Software AG TechEd 2015 User Conference
April 21-22, 2015

Natural Mainframe 825 and Product Family Update

Patrick Gould
Chief IT Architect
Natural for Mainframes
# Natural for Mainframes High Level Roadmap

## 8.2.5 (April 2015)
- Optimizer Compiler v8.3.2 (ARCH=10)
  - Improvements of arithmetic operations
- NDV and NWO v8.3.2 (IPv6)
  - Additional TRACE levels
- ICS141 with ICU 53.1
  - Dynamic load of datfiles
- PARSE XML Support ARCH=9
  - Performance improvements on IBM

## 8.2.6 (April 2016)
- Usability
  - In-place upgrade from v8.2.5
- Optimizer Compiler v8.3.3 (ARCH=11)
  - Improvements with packed decimal
- ICS121 with ICU 54.1
- Selectable Units
  - Optional New Features
- Roll-Server Enhancements
  - Performance improvements
New and Changed Features of Base Natural
- Customer Feature Requests for Base Natural
- Operations and Performance
- Operating System Interfaces
- Unicode and Code Page Support
- Programming Language
- System Commands
- Editors
- Utilities
- Profile Parameters
- Application Programming Interfaces
New command: SHOWDBS

- Databases defined for the current system (mainframe only)

- Information about DB Options (read, open, acode, wcode)

- Information also available via userexit USR8211N
New and Changed Features of Natural Add-On Products
• Natural CICS Interface
• Customer Feature Requests for Natural Add-On Products
• Natural Complete/SMARTS Interface
• Natural IMS TM Interface
• Natural ISPF
• Natural RPC (Remote Procedure Call)
• Natural Security
Most Recent Natural Add-On Product Versions
• Natural Development Server Version 8.3.2
• Natural Optimizer Compiler Version 8.3.2
• Natural Web I/O Interface Server Version 8.3.2
zIIP Enabler for Natural
Mainframe Optimization with zIIP Enabler for Natural

GPP: General Purpose Processor
Capacity influences SW licensing cost

zIIP: z Integrated Information Processor
Significantly cheaper than GPP
Capacity “free” in terms of SW licensing

Move eligible Natural batch workload from GPP to zIIP

Realize significant GPP CPU savings, often >90% of Natural batch load

Reduce mainframe TCO
How we show the value of NAZ (zIIIP Enabler for Natural)?

- Analyze SMF data to check for Natural Batch CPU usage as compared to total CPU
- On premise or remotely with MXG software (or raw SMF sent to MXG)
- z/OS customers with large batch windows
- z/OS customers with a lot of EntireX Natural ACI/RPC batch servers
- Present findings to customer and determine ROI in Natural Batch CPU reduction
- Propose a trial/POC
  - Create POC document
- Execute the POC and analyze the results
- Final presentation to customer showing expected cost savings based upon POC
Who benefits most from NAZ (zIIP Enabler for Natural)?

- Large z/OS Natural Batch customers who have a ROI with CPU reduction
- Customers who pay IBM and other ISV’s by the CPU second on a monthly basis or regular basis
- Outsourcer who benefits from CPU reduction from IBM or other ISV’s
- Customers who are facing a machine upgrade and would like to defer it
- Customers who need to acquire a new machine and would like to defer or have option to purchase a smaller machine based on potential CPU reduction
- Customers whose Natural/Adabas Batch CPU workload is increasing year to year
Customer References

CALTech

- BOE - Board of Equalization - Just starting to use and POC proved value
- CALSTRS - California Teachers Retirement START system, which runs on Software AG’s Adabas-Natural, manages teachers’ retirement programs so that disability, retirement and survivor benefits can be provided to 868,000 public school educators and their families.
- ROI in 4 months
- Zero coding changes
- Over $685,000.00 saved in 8 months

Reference story on Software AG Web site
Natural Engineer
Perception - Application is Complicated

“an Application is not necessarily complicated, it is just unknown”

• Need to understand the existing Application before embarking on any project - be it migration, modernization or even replace!
Two main areas to consider:

1 - The Code base & Style
   • Status of the code, style, data model

2 - The Culture & Business Strategy
   • Quality of Resources, Tools, Processes
Modernization Project Steps

All modernization projects tend to go thru these 5 iterative steps:

1. **Understand Current Application**
2. **Remove redundant & unused objects**
3. **Componentize Application & ‘tidy up’ Existing**
4. **Migrate platform / language**
5. **Performance Tuning**
Code Styles - Large objects, Bad names....

- Large objects, unstructured code
  - High complexity values (McCabe/Halstead)
- Bad Fields Names
- Difficult to maintain - lack of understanding
- Old Globals or AIVs
  - No control - No implicit definitions
- Generic Globals
  - +PRINTLINE - Redefined many ways based on function
  - Makes Impact Analysis cluttered with “noise”
Natural Engineer - Identifying Object Logic

Diagrammatic representation of Object Logic using JSP notation

- Showing detailed information of large object - Natural & COBOL
Natural Engineer - Identifying ‘Hot-Spots’

Using industry standard metrics - Halstead/McCabe

- Ability to determine most complex objects relative to others within an application
Code Styles - Reusability, Duplication, Testing...

- Logic tightly coupled to function
  - Difficult to reuse
  - No ability to quickly adapt to new Business Needs
    - Lack of IT Agility

- Lack of Governance & Quality Assurance
  - Duplication of code across objects
    - Formatting & Validation Rules
  - Lack of testing
    - No proper data
    - Using Production Data without Data Masking...
Duplication - but not all copies have same logic!
Natural Engineer - Highlights

- Similar Code Identification
- Interactive Literal Searches...
Code Styles - Cluttered Libraries

Large amount of unused objects within libraries
  • Replaced by new objects
  • No tidy up
  • Confuses future analysis & wastes time

  • Unused code within objects
    • “Quick & Dirty” code to remove features
    • Accidentally leaving in unused internal subroutines

  • Tidy up the libraries!!!!
Natural Engineer - Helping to de-clutter…

- Code Improvement - Unused Code
- Quality Logs
- Unused Objects

One customer removed 30% of Production library source code to archive!
Code Styles - “Attack of the Clones”

A copy of an object - or a whole suite of objects

Due to not understanding the context & usage of existing objects

Tight deadlines necessitate “quick wins”

** Note: Never the programmers fault!!
Code Styles - “Bucket Files”

No Control over the Data Model

New fields required are added to the file

The files contain numerous entities within one file
Code Styles - “Hidden Data”

DB fields contain concatenated data - data combined
Maybe due to bad file design & key building
Maybe due to bad file design & “generic tables”

Not easy to find the code accessing specific tables.
• Naming standards may not inform how the table is actually used!
Natural Engineer - Helping locate file/field usage

- Object Viewer with “Used Fields” & XREF
- CRUD Analysis
Natural Engineer - Helping locate file/field usage

Determining via CRUD where & how files & fields used

- Handles DDMs, PUVs & SQL Tables
Code Styles - Mixed Language Applications

• Ability to review mixed language applications in one tool
Natural Engineer - Helping with multi-language interactions

- Integration with Batch Job Scheduler
Natural Engineer - Helping with Multiple Languages / JCL

- Interactive JCL Diagram

- JCL
- COBOL/Natural code
- DDMs/PUVs/DB 2 tables
- Work File layouts
Culture - Where’s the documentation?

Many “mature, heritage” systems lack documentation

If documentation exists, not many know where it is - or use it

No central, indexed repository for developers
Culture - “The GateKeeper”

• Most sites may have “one” person who, allegedly, knows the system
• The “GateKeeper” controls changes made to the system

• Most of the time, they “think” they know how the system works
• Tend to stifle innovation & agility of systems to match Business Needs
Natural Engineer - Helping with Documentation...

• **Object Documentation**
  • Linking external sources of documentation to objects/functions
    – URL
    – Directory
    – Individual Files

• **Keyword Catalogue**
  • Ability to index & search information
Natural Engineer - Helping with Documentation...

- Object Overview
  - Ability to self-document objects
  - Useful when dealing with outsourcing partners & Audit Requirements
Natural Engineer - Helping with Disseminating Information...

- Combination of Functions & Interfaces enables all levels of users - Developers/Systems Analysts/Business Analyst/Business Owners to search & retrieve information on IT resources.
- Object Documentation, Object Overview, Keyword Catalogue
- Zero-footprint Web Interface - NEEGUI
Culture - Mismatch of Understanding - Business & IT
Natural Engineer - Helping with Functional Process

ARIS EPC from Source

• Taking the AS-IS to help design TO-BE