



xTrace - Advanced tracing for z/OS applications

Session # 4

October 29, 2007 10:15 a.m. – 11:15 a.m.

Henrik Meisner
Meisner IT

Today's agenda



- Introduction
- Overview
- Demonstration
- Future



Introduction

- **xTrace developed by Meisner IT**
 - Customers trace nightmare
 - Developed to solve real-life problems
- **IET & Meisner IT Partnership**
 - IET involved with
 - Product testing
 - Product Installation Suite
 - Documentation
 - GUI enhancement (IETeGUI)
 - Marketing
 - Product Support



EDGE *ucate2007*

Overview

- Why xTrace?
- How xTrace works
- xTrace Features
- xTrace Benefits
- Demonstration



EDGE *ucate2007*

Why xTrace ?



- Limited options for tracing z/OS applications (TSO Test, CICS DTF, COBOL DISPLAY statements)
- No support for Batch or IMS
- Windows trace not always adequate substitute for z/OS trace
 - Different runtime behaviour, lack of test data, EABs, etc
- Have to trace through every p-step and action block that is generated for trace
 - Try typing *TRACE OFF* thousands of times!
 - No breakpoint support
- Results in much time spent looking for faults or stepping through code



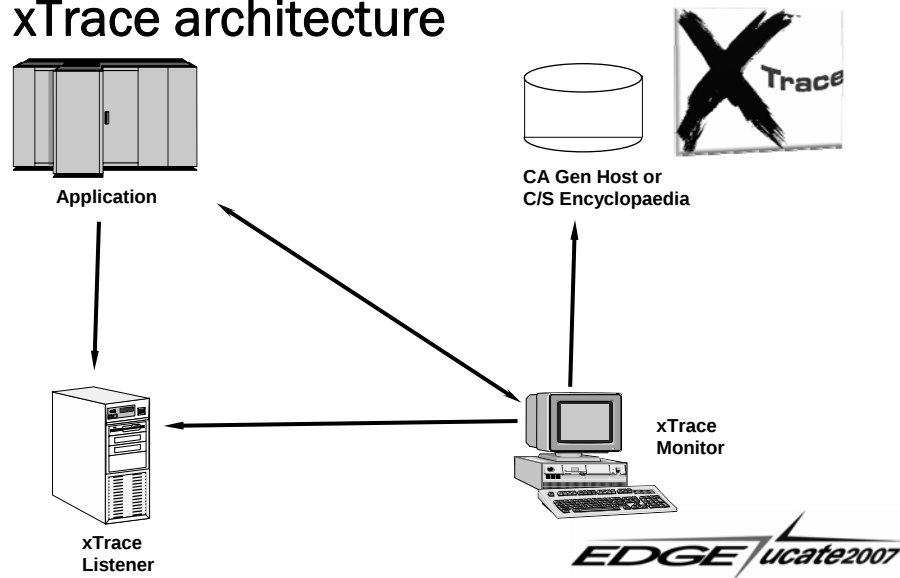
xTrace objectives



- Reduce the developer time from an error report is received till the problem is identified
- Enable tracing for on-line, server and batch code
- Low runtime overhead
 - allows all code to be generated with trace support for development/test environments
- Provide overview of call stack
 - Call Stack provides overview of called modules plus estimate of CPU consumption
- Easy to install and use



xTrace architecture



How xTrace works

- **Generate code with trace support**
 - Standard generated code
- **Uses alternate TIRTRCE runtime routine**
- **TIRTRCE calls XTRACE runtime**
- **XTRACE communicates with xTrace Listener to establish if trace required and IP addresses / ports of Monitor**
- **xTrace Monitor controls tracing and displays action diagram progress**



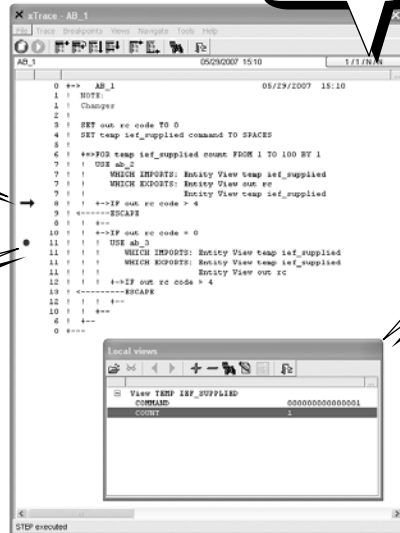
xTrace monitor

Breakpoint definition

Current Statement

Statement Breakpoint

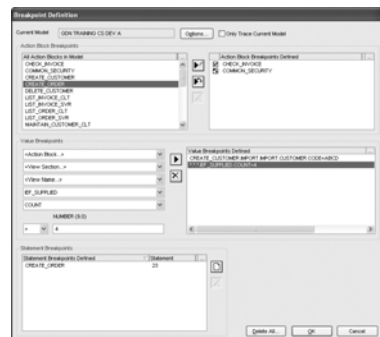
Display & Modify Views



EDGE/ucate2007

Breakpoints

- Enable / Disable tracing by action block
- Value based, including wild cards
 - e.g: *.*.*.work.rc > 4
- Select whether to break on entry/exit
- Break at statement (can be set before starting trace)



EDGE/ucate2007

Callstack

- View complete path through actionblock calling hierarchy
- Collect summary CPU stats



EDGE/ucate2007

xTrace benefits

- ✓ Enables tracing of native Batch & CICS applications
 - Support for IMS planned
- ✓ Custom Breakpoints save developer time when tracing, especially for batch applications
- ✓ Significant productivity improvements when debugging complex applications
- ✓ Low execution overhead allows non-production code to always be generated with trace support allowing trace on demand
- ✓ Performance analysis helps tune application code
- ✓ Easy to install and use



EDGE/ucate2007

Demonstration



EDGE/ucate2007

xTrace –Release 2.2



- genIE Interface
 - Reduce the developer time from an error report is received till the problem is identified **and fixed**
- Breakpoints on commands and exit states
- Infrastructure improvements
 - Get z/OS system information
 - More ways to get application runtime userid
 - Better error handling
 - CP support for 37,273,277,278,284,285,297,500,860,1047
 - Extended ODBC support
- Save Action Diagram print
- Action Block detail
- Source code post processor
 - Get current subscript
 - More reliable CPU consumption figures
 - Capture Gen runtime errors
 - Error handling
 - EAB trace
- Other stuff

EDGE/ucate2007

xTrace – Beyond 2.2...



- Customer feedback will decide
- “Low fruits”
- Support for more target platforms
- Code coverage
- Automatic test
- ???

EDGE / *ucate2007*



Thank you for listening

More information ?

www.iet.co.uk/xtrace

The rest of the conference

EDGE / *ucate2007*

EDGE Needs Your Feedback!

- Please complete the conference evaluations
- Your feedback/suggestions are necessary to continue to bring you top-notch events
- Thank You for Coming – See you in 2008!

