

DumpXDC®

Integrated Dump Analysis Turn cryptic Dumps into Actionable Insights

DumpXDC brings the power of Z/XDC's interactive debugging commands to post-mortem dump analysis. View dumps at source level, format control blocks, and diagnose production failures - all without reproducing the issue. It even works seamlessly with IPCS.

Why You Need DumpXDC

DumpXDC transforms dump analysis from an expert-only, time-consuming task into something any developer can do effectively—using the same intuitive commands they already know from Z/XDC.

Powerful Z/XDC Commands

What It Does: Apply Z/XDC's full command repertoire to dump analysis.

Why it matters: Commands such as FORMAT, LIST, MAP and DMAP s can be leveraged to diagnose historic failures, not just live programs, eliminating the need to recreate the failure.

Object Code Disassembly

What it Does: Display dumped executable code as disassembled instructions, with mnemonics, operands and addresses clearly shown.

Why it Matters: When source isn't available or when investigating compiler generated code, disassembly lets you understand what was executing.

IPCS Integration

What it Does: Seamlessly combine DumpXDC commands with standard IPCS commands in the same session.

Why it Matters: IPCS has powerful capabilities. DumpXDC doesn't replace IPCS, it enhances it by adding Z/XDC's superior formatting and navigation commands.

C/C++ Variable Analysis

What It Does: Display C variables, arrays, structures, and slices with proper typing, formatting, and hierarchy, seeing your C data exactly as you defined it.

Why It Matters: C programs use complex data structures. DumpXDC understands C types and displays them naturally, not as raw hex.

Stop Wrestling With Cryptic Dumps!

Transform production support from days of tedious hex analysis to hours of efficient, source-level investigation. Join support teams who've reduced MTTR by 50-70% with DumpXDC. Try Z/XDC complimentary for 30 days!

Step 1: Production Failure Occurs

System captures dump when program abends or encounters serious error. Dump contains complete program state at failure time.

Step 2: Load Dump in IPCS

Use standard IPCS commands to allocate and select the dump for analysis.

Step 3: Activate DumpXDC

Start DumpXDC within IPCS session. DumpXDC gains access to dump contents while maintaining IPCS integration.

Step 4: Use Z/XDC Commands

Apply familiar Z/XDC commands to analyze the dump:

- LIST PSW to see failure point
- Display source code at that address
- FORMAT control blocks to understand state
- Trace back through call stack
- Examine data structures that might be corrupted

Step 5: Identify Root Cause

Using source-level views and formatted displays, quickly identify:

- What code was executing
- What data was being processed
- What went wrong
- Why it went wrong.

Step 6: Develop and Verify Fix

With clear understanding of root cause, develop fix with confidence. Test thoroughly before deploying to production

Traditional Dump Analysis:

Day 1: Initial hex dump review (4-6 hours)

Day 2: Cross-reference with source (4-6 hours)

Day 3: Attempt reproduction (8 hours)

Day 4: More dump analysis (4 hours)

Day 5: Finally understand issue

Total: 5 days

With DumpXDC:

Hour 1: Load dump and view source at failure point

Hour 2: Format control blocks and understand state

Hour 3: Trace call sequence and identify cause

Hour 4: Verify hypothesis and document findings

Total: 4 hours

Ideal Workflow:

- Develop and test with Z/XDC to catch bugs early
- Ship to production
- If production failure occurs, capture a dump
- Analyze with DumpXDC to diagnose quickly
- Return to Z/XDC to verify fix before redeploying

DumpXDC®

Who should use DumpXDC? Ideal for application developers, SYSPROGS, production support teams and customer support teams!