

Event Parser für Adabas Auditing

Was ist was?

- **Adabas** – ein nichtrelationales Database Management System (DBMS). Eine Adabastabelle wird auch „File“ genannt
- **Adabas Auditing for Mainframe (ALA)** – erlaubt die Protokollierung von Zugriffen auf Adabas Files
- **Event** – ein von ALA erkanntes Ereignis. Ein „Event Message“ mit allen wichtigen Daten zum Ereignis wird auf eine Message Queue geschrieben

Mainframes sind leistungsstarke Großrechner für geschäftskritische Anwendungen in Unternehmen

Message Queue - ein Zwischenspeicher für Nachrichten, die zwischen Anwendungen ausgetauscht werden

Projektbeschreibung

Ein Java API, das binäre Eventdaten verarbeitet und in eine JSON-Struktur konvertiert.

Motivation

- Aktuelle Daten von ALA werden in sequenziellen Dateien gespeichert.
 - Drittanbieter überarbeitet diese Dateien und gibt die Ergebnisse frei.
 - **Problem:** Dieser Prozess ist ineffizient und erlaubt keine Echtzeit-Datenverarbeitung. **Lösung:** Event Parser entwickeln, der direkt aus der Message Queue liest
- Daten in der Message Queue sind binär und für Mainframe-Computer geeignet.
 - **Problem:** Binäre Datenformat nicht für uns lesbar.
 - **Lösung:** Event Parser, um binäre Daten zu verarbeiten.

JSON ist ein Datenformat, das für uns und für Computer lesbar ist

Docker ist eine Technologie, die es erleichtert, Anwendungen in kleinen, isolierten Paketen (Container) auszuführen und zu verwalten

```
Buffer of size: 1808
0000: e4c1c2c8 00000710 00000040 f0f10001 .....@... UABH..... 01..
0010: de59ac25 f2e8e006 c1d5e2c5 d9e5c5d9 .Y.%.....2Y\ANSERVER
0020: 00000003 00005676 00000000 00000000 .....Vv.....
0030: 00000000 00000000 00000000 00000000 .....
0040: e4c1c2c9 00000040 00000040 00000000 .....@...@... UABI...
0050: d4c5e3c1 00005673 00000000 00000001 .....Vs... META...
0060: de59ac20 f276a213 de59ac25 f2d46607 .Y. .v...Y.%..f. ...2.s...2M..
0070: e2e4c2e2 e8e2f240 00000000 00000000 .....@...@... SUBSYS2
0080: e4c1c2c4 000003f0 00000030 000003c0 .....0... UABD...0...
0090: 00000001 c7c6c6e3 00000000 00000000 .....GFFT...
00a0: da12c534 d3400000 00000000 00000000 .....4.@... ..E.L
00b0: e4c1c4c6 000003c0 00000080 00000340 .....@ UADF...
00c0: f0f10001 000d0040 0000ffff ffe00006 .....@1...
00d0: c3d3c9c5 d5e34040 40404040 40404040 .....@00000000 CLIENT
00e0: 40404040 40404040 40404040 40404040 @0000000000000000
```

Hexadezimale Darstellung

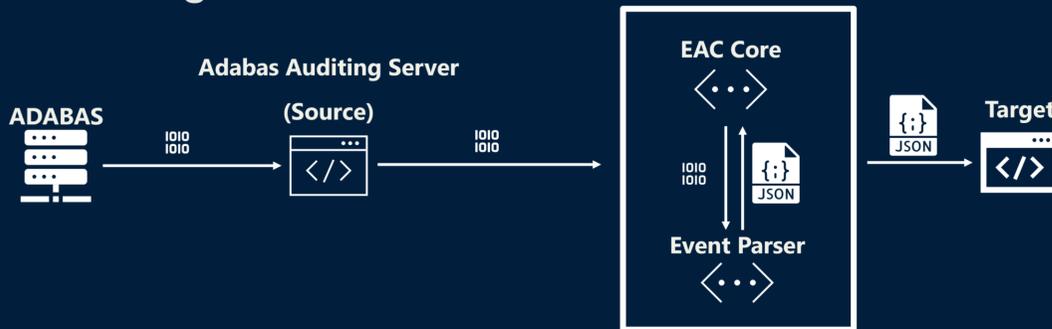
Dekodierte Darstellung

Teil einer Nachricht aus der Message Queue

```
{
  "UABHASID": 22134,
  "UABHTIME": "2024-02-19T14:40:48.468+0000",
  "UABHNUCI": 0,
  "UABI_ITEMS": {
    "UABIDBID": 22131,
    "UABINUCI": 0,
    "messageType": "metadata",
    "UABIDCNT": 1,
    "UABILITY": "META",
    "UABIPTIM": "2024-02-19T14:40:48.457+0000",
    "UABIITIM": "2024-02-19T14:40:43.211+0000",
    "UABISNAM": "SUBSYS2",
  }
}
```

Teil des JSON-Outputs

High-Level Architektur



Event Parser

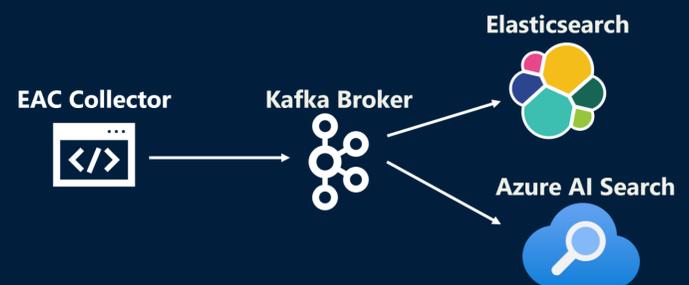
- Custom Java Annotations benutzt, um Struktur von binären Messages zu definieren
- REST API, um Metadaten von Adabas Files zu verwalten
- Unit + Integrationstests mit Mockito (Java Testframework)
- Github Actions benutzt, um Tests und das Bauen von Docker Images zu automatisieren
- Hohe Skalierbarkeit mit Docker – mehrere Container parallel laufen lassen, um schneller zu verarbeiten

Programmaufbau

- **EAC Collector** – Java Kernprogramm. Arbeitet **Pluginbasiert**: die Quellen (Source) und Ziele (Target) können durch ein paar Codezeilen schnell ersetzt werden
- **BrokerAudit** – Source Plugin, liest Daten aus der Message Queue
- **Event Parser** – verarbeitet die binären Daten und gibt sie in einem JSON-Format zurück. Wird im EAC Core aufgerufen
- **Kafka** – Target Plugin, sendet die JSON-Daten an einen Kafka Broker
- **Elasticsearch** – Target Plugin, sendet die JSON-Daten an ein Programm, dass die Visualisierung von den Daten ermöglicht

Apache Kafka ist eine Technologie, die verwendet wird, um große Datenmengen in Echtzeit zu sammeln, zu verarbeiten und zu speichern.

Kafka Connect Architektur



Kafka Connect

- Ein Framework, das die Integration von externen Datenquellen und -zielen mit Apache Kafka erleichtert.
- Besteht aus Connectoren, die Daten zwischen Quellen/Zielen und Kafka übertragen, und einem Worker, der Connectoren ausführt und verwaltet.
- Kafka Connect bietet Skalierbarkeit und Hochverfügbarkeit

Weitere Schritte

- Weitere Ziele mit Kafka Connect
- Azure AI Search – ein Microsoft Service, der KI-Algorithmen und Funktionen benutzt, um komplizierte Suchfunktionen auf die Daten zu ermöglichen
- OpenAI Chatbot Training – einen Chatbot trainieren, der als Helfer für die Daten dient



Elasticsearch Demo

