

Expectation-Driven Event Monitoring



Overview

Expectation-Driven Event Monitoring (EDEM) provides developers with a platform for creating software agents that are deployed over the Internet to collect application usage data and user feedback on a potentially large and ongoing basis.

EDEM can be used to illuminate how applications are used, to uncover mismatches in actual versus expected use, and to increase user involvement in the evolution of interactive systems.

Applicability

Researchers can use EDEM to gain insight into how applications are used in practice. Developers can use EDEM to make more informed design, impact assessment, and effort allocation decisions, ultimately leading to more cost-effective development of software that is better suited to user needs.

While EDEM currently monitors Java-based interactive applications, the underlying concepts can be applied to arbitrary software systems with “observable” event and state data.

Features

EDEM performs in-context data abstraction, selection, and reduction allowing meaningful information to be collected on a potentially large and ongoing basis over the Internet. Encapsulation of data collection code within software agents insulates users from updates to data collection:

Abstraction: low-level data may be flexibly related to higher-level concepts such as user interface and application features and users' tasks and goals.

Selection: data of interest is selected, and unnecessary data filtered, prior to reporting.

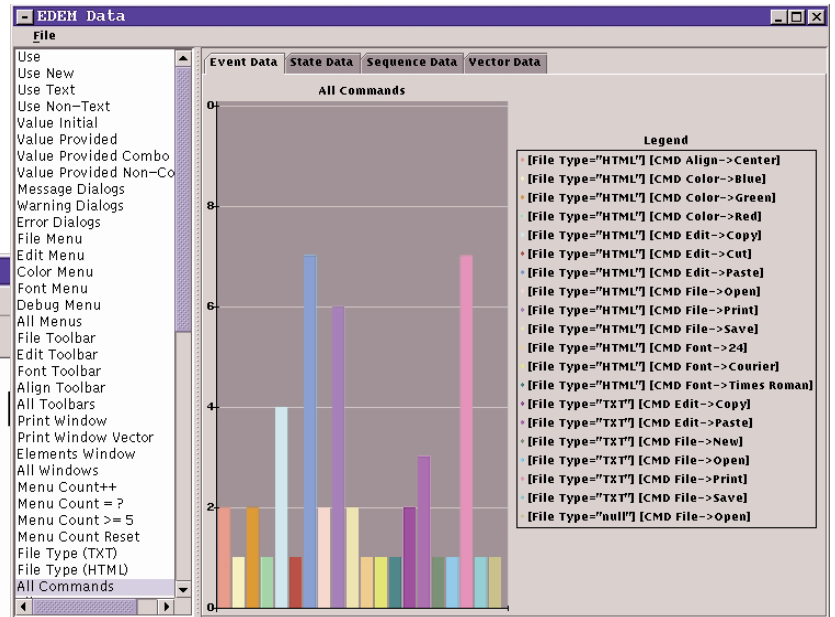
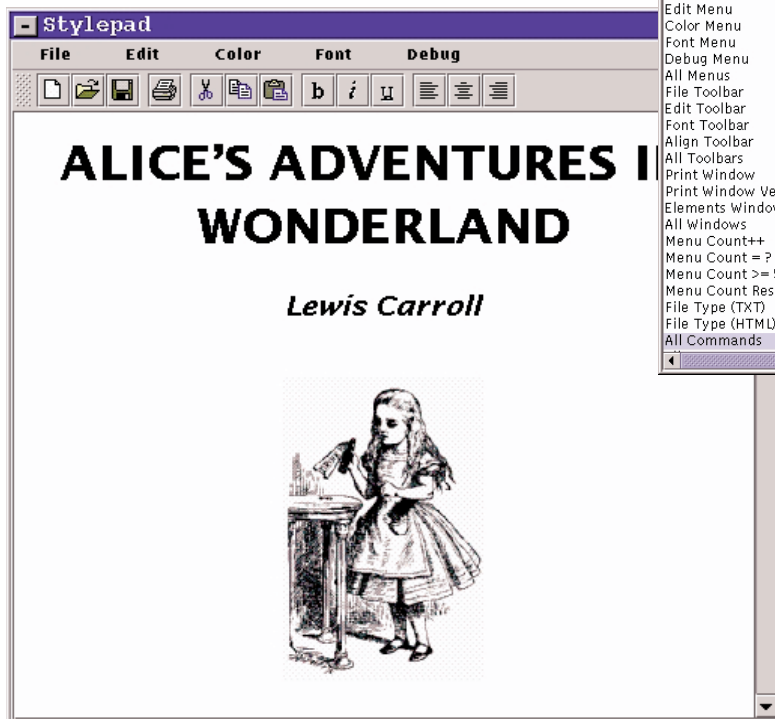
Reduction: pre-defined reduction algorithms are applied during data collection resulting in greatly reduced data reporting and post-hoc analysis needs.

Context: relevant user interface, application, artifact, and user state information may be captured in addition to events.

Evolution: data collection may evolve over time without impacting application deployment or use.



Below is a simple word processing application. EDEM (right) can be used to discover the frequency of feature use, help use, and errors; whether menus, toolbars, and dialogs are appropriately laid-out given the frequency and order in which users interact with them; and whether defaults in dialogs, user preferences, and basic application configuration are appropriate given user behavior.



Above is a real-time plot indicating the frequency of feature use based on the type of document being edited (left). In this example, color-related features are more frequently used in HTML documents than in non-HTML documents. Recording state information along with event data is useful in comparing patterns of use across multiple “modes” of use. Such data may be used to optimize or adjust the user interface for specific types of use, or to help focus development and evaluation effort on features associated with particular types of use (e.g., HTML authoring).

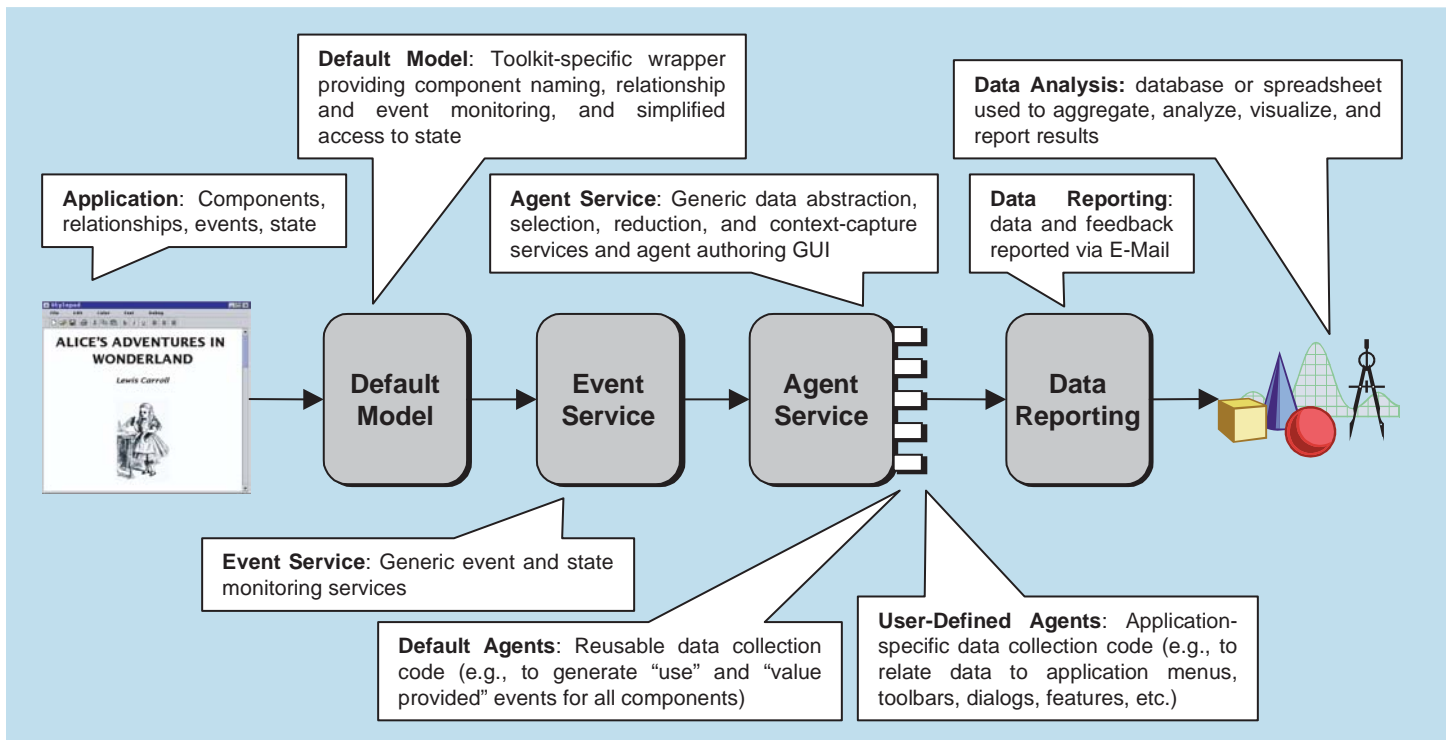
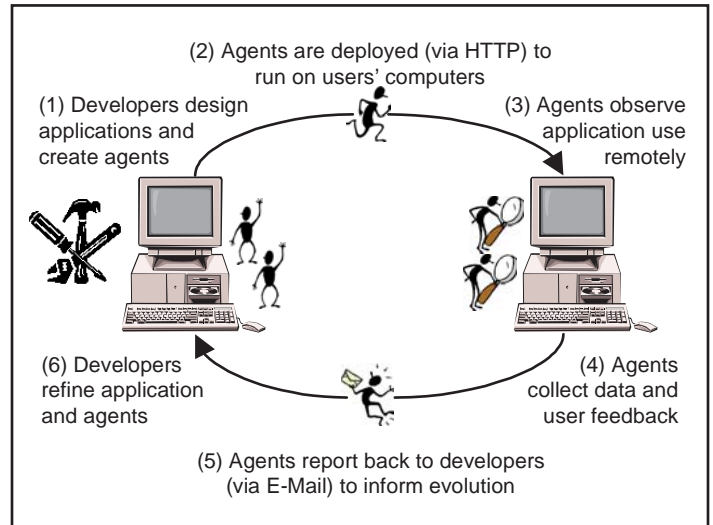
Approach

EDEM is based on the following basic principles:


Expectations: developers have expectations about application use that affect application design; and designs often embody expectations even when developers are not explicitly aware of them.

Mismatches: mismatches between expected and actual use indicate potential problems in design or use that may negatively impact application usability and utility.

Resolution: making expectations more explicit and observing use to compare it against expectations can help in identifying and resolving mismatches, thereby bringing expectations, and thus designs, into better alignment with actual use.



Concluding Remarks

 *Agent specifications are stored in ASCII and associated with URLs on development computers. URLs are passed as command-line arguments to applications of interest allowing up-to-date agents to be downloaded and instantiated on users' computers each time the application is run. Agent-collected data and feedback is reported to developers via E-Mail.*

Internet and Web-based software release practices have begun to blur the distinction between development and use. EDEM leverages this opportunity to enable usage- and usability-related information of much higher quality than currently available in beta tests to be collected on a much larger scale than currently possible in usability tests.

Contact Information

Professor Richard N. Taylor
Professor David F. Redmiles
Information and Computer Science
University of California
Irvine, California 92697-3425

{taylor, redmiles}@ics.uci.edu
949-824-{6429, 3823}
949-824-1715 (fax)

Freely Available Software

Information about the evolutionary design environments group, as well as its software, is freely available on the Web or by anonymous FTP:

<http://www.ics.uci.edu/pub/eden/>
<ftp://www.ics.uci.edu/pub/eden/>

This material is based upon work sponsored by the Air Force Materiel Command, Air Force Research Laboratory, and the Defense Advanced Research Projects Agency under contract number F30602-97-2-0021. The work is also supported by the National Science Foundation under contract number CCR-9624846. The content of the information does not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred.