Pattern matching in DAME using Grid enabled AURA technology

Robert Davis, Bojian Liang, Mark Jessop,
Andy Pasley and Jim Austin

Advanced Computer Architectures Group

Department of Computer Science

York, UK

Overview

- DAME
- The data
- AURA
- Performance
- AURA-G
- Data Encoding
- Tool
- Next stages

DAME

- Aims to show how Grid can support diagnostics and prognostics
- Undertaken on Rolls Royce Aero-engine vibration and performance data within a Grid frame work
- Detailed paper in already presented

The pattern matching problem

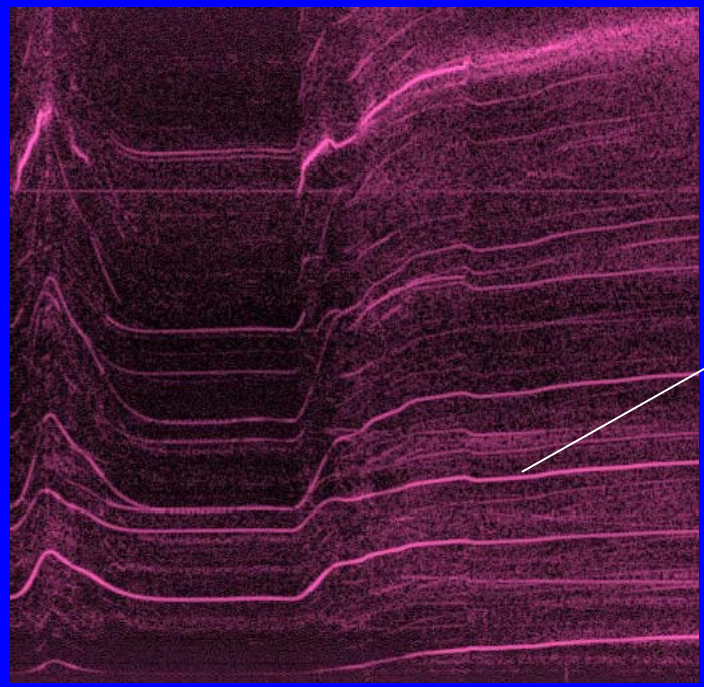
- Vibration data from all engines in flight
- *Detect*
  - events on that engine that look unusual
- *Search*
  - for similar events on other engines that are known about or not known about
- *Reason*
  - Using historical data that is associated with past similar events

Solutions

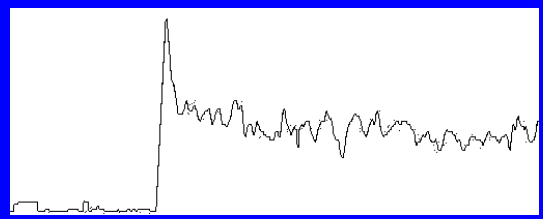
- Detect:
  - Uses a QUOTE on wing statistical classifier system – Oxford University
- Search:
  - Uses AURA pattern matching methods to search large vibration data sets - York
- Reason:
  - Uses CBR tools based on existing technology - Sheffield

The data

- Engines can produce 1Gb per engine per flight.



Tracked orders

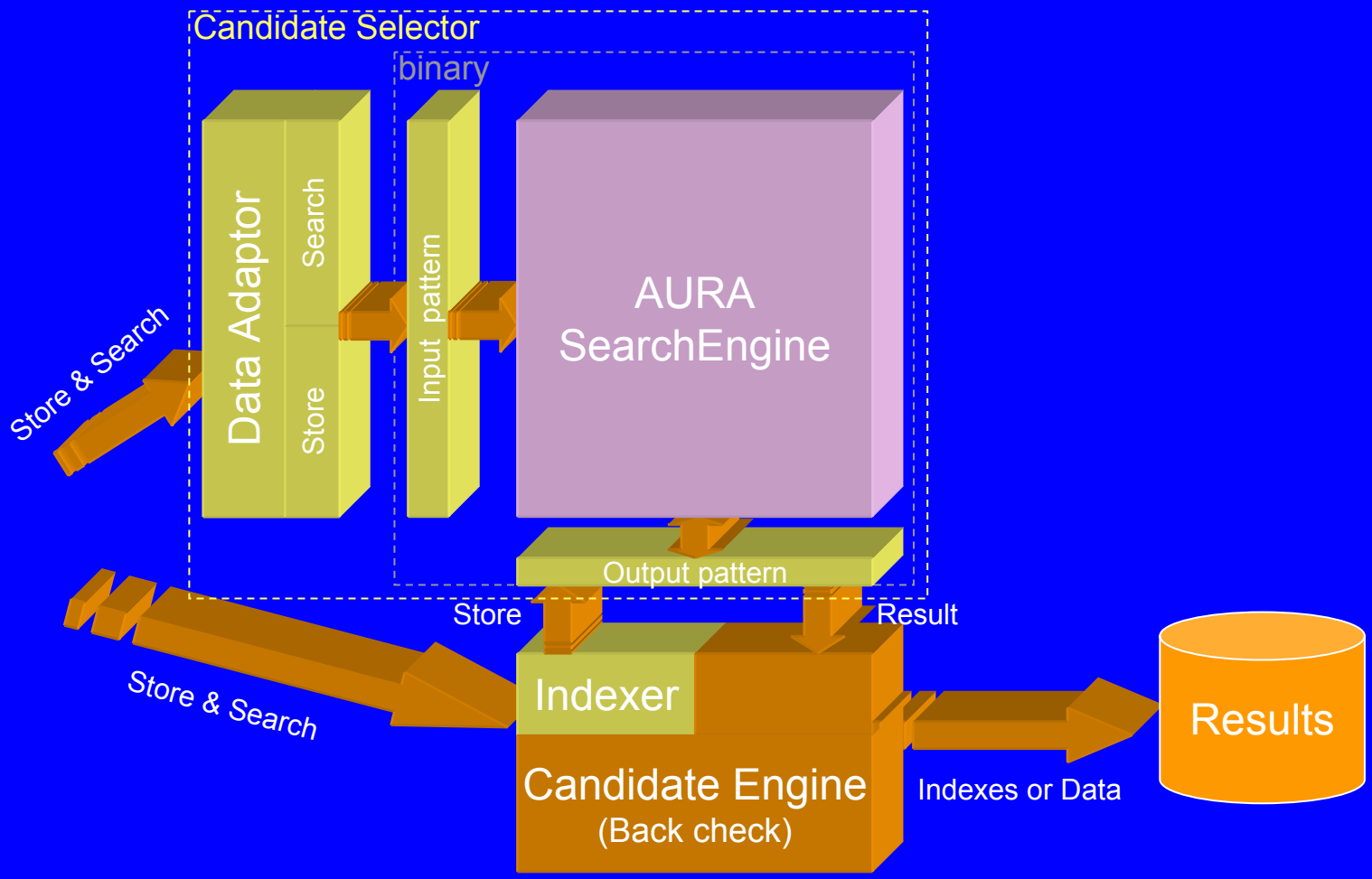


Searching the data

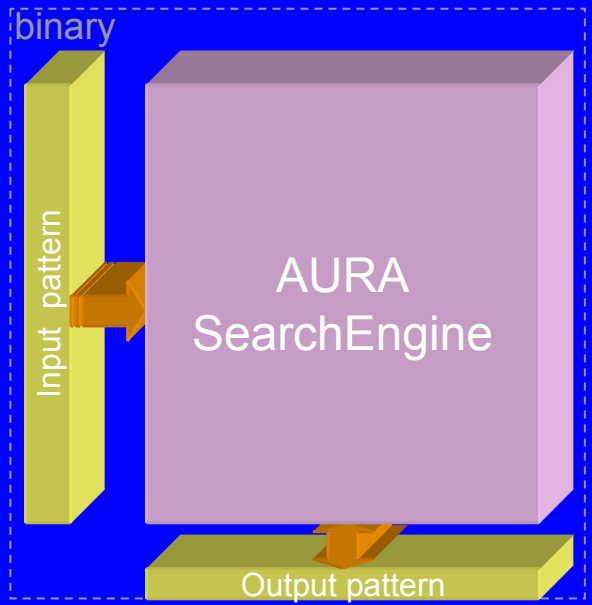
- Large amounts of noisy signal data.
- AURA provides
  - Proven technology for searching large data sets
  - Ability to scale and maintain performance
  - Can be parallelised

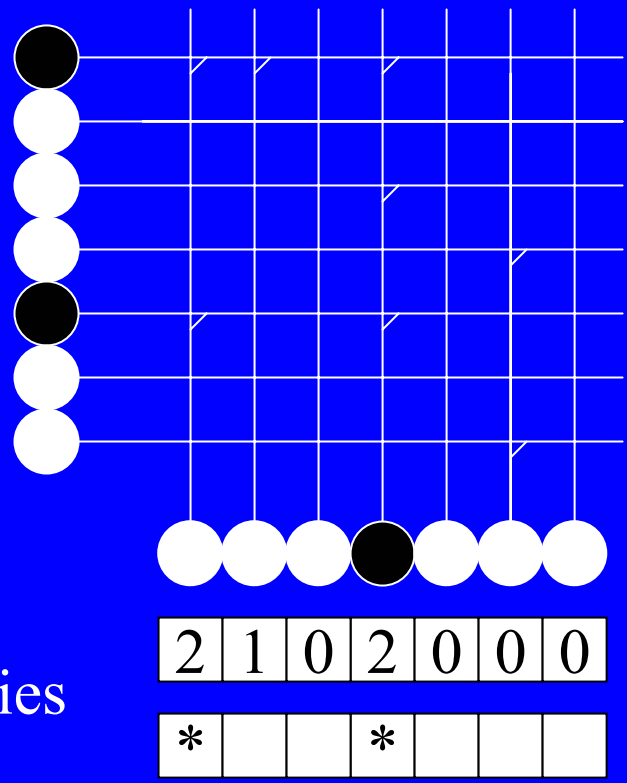
AURA

- Takes vectors and compares them to stored examples
- Uses bit level comparison methods and binary matrix operations.


Candidate Selector
binary
Data Adaptor
Search
Store
Input pattern
AURA
SearchEngine
Store & Search
Output pattern
Store
Result
Store & Search
Indexer
Candidate Engine
(Back check)
Indexes or Data
Results

Data storage & recall

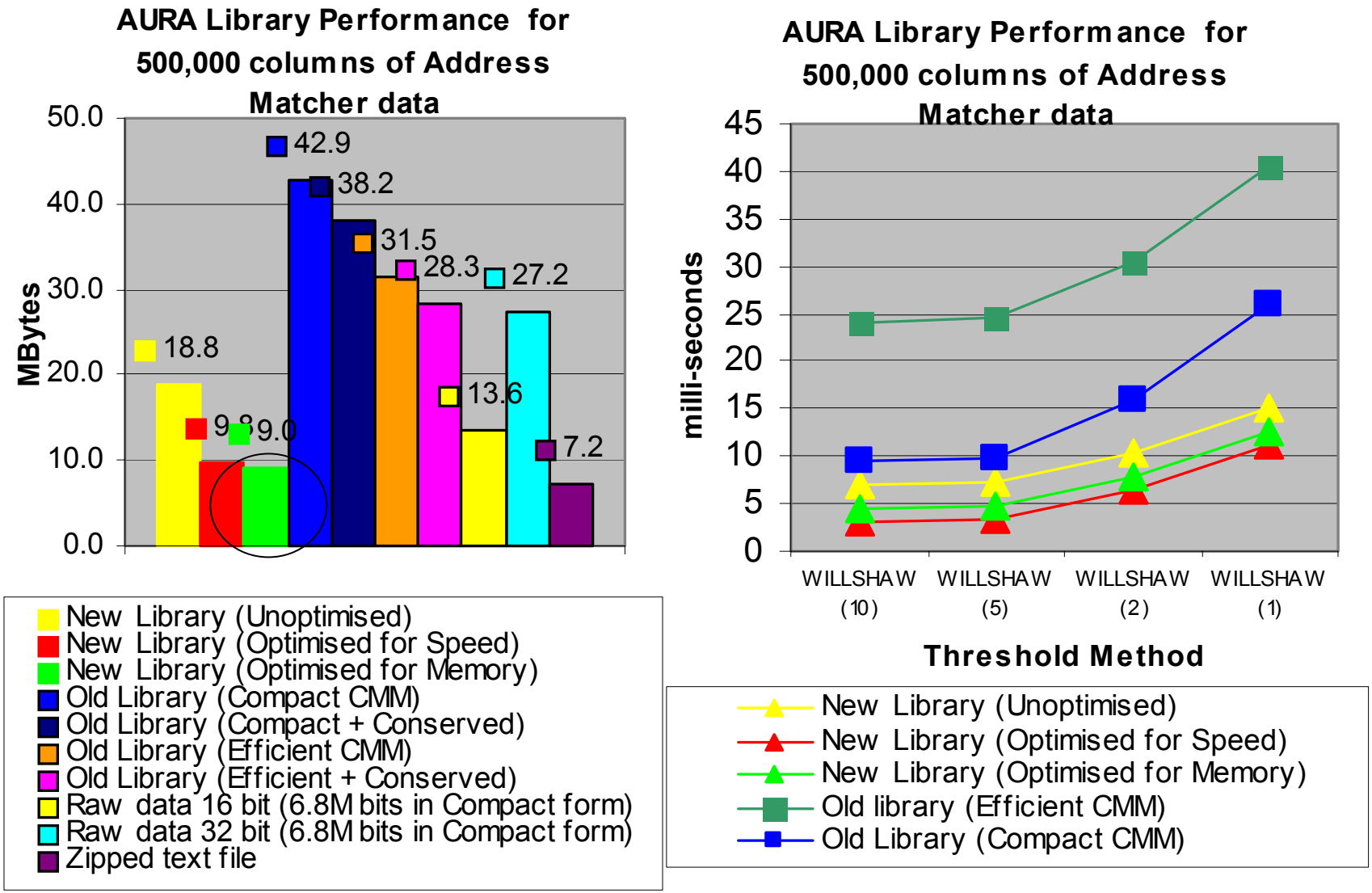




Correlation Matrix Memories

AURA software

- AURA redesign
  - To allow use within the Grid
  - To improve performance of the AURA library in terms of both memory usage and search times.
  - To make the library easier to use.
  - To engineer the library to commercial software standards.


AURA Library Performance for 500,000 columns of Address Matcher data
MBytes
50.0
40.0
30.0
20.0
10.0
0.0
18.8
9.8
9.0
42.9
38.2
31.5
28.3
27.2
13.6
7.2
New Library (Unoptimised)
New Library (Optimised for Speed)
New Library (Optimised for Memory)
Old Library (Compact CMM)
Old Library (Compact + Conserved)
Old Library (Efficient CMM)
Old Library (Efficient + Conserved)
Raw data 16 bit (6.8M bits in Compact form)
Raw data 32 bit (6.8M bits in Compact form)
Zipped text file
AURA Library Performance for 500,000 columns of Address Matcher data
milli-seconds
45
40
35
30
25
20
15
10
5
0
WILLSHAW (10)
WILLSHAW (5)
WILLSHAW (2)
WILLSHAW (1)
Threshold Method
New Library (Unoptimised)
New Library (Optimised for Speed)
New Library (Optimised for Memory)
Old library (Efficient CMM)
Old Library (Compact CMM)

AURA-G

- OGSI GT3
- Supports
  - Multi-users
  - Multi-memory systems

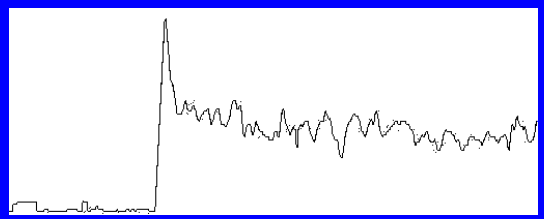
- Layers
  1. First is the client side
  2. Globus service management
  3. Local high performance connectivity –
     - cortex-1 clustering software
     - PRESENCE II hardware
  – We have an AURA service
     - CMM, pre-process, backcheck services

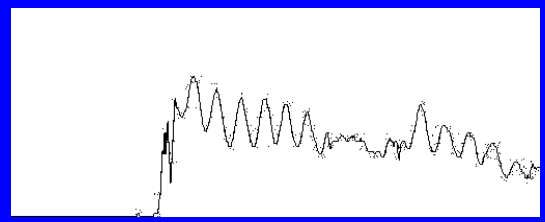
Implementation

- Operation in DAME architecture
  - Part of DAME automatic workflow
  - Developing a tool set

Data representation

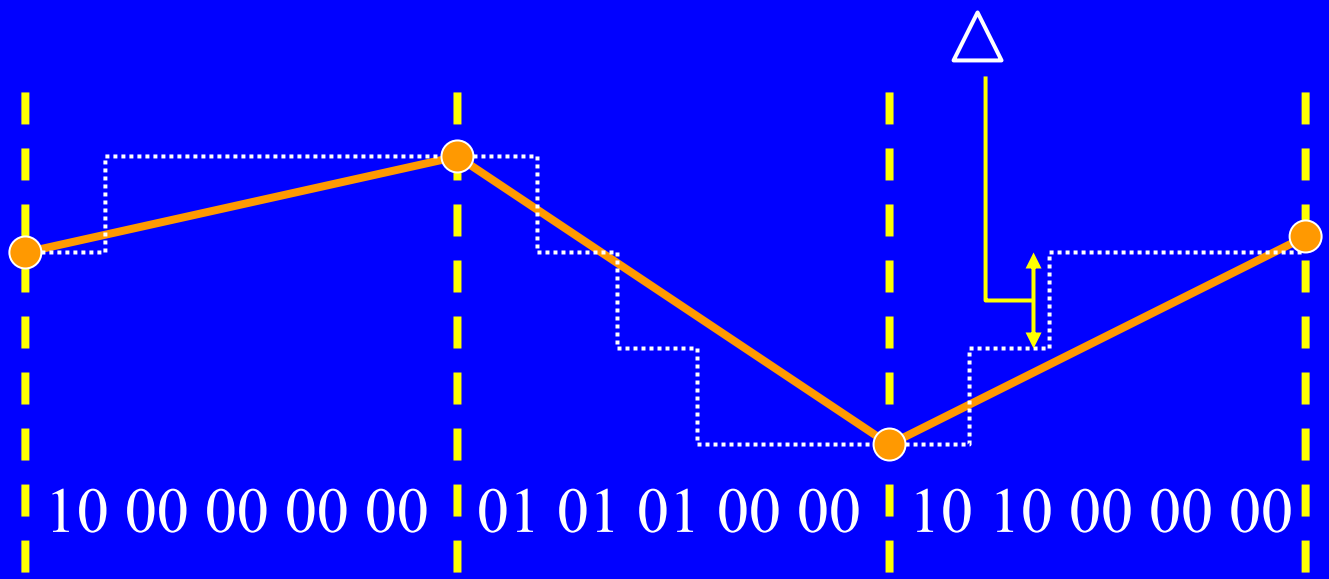
- Extract tracked order – then search for similar examples



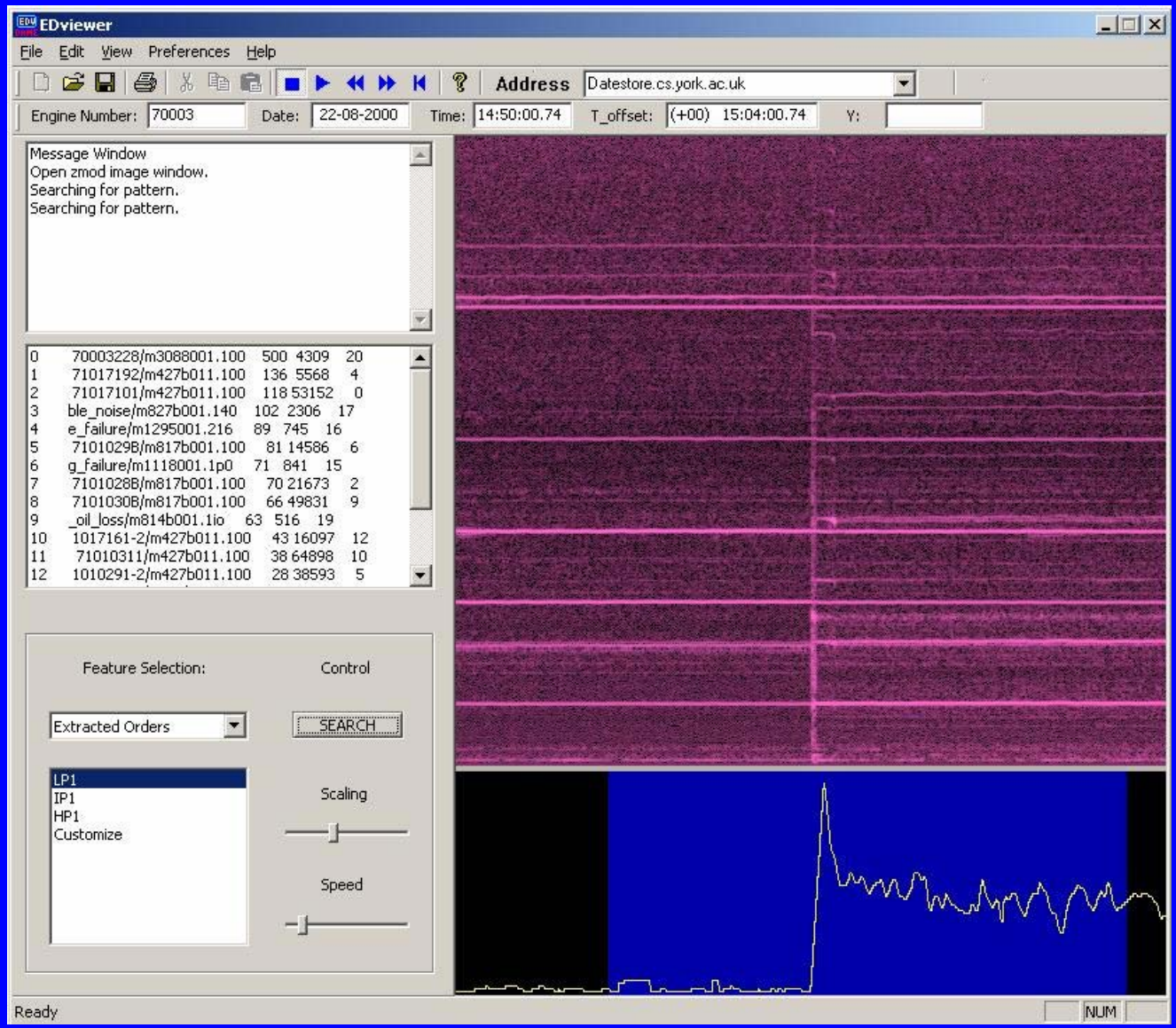


- Must code the data for use in the memories
- Use Delta Modulation coding
  - A variant of Differential Pulse Code Modulation

DM encoding



1000000000010101000010100000000 Bit string to search for



AURA-G time series match tool

Next stages

- Implement next tier of AURA-G
- Perform scaling trials on engine data
- Refine similarity methods