

CLAIMS

What is claimed is:

We claim:

1. A system for capturing information from any complex web applications including a portal application and its associated subsystems, the system comprising:

a behavior tracking component configured to capture activities of a user when the user interacts with the portal application;

a performance monitoring component coupled to the behavior tracking component and configured to determine performance information relating to the activities of the user when the user interacts with the portal application; and

an auditing component coupled to the behavior tracking component and the performance monitoring component and configured to capture event details related to the activities of users when the user interacts with the portal application, wherein

the behavior tracking component, the performance monitoring component and the auditing component operate together in a framework and in a non-intrusive manner with regard to the portal application, and wherein the framework is coded using Aspect Oriented Programming (AOP), and wherein

the behavior tracking component, the performance monitoring component and the auditing component operate together in the framework

collecting data seamlessly from multiple threads spawned for efficient parallel processing of user's requests within the portal application.

2. The system of claim 1, wherein the behavior tracking component is further configured to collect all the activities that the user does on the portal associated with the portal application including at least login, logout, and page visits activities.

3. The system of claim 2, wherein the performance monitoring component is further configured to measure real time performance numbers of various subsystems of the portal application, including calculating performance numbers of the activities of the users as captured by the behavior tracking component.

4. The system of claim 3, wherein the performance monitoring component is further configured to provide drill-down of the performance numbers at various subsystems including at least portlets, Javasever Faces (JSF), database calls and remote called subsystems.

5. The system of claim 4, wherein the auditing component is further configured to capture at least event details of critical events and associate the event details with a user that is involved in the events

together with a time when the events occur, and wherein the remote called subsystems include web services.

6. The system of claim 5, further comprising an alert component in the framework and configured to communicate alerts and exceptions associated with the portal application to a recipient.

7. The system of claim 6, wherein the information captured by the behavior tracking component, the performance monitoring component and the auditing component is stored in a database associated with the framework, and wherein the system further comprises a reporting component to generate reports based on the information stored in the database.

8. The system of claim 1, wherein the framework is configured to start or stop operating with the portal application at runtime with minimal impact on operations of the portal application, and wherein the framework is configured to use asynchronous Java Message Service (JMS) queues.

9. A method for managing operations of a portal comprising:
identifying a first set of connection points available from a portal application where behavior related information can be captured;

configuring a first component of a performance, analytics and auditing (PAA) system to capture information from the first set of connection points;

identifying a second set of connection points available from the portal application where performance related information can be captured;

configuring a second component of the PAA system to capture information from the second set of connection points;

identifying a third set of connection points available from a portal application where auditing related information can be captured;

configuring a third component of the PAA system to capture information from the third set of connection points; and

storing the captured information in a database associated with the PAA system, wherein the framework is coded using Aspect Oriented Programming (AOP).

10. The method of claim 9, wherein the second component and the third component are configured to use the information captured by the first component to determine the performance related information and the auditing related information, respectively.

11. The method of claim 9, wherein the PAA system is coded using Aspect Oriented Programming (AOP) and configured to operate with the portal application in a non-intrusive manner.

12. The method of claim 9, wherein the first component is further configured to capture information relating to activities performed by the users on a portal associated with the portal application.

13. The method of claim 12, wherein the second component is configured to measure real time performance numbers of various subsystems of the portal application, including calculating performance numbers of the activities performed by the users as captured by the first component.

14. The method of claim 12, wherein the second component is further configured to provide drill-down of the performance numbers at the various the subsystems including at least portlets, Javasever Faces (JSF), database calls and remotely called sub systems.

15. The method of claim 13, wherein the third component is further configured to capture at least event details of critical events, and wherein the third component is configured to associate the captured event details with a user who is involved in the events together with time information when the events occur.

16. The method of claim 15, further comprising configuring a fourth component of the PAA system to communicate alerts and exceptions associated with the portal application to a recipient.

17. The method of claim 16, further comprising configuring a fifth component of the PAA system to generate reports based on the information stored in the database associated with the PAA system.

18. The method of claim 9, wherein the PAA system is configured to start or stop operating with the portal application at runtime with minimal impact on operations of the portal application.

19. The method of claim 18, wherein the PAA system is configured to use asynchronous Java Message Service (JMS) queues.

20. A computer-readable media that stores instructions, which when executed by a machine, cause the machine to perform operations comprising:

identifying a first set of connection points available from a portal application where behavior related information can be captured;

configuring a first component of a performance, analytics and auditing (PAA) system to capture information from the first set of connection points;

identifying a second set of connection points available from the portal application where performance related information can be captured;

configuring a second component of the PAA system to capture information from the second set of connection points;

identifying a third set of connection points available from a portal application where auditing related information can be captured;

configuring a third component of the PAA system to capture information from the third set of connection points; and

storing the captured information in a database associated with the PAA system, wherein the framework is coded using Aspect Oriented Programming (AOP).

CLAIMS

We claim:

1. A system for capturing information from any complex web applications including a portal application and its associated subsystems, the system comprising:

a behavior tracking component configured to capture activities of a user when the user interacts with the portal application;

a performance monitoring component coupled to the behavior tracking component and configured to determine performance information relating to the activities of the user when the user interacts with the portal application; and

an auditing component coupled to the behavior tracking component and the performance monitoring component and configured to capture event details related to the activities of users when the user interacts with the portal application, wherein

the behavior tracking component, the performance monitoring component and the auditing component operate together in a framework and in a non-intrusive manner with regard to the portal application, and wherein the framework is coded using Aspect Oriented Programming (AOP), and wherein

the behavior tracking component, the performance monitoring component and the auditing component operate together in the framework collecting data seamlessly from multiple threads spawned for efficient parallel processing of user's requests within the portal application,

wherein the framework is configured to start or stop operating with the portal application at runtime with minimal impact on operations of the portal application, and wherein the framework is configured to use asynchronous Java Message Service (JMS) queues.