



*Address z/OS auditing issues such as compliance, sam, change comparison, cross-site changes, IPL volume(s) and fully documented change management*

## **z/OS Auditing Issues addressed by eventACTION**

**eventACTION** and **ussACTION** are comprehensive event tracking products that can help you pro-actively manage and audit your z/OS environment with real time capture of any change, reference, or execution to any system level data. The reporting tools provide a complete and accurate picture of all changes made to the structure of the system, both for MVS and UNIX System Services changes.

### **What needs to be audited?**

It is essential that the auditors and the system programming personnel discuss and implement auditing strategies together, as each of these groups have a different view of the system and the importance of the various elements. In order to achieve the best results, elements such as programs, JCL, parameters, data etc. need to first be categorised into three main groups:

- Mission Critical** – If this data gets corrupted due to deletion, changes etc. it could cause the loss of the system, application or other essential parts of the system.
- Important** – Corruption or loss could lead to problems or loss of data.
- Unimportant** – Loss or corruption would not effect the day to day running of the system.

Once the system elements (Datasets, Members or for USS Directories and Files) that need to be audited have been identified, their tracking and backup requirements need to be determined. For example in Europe most banks are required to retain certain data for up to 10 years. If such an element were to be changed once a week, then at least 520 backups would have to be kept, naturally not all of them would need to be permanently online.

For each element or group of elements various decisions would need to be made, such as:

- How many STATistics (Information on when an element was changed and by whom) should be kept and how many should be held online. Only using STATs would be ideal for the "Unimportant" category, where you nevertheless would like to see what is being changed.
- How many BKUPS (the STAT information plus actual backups that can be restored or compared with the previous versions) should be kept and how many of these should be online. Here one needs to balance the amount kept online, with the system backup cycles and the need to be able to restore data to the point of impact. This is ideal for the "Important" category.
- Finally for the "Mission Critical" elements, one should turn on the "Change Management" options, whereby there are three levels of controls:
  - Change Request Required – No changes can be made to these elements without a change request being present.
  - Authorisation Required – Single or multiple authorisation by one or more groups. Unless the authorisations have been made, the change will be disallowed.
  - Change Scheduling – This allows the implementation of changes to be placed under the control of the "Change Group" who can then schedule the change to be made at a time where it will not impact production. If a change underlies scheduling it can only be implemented in the specified window.

Once all the definitions have been made, the "Change Management" implementation and activation can be set to a predetermined date by using the so called "Warn Mode". This allows users to become acquainted with the "Change Controls" in that the system will warn if the change request does not exist or has not been authorised, but will nevertheless allow the change to be made. On expiration of the "Warn Mode" date the controls will be strictly implemented and no further changes will be allowed unless the requirements have been met.

Please also be aware that tracking of changes will continue even if the **eventACTION** STC (Started Task) is stopped. However, it should be in the auditors interest to implement an alert if this task is stopped and then to determine why this was done.

Finally, the financial aspect of implementing **eventACTION** cannot be ignored. An internal or external audit of the systems should only take a fraction of the time it used to take, thereby potentially saving hundreds of man hours and thousands of Euro.

#### Some Auditing examples.

**Compliance for z/OS:** With Change Tracking and Control you can track changes in the z/OS systems area, prevent unauthorised or uncontrolled changes from being made in the systems area, and satisfy your auditors that all changes have been recorded and are fully documented. Only **eventACTION**'s unique tracking and management capabilities make certain that all changes are tracked and that no unauthorised changes bypassed the Change Management System/Controls. With the data collected and the Reporting facility you will have a secure and complete audit trail. Changes to the datasets in the dynamic in-storage z/OS system lists, i.e. APF, LINK, LPA and PARM list can be automatically tracked and controlled.

**Asset Management:** By defining major load modules, JCL procedures or TSO clists to the Reference Tracking by Member facility, **eventACTION** will track each access to these members telling you who accessed it and how many times, from which system, out of which library. Such information will keep you abreast of the usage of old, new, or test versions of products. By also defining the main product load modules to Product Execution Control, you can ensure that products are run where they are licensed to be run; if not, you can warn the user, terminate the execution of the product, or simply send a message to an administrator via email. It should therefore no longer occur that an OEM audits the use of their products and then demands payment of license charges for software having been used on non-licensed systems.

**Compare Utility:** The Compare Utility has a unique side-by-side display facility for pointing out differences between files. It can be used online to compare members of libraries or **eventACTION** backups of members. It can compare all or a subset of the members of two different libraries to indicate which members are the same or different. This utility can be used to investigate problems, to see how something was last changed, or to see how one system library may differ from another. In batch mode, entire volumes may be compared or sets of datasets via the catalog. This is an absolutely essential tool for auditors when they are trying to track down what changes were made that led to a system and/or application outage.

**Library Cleanup:** Reference Tracking of datasets at a member level allows you to see what members within a library have not been referenced for a period of time; it may now be safe to delete these members from a library. It has long been a problem for auditors to determine what elements of the system, such as program source, objects, listings, JCL, parameters, panels, CLISTs, REXX EXECs, etc. etc. are obsolete. Just the documentation alone for all the obsolete items can fill many cupboards, disks etc. Not only that but such extraneous data can increase disk usage, access times for current programs and waste system resources. With **eventACTION**, after one year of tracking this obsolete data can be deleted from the main libraries. For auditors this results in less data to examine and improvements in the system performance and resources.

**Cross-site Communication:** The LULU component of **eventACTION** allows communication between two **eventACTION** databases or Changeplexes but is not limited to only local site access. For instance, you can compare the contents of a system resident volume in one city to a volume in another city or you can define change requests for remote sites.

**Command Manager:** Operator commands can be defined to the Command Tracking and Control facility to track who is issuing certain commands, when are they being issued and from where such as TSO, Batch, or operator console. The tracking information is essential to the auditors if they need to track down who issued a command that led to the outage of an application or the whole system. Also, locating such a command takes seconds, rather than the hours that would be need to sift through the SYSLOG. This facility is also particularly useful for managing the use of the 'SET' command, which can easily lead to system problems if the changes have been incorrectly typed.

**eventACTION** includes the following.

**Change Tracker** automatically and transparently tracks and records all changes to defined data sets down to a member level, regardless of what program was used to make the changes.

**Change Manager** is used to control changes, according to criteria specified by you; ensures that any changes are logged / documented in a change request, and optionally, approved.

Unlike other change management products, **eventACTION** does not require the use of specific tools to make changes. It works transparently regardless of the tools used, so that all changes made are tracked, providing a comprehensive picture.

**Command Manager** allows an installation to track and/or control operator commands; provides capabilities similar to Change Tracker / Manager for operator commands. Since an operator command can implement a change, this is an important control point.

**Reference Tracker** allows an installation to track all references to defined data sets and PDS members; and can be used for library cleanup, program usage measurement and product execution control.

**ussACTION** provides the **Change Tracker**, **Change Manager** and **Reference Tracker** in the same form as **eventACTION** for z/OS UNIX System Services.

With its numerous other features (such as a unique side-by-side compare utility, automatic batch job scheduling, extensive reporting functions, and flexible backup/recovery options) **eventACTION** is a self-contained and fully integrated management solution to provide dynamic change tracking, control, and distribution for single or multiple site MVS systems.

### **Summary:**

At the end of the day **eventACTION** provides the auditor with:

- System controls to assist in z/OS compliance
- Complete / secure audit trail
- Basis for software asset management
- Assure Software providers that their products only run on licensed systems
- Powerful compare facility
- Reporting for all data thru online, batch, email, scheduled
- Investigate 'incidents' real-time (changes, program usage, operations)
- Repository for z/OS system data



Copyright © 2010 Action Software GmbH  
May not be reproduced without permission.  
All rights reserved.

If you want more information about **eventACTION** please contact:

**Action Software GmbH**  
**Alte Steinhäuserstr. Im CH-6330 Cham, Switzerland**

**Tel.: +41 41 748 6266**

**Fax: +41 41 748 6267**

**E-Mail: Marketing(at)actionsoftware.ch**