Delphi Bugs Meet AI

Using Claude Code to debug faster and smarter

Primož Gabrijelčič

thedelphigeek.com

gabr42@gmail.com

gabr42

github.com/gabr42

linkedin.com/in/gabr42

DelphiDay italian conference

9,10 Giugno 2026
Piacenza

wintech italia

OPEN-SOURCE PROJECTS

github.com/gabr42

OmniThreadLibrary

Parallel programming library for VCL/Windows

OmniThreadLibrary-NG

Platform-independent parallel programming library [WIP]

GpDelphiUnits

Various helper units

The Delphi Geek

random ramblings on Delphi, programming, Delphi programming, and all the rest

Tuesday, April 21, 2026

OmniThreadLibrary-NG teaser

Normal suite (default — Stress excluded)

Platform	Found	Ignored	Passed	Failed	Time
Win32	322	0	322	0	40 s
Win64	322	0	322	0	42 s
Linux64	325	3	322	0	32 s
Android	325	3	322	0	38 s
Total			1288	0	~2 min 32 s

Stress only (--include:Stress)

Platform	Found	Ignored	Passed	Failed	Time
Win32	7	0	7	0	293 s
Win64	7	0	7	0	293 s
Linux64	7	0	7	0	275 s
Android	7	0	7	0	284 s
Total			28	0	~19 min 5 s

Debugging with Claude Code

- AI as a moving target
- Alternatives
- Installing & configuring Claude Code
- Talking to AI
- Debugging

Alternatives

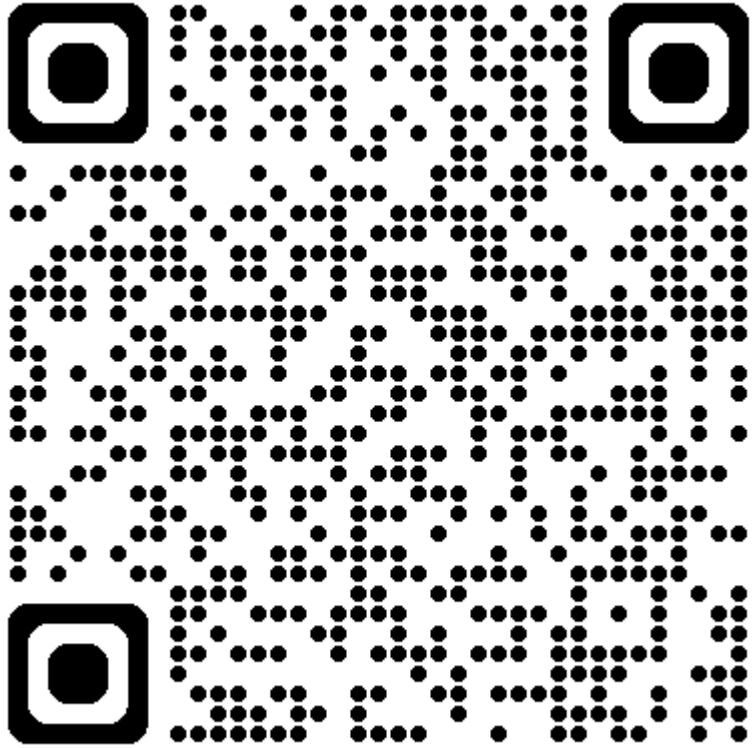
- OpenCode [open source]
- Codex [OpenAI]
- Gemini CLI [Google]
- Github Copilot [Microsoft]
- Kai [Embarcadero]
- Codebot [RemObjects]
- …

Watch out!

- **What happens to submitted code?**
  - Is it used to train models?
  - Will you leak proprietary information?

Running Claude Code locally

- `ollama load qwen3.6:27b-q4_K_M`
  - Runs on 24 GB card
- `ollama launch claude`
  `–model qwen3.6:27b-q4_K_M`
- https://ollama.com/blog/claude



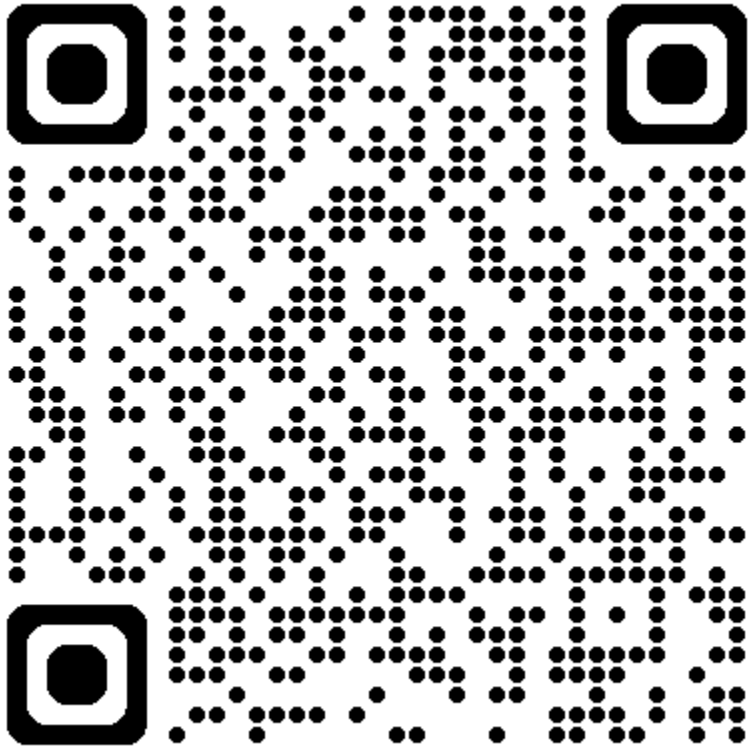
AI as a moving target

- https://isaiprofitable.com
- Prices will change!
- Watch your budget!

Installing Claude Code

Installing Claude Code

- Claude Code
  - https://code.claude.com/docs/en/quickstart
- Claude Desktop
  - https://claude.com/download
- Check online settings!
  - https://Claude.ai, Settings, Privacy

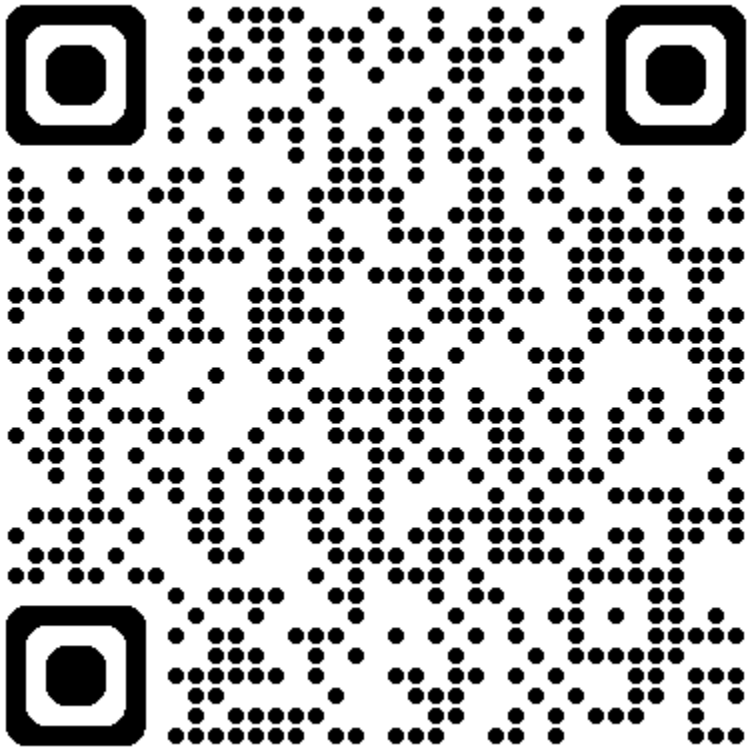


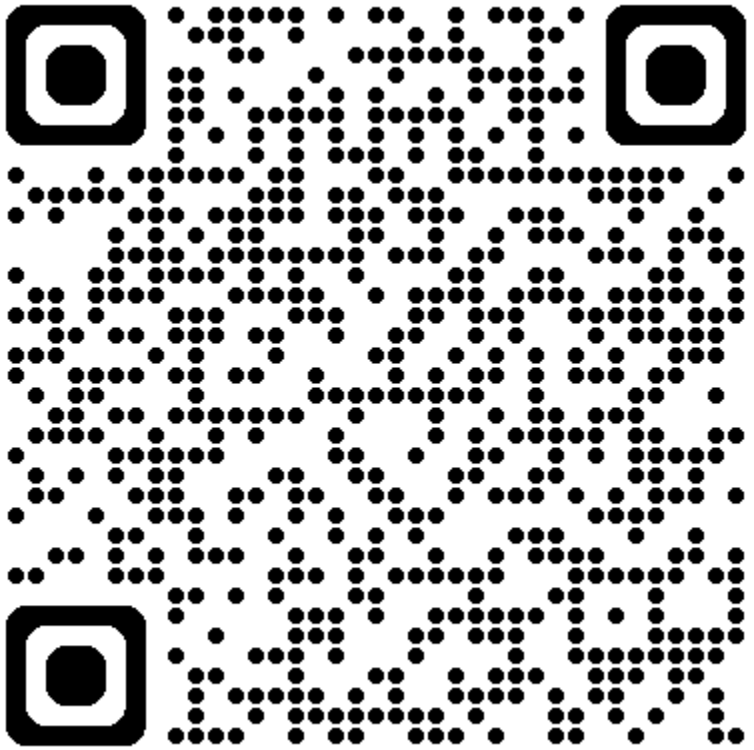
Configuration

- Add-ons
- Good starting CLAUDE.md
- Plugins
- Skills

Add-ons

- **ccstatusline**
  - Status line configuration
  - https://github.com/sirmalloc/ccstatusline
- **claude-swap**
  - Swap between different Claude accounts
  - https://github.com/realiti4/claude-swap





CLAUDE.md

- Tell Claude how do you want it to response (terse, long …)
- What are your coding conventions
- How to do some stuff (compile Delphi units …)
- Any other customizations you need
- Example: https://github.com/gabr42/DelphiAndAI/blob/main/10%20Coding%20GpTimestamp%20with%20Claude%20Code/CLAUDE.md

Plugins

- /plugin
- Claude
  - claude-md-management, skill-creator
- GitHub
  - github, commit-commands, gitkraken-hooks
- Code
  - code-review, code-simplifier, **superpowers**

Skills

- Problem-specific instructions
- Download from internet or add your own
- Text files stored in .claude/skills
- Many examples: https://github.com/anthropics/skills

Talking to AI

- Be specific
  - “When I do this, that happens but this other thing should instead”
  - Rather then “fix my code”
- Ask Claude to generate instructions for itself
  - “I want to do XXX. Prepare instructions for Claude Code to do that.”
- Clear context
  - Restart Claude or /clear
- Complex problems
  - Plan mode, or
  - Make a spec for XXX, then make plan for this spec, then execute plan, or
  - *superpowers*
- Don’t trust the answers!
  - AI will always find a nice story which may not be true

Debugging with Claude Code

Debugging with Claude Code

- Debugging helper
- Stack trace analysis
- Memory leak analysis
- Preemptive debugging
- Writing documentation
- Writing tests

Debugging helper

- Helps understanding code
  - Let Claude describe inner working of your code
    - Create skills that describe your code
    - Skill-creator plugin
- Help finding bugs
  - Describe the problem, let Claude find the reason
  - Always verify!
- Help fixing code
  - Ask Claude to create a fix
  - Then ask Claude to verify that fix
  - Then test the fix

Stack trace analysis

- Send stack trace(s) to Claude, ask it to find a reason for XXX (crash, deadlock)
  - Ctrl+C on Call Stack in Delphi
  - Stack from a logger (MadExcept ...)
- Single thread
- Multiple thread
- Or even multiple trace logs from multiple crashes
  - Ask to find a common reason – if any

Memory leak analysis

- Generate memory leak report with FastMM4
  - FullDebugMode
- Ask Claude to find the reason for the leak

Preemptive debugging

- Check changes before committing
- Static code analysis
  - Simple: “Check this code for potential bugs”
  - Or extensive: “Create a plan for testing units in folder XXX for various kinds of programming problems”
- Add Claude to repository hooks (post-commit analysis)
  - Simple with GitHub
  - More work with internal repositories

Writing tests

- Unit tests
- Stress tests
- …

Writing documentation

- Document public APIs
- Document internal operations
- Document interactions between parts
- …

Examples

Debugging helper: **Analyze code**

- Analyze replication mechanism as it applies to ‘repeating records’
  - repeating_record_analysis.txt

Writing documentation: **Document internal implementation**

- Document ‘repeating records’ behaviour in combination with replication
  - repeating records.md

Debugging helper:
Wide-scope code analysis

- We have UI that behaves differently depending on wheter some conditional is defined. Why?
  - childwin_focus.txt

Debugging helper:
Fix & test cycle

- UI is misbehaving, what could be the reason?
  - subtitler_mdi_analysis.txt

Debugging helper: **Verify bug fixes**

- Find a problem, then check the fix again.
  - gpwinhttp_fix_error_path.txt

Stack trace analysis: **Service doesn’t start**

- Why does it crash?
  - service_exception.txt

Stack trace analysis:
Fix a deadlock

- Find a reason for deadlock
- Fix
- Verify
  - FWASAPICapture_Shutdown_analysis.txt

Stack trace analysis:
Find common problem, do deep analysis

- Analyze a bunch of stack traces
- Find a common reason
- Analyze potential reasons for this problem
- Twist: bug was already fixed (I had no idea)
  - aveco_deadlock.txt

Memory leak analysis:
Find the reason for a MT leak

- Load skills (internal tool unit documentation)
- Find the reason for a leak
- Do an analysis of a fix first
- Fix & test (unit tests)
  - gpeventbus_memory_leak.txt

Preemptive debugging: **Check uncommited changes**

- I was trying to fix XXX. Are my changes OK?
  - preemptive_debugging.txt

Preemptive debugging:
How to analyze large codebase

- Ask Claude: I need a prompt for Claude Code to analyze larger Delphi project for bugs of all kinds - from simple copy&paste bugs to multithreading issues.
  - Audit this Delphi codebase for bugs.txt

Preemptive debugging: **OTL-NG and OTL**

- Create a specification for checking OTL-NG for different kinds of problems
  - STATIC-ANALYSIS-SPEC.md
- Execute this spec
  - STATIC-ANALYSIS-RESULTS.md
- Do a deeper analysis of potential problems to see which are real problems
  - OTL_static_analysis.txt
- Check the list of fixes in OTL-NG and see which apply to OTL-v3
  - STATIC-ANALYSIS-APPLICABLE-v3.md
- Do a deeper analysis … and fix
  - STATIC-ANALYSIS-v3-recap.txt

Preemptive debugging: **Analyze DIOCP**

- Analysis of https://github.com/ymofen/diocp-v5
  - diocp-v5-win64-analysis.txt

Preemptive debugging:
Analyze streaming code in MVC Framework

- Analyze part of https://github.com/danieleteti/delphimvcframework
- Do detailed analysis of top 10 problems
  - serialization_bug_analysis.txt

Writing tests:
GpEventBus stress tests

- Analyze GpEventBus.pas/.md and devise a spec
- Create a plan
- Implement a plan
- Fix a bug exposed by the test
  - gpeventbus_stresstest.txt

Writing tests:
Create unit tests for OTL

- I need more unit tests for OTL. Analyze units and existing unit tests, find out which areas need more unit tests and create a plan for writing these units.
- Now implement this plan.
  - TESTING-PLAN.md