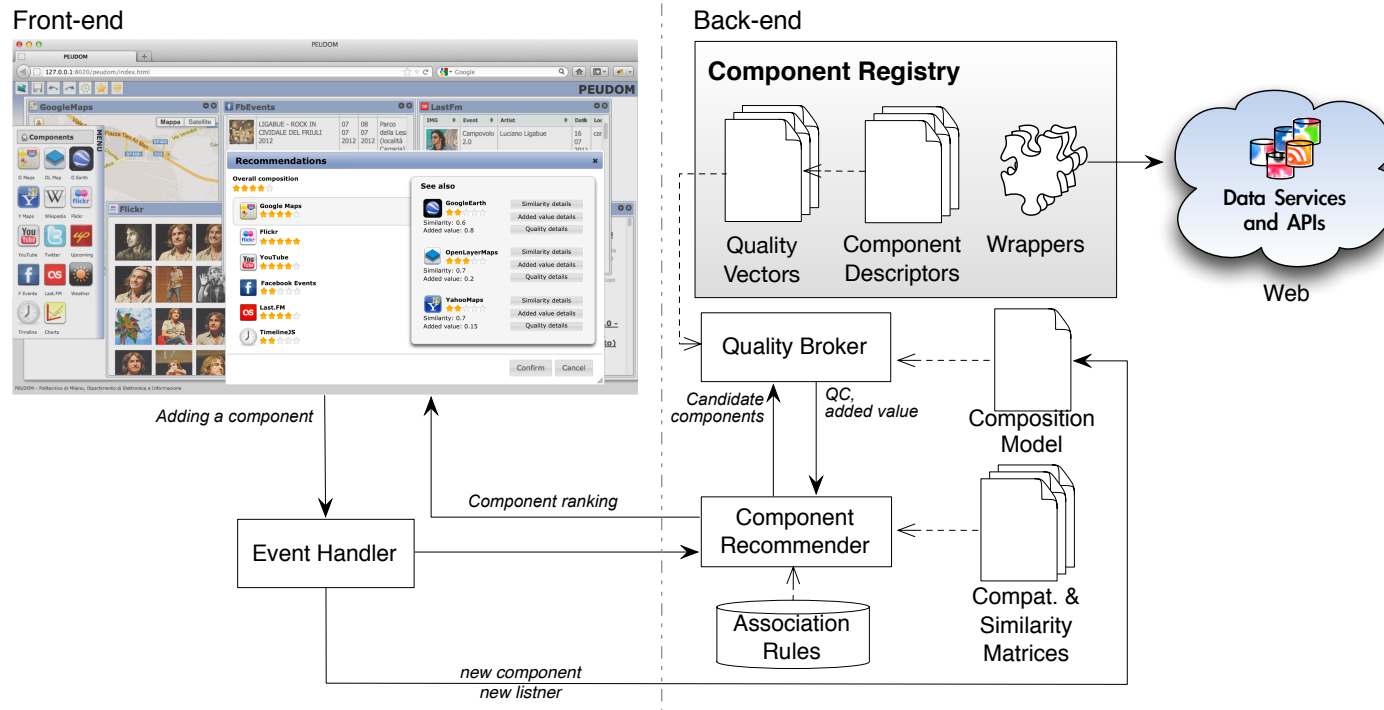


Fig. 9.8 Modules for quality-aware recommendations in PEUDOM [61].

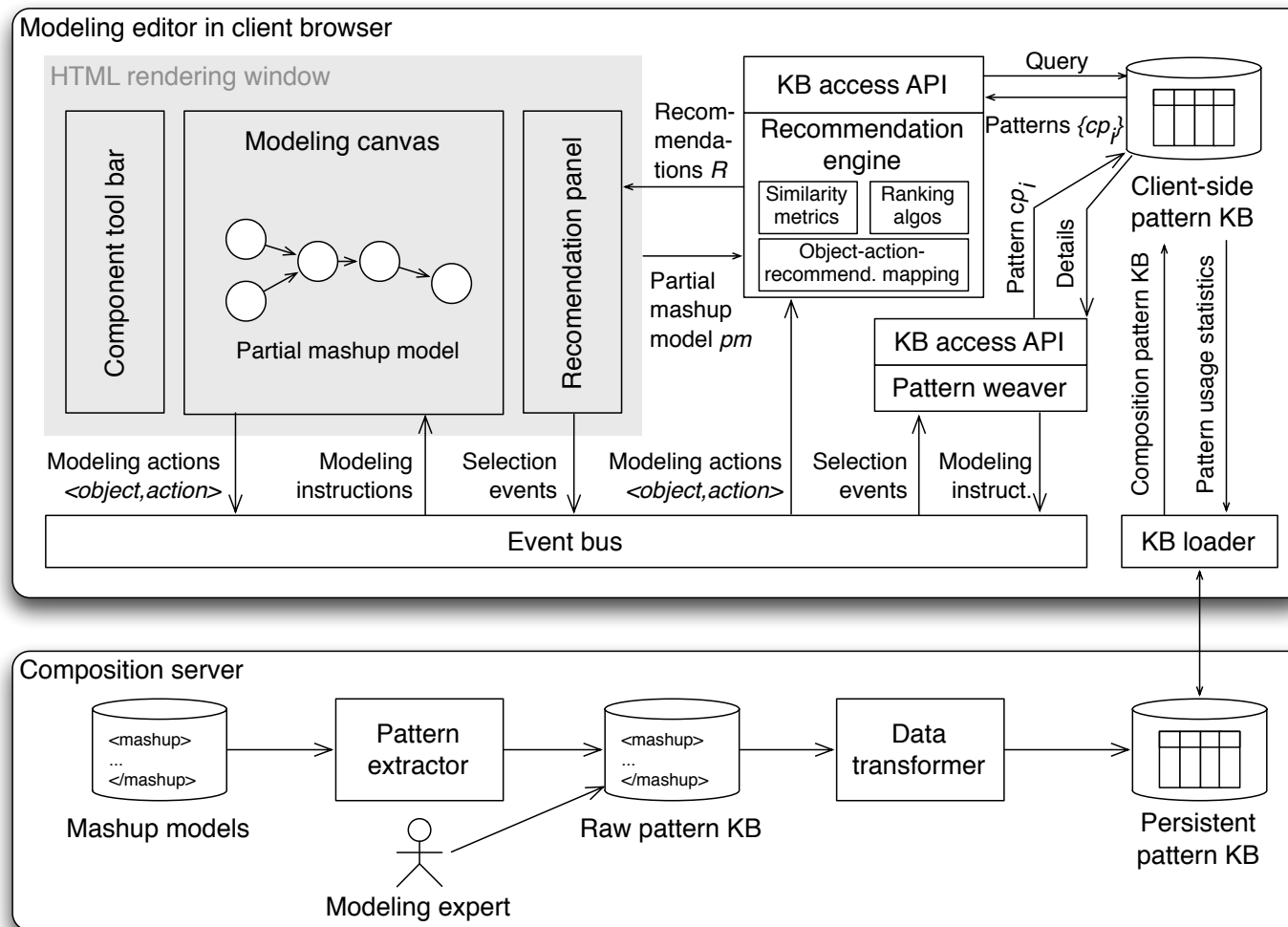


Fig. 9.9 Simplified architecture of the assisted modeling environment with client-side knowledge base and interactive recommender proposed in [73].

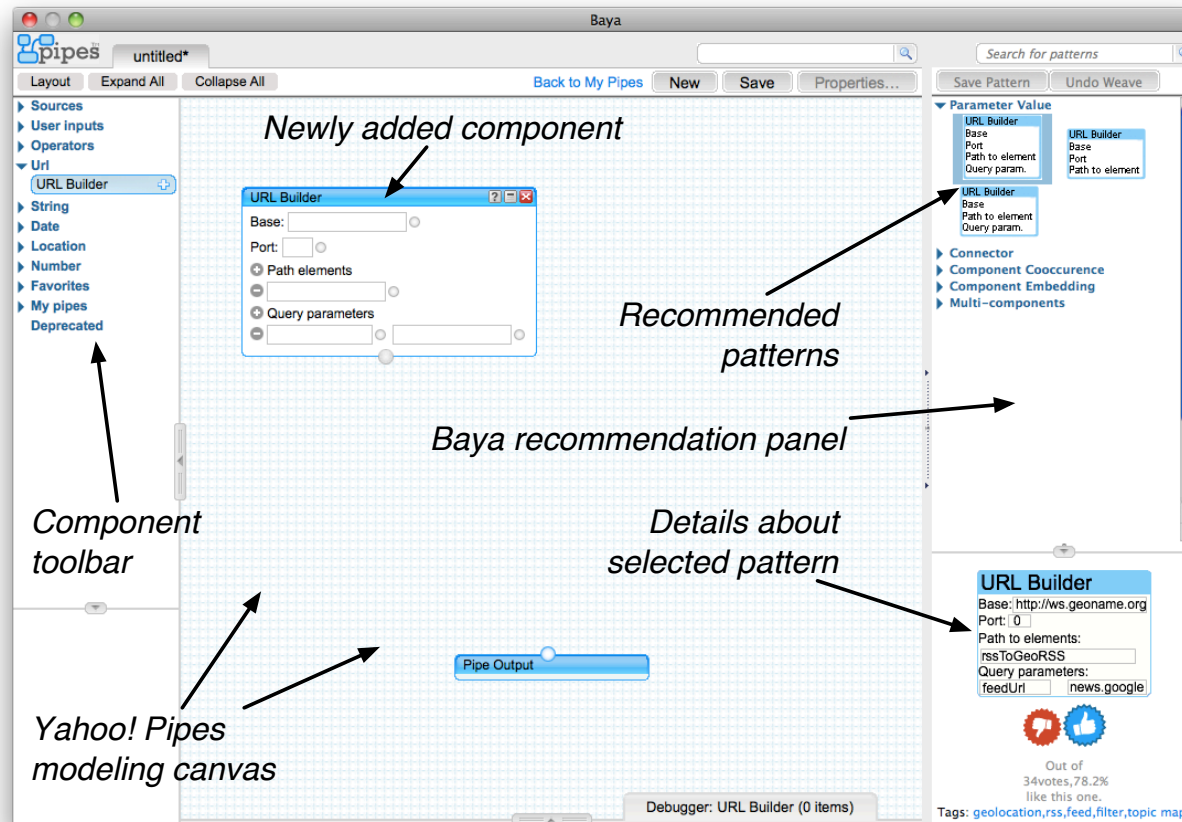


Fig. 9.10 Screen shot of the Baya plug-in for Yahoo! Pipes at work [74]: mashup model patterns are recommended in the panel at the right-hand side and woven into the model in the canvas by dragging and dropping them onto the canvas.