

XINFO

SMF Historie

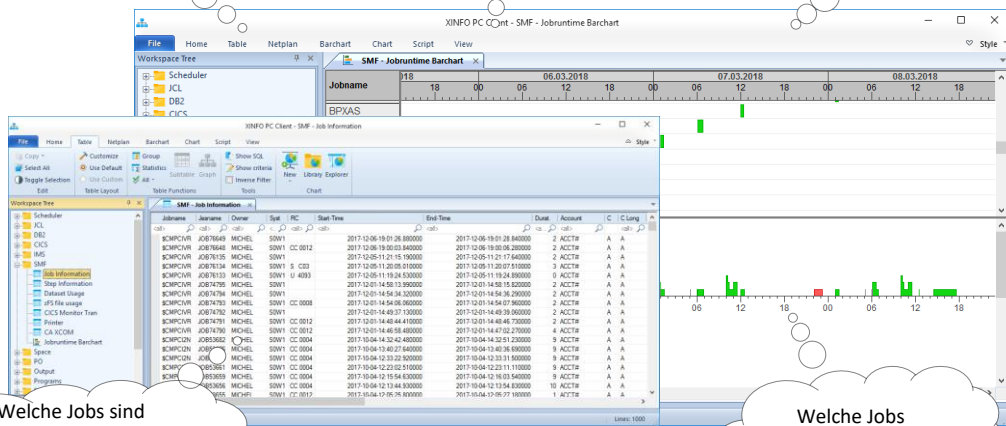
Üblicherweise werden Daten im Produktionsbetrieb über Jahre archiviert: Joblogs, Listen und sonstige Ausgabedateien. Outputmanagement und Joblog-Archivierungssysteme erledigen diese Aufgabe zuverlässig.

Wenn aber Joblaufzeiten, Returncodes, Dateizugriffe und CPU-Verbrauch abgefragt oder statistisch ausgewertet werden müssen, bieten diese Systeme keine ausreichende Unterstützung.

Welches Programm hat die Datei X.Y.Z geändert?

Welche Jobs enden nicht mit RC=0?

Welche Programme benötigen mehr CPU-Zeit als der Durchschnitt?



Welche Jobs sind gestern zwischen 03:00 und 04:00 gelaufen?

Welche Jobs laufen länger als eine Stunde?

XINFO - Das IT-Informationssystem

Selbst einfachste Fragen wie z.B. "Welche Jobs sind gestern zwischen 03:00 und 04:00 gelaufen", "Welches Programm hat die Datei X.Y.Z geändert" oder "Welche Jobs laufen länger als eine Stunde" können nicht oder nur mit aufwändigen Suchaktionen beantwortet werden.

XINFO analysiert den täglichen SMF-Dump und archiviert alle notwendigen Step- und Jobinformationen. Darüber hinaus werden die Dateizugriffe ausgewertet. Ein intelligentes "Cleanup"-Programm sorgt dafür, dass die Daten nicht unnötig lange gespeichert werden.

Welche SMF Daten werden ausgewertet?

- Job/Step Laufzeitinformationen
 - Wann wurde ein Job/Step auf welchem System submitted, wann ist er gestartet und mit welchem Returncode hat er geendet
 - Wie lange ist der Job/Step gelaufen, wie viel CPU-Zeit und EXCP (I/O) wurde verbraucht
 - Gab es unerwünschte Behinderungen, z.B. waiting for dataset (ENQ), allocation oder Jes initiator
 - Unter welchem RACF-User und welcher Account-Nummer ist der Job gelaufen
- Datei Verwendung
 - Wann wurde von welchen Job/User/STCs auf welche Datei zugegriffen
 - Timestamp der Allokierung und Art des Zugriffs: Catalog, Delete, Read, Update
 - Programm, DD-Name, Disposition
- CICS Monitor Transaktionen
 - Wann ist eine Transaktion gestartet und mit welchem Returncode geendet
 - Von welchem Terminal mit welcher Userid
 - Wie lange war die Transaktion aktiv, wie viel CPU-Zeit und Speicher wurden dabei benötigt

Anwendungsbeispiele:

- Kontrolle des Batchablaufs
 - Wann lief/läuft (normalerweise) ein Job, wann wird eine Datei verarbeitet?
- Analyse von Fehlersituationen
 - Welche Jobs waren in einem bestimmten Zeit aktiv, gab es in der Zeit bestimmte Auffälligkeiten, viele Wait-Jobs, hoher CPU-Verbrauch von bestimmten Jobs, Häufung von Returncodes?
 - In welcher Reihenfolge wurde eine Datei verarbeitet, wann erfolgten updates, welche GDG-Generation wurde von welchen Jobs verarbeitet?
- Analyse von bestimmten (kritischen) Jobs oder Transaktionen
 - Wann läuft der Job wie lange, CPU-Verbrauch?
 - gibt es Auffälligkeiten, z.B. sind Freitags die Laufzeiten besonders lang?
 - Zeigen Optimierungsmaßnahmen Wirkung, z.B. geht die Laufzeit oder CPU-Zeit zurück?
- Ermittlung von Problemstellen
 - Welche Jobs/Programme laufen länger als x Stunden/Minuten, oder brauchen mehr als x CPU-Zeit?
 - Gibt es eine Häufung von Wait-Situationen?
- Erstellung von Ad hoc Statistiken
 - Wie viel Jobs liefen an einem Tag?
 - Wie viele okay und wie viele endeten in error ?
 - Wie oft wurde eine CICS-Transaktion aufgerufen, wie hoch war dabei der CPU-Verbrauch?