The Joy of Breaking Things

Pat Parseghian

Transmeta

USENIX Technical Conference

June 14, 2002

You make it . . .

. . . we break it!

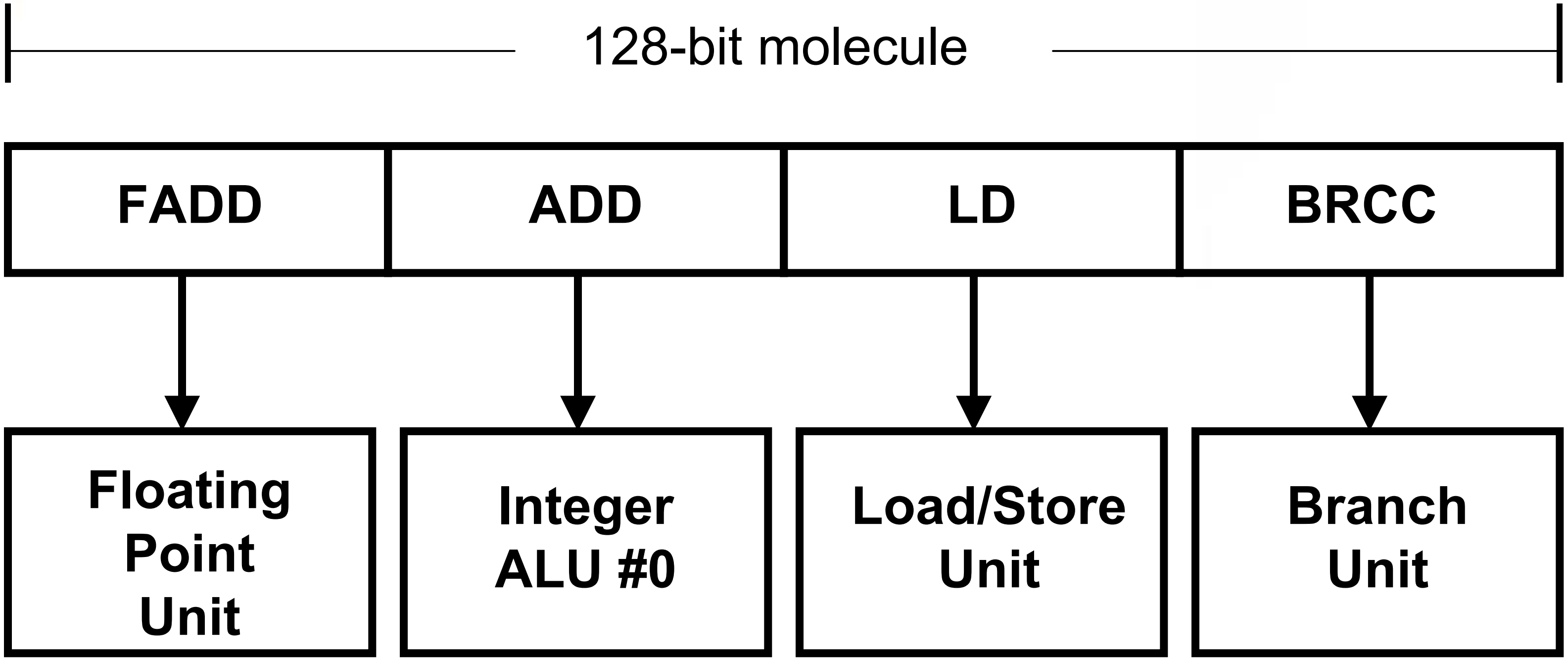
Overview

- How the Crusoe™ microprocessor works
- Testing challenges
- Transmeta Lab for Compatibility (TLC)
- Examples
- Conclusions

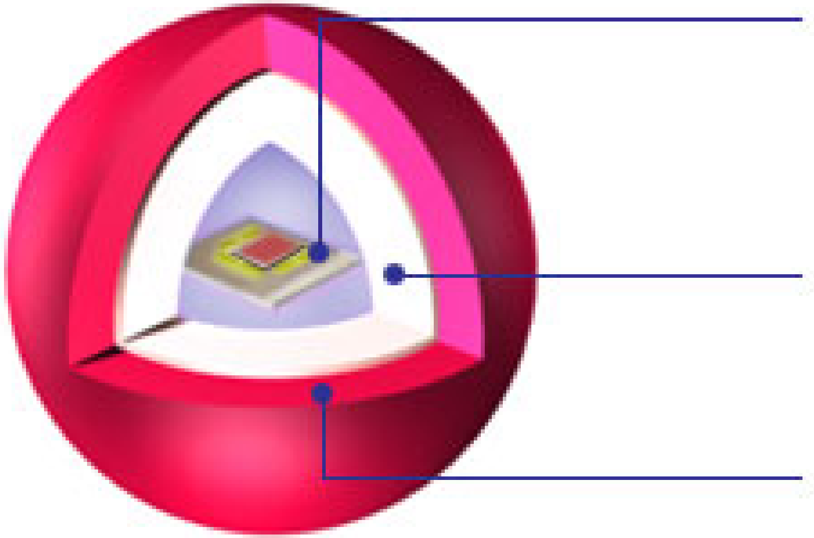
What is Crusoe?

- x86-compatible microprocessor
  - Proprietary VLIW architecture, plus
  - Software layer that translates x86 into VLIW instructions dynamically
- Low power consumption
  - Fewer transistors
  - Adaptive power management techniques
- Upgradeable
  - Fix bugs
  - Improve performance
- Attractive for mobile devices

Very Long Instruction Word Example



Layered Design of Crusoe Processor

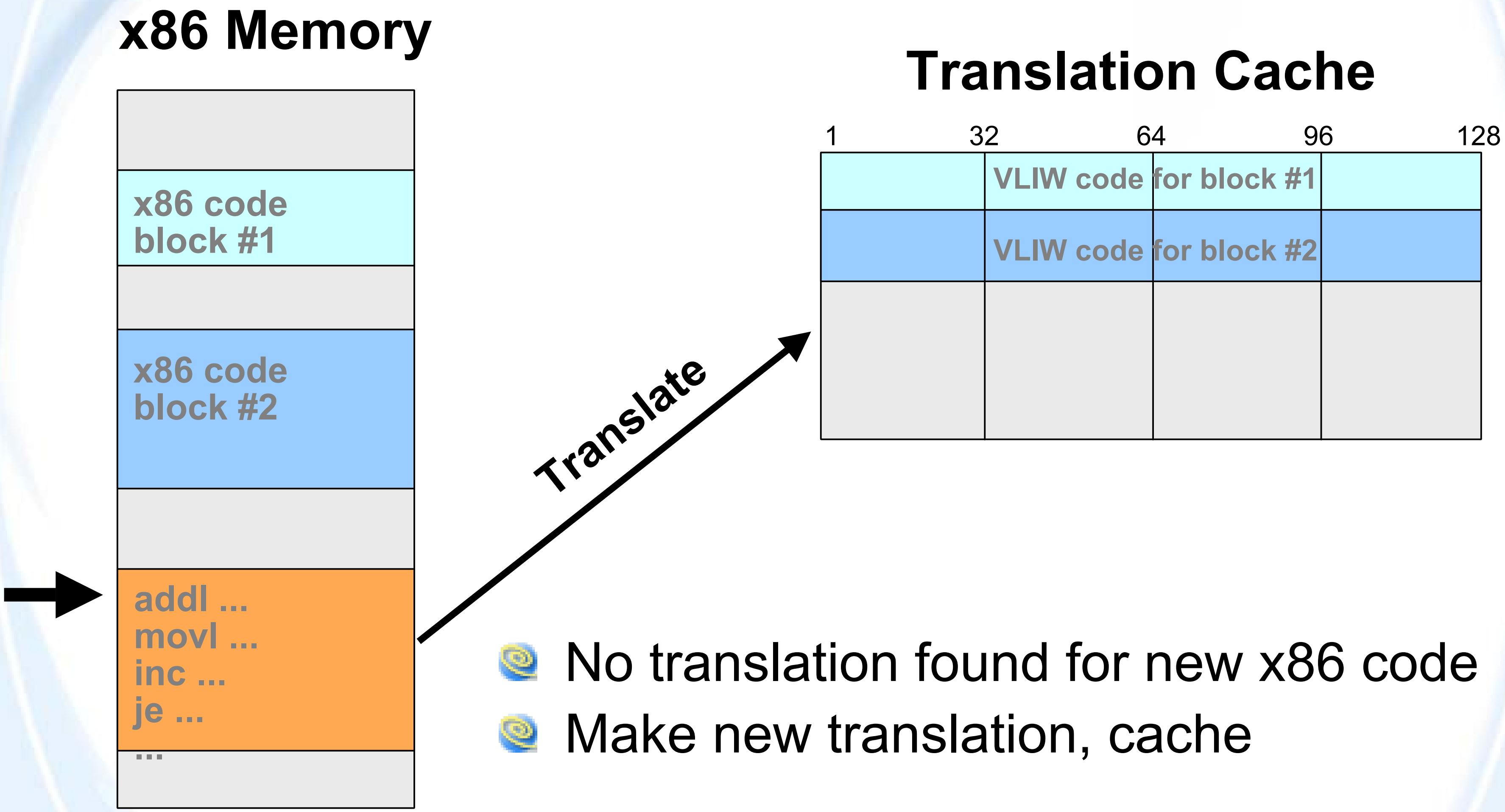


VLIW CPU

Code Morphing™ Software

Operating System, Applications

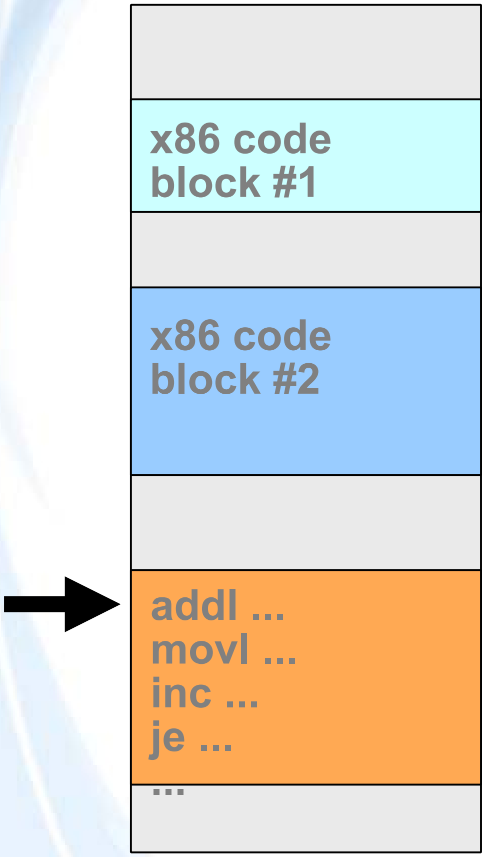
Code Morphing Overview



- No translation found for new x86 code
- Make new translation, cache

Code Morphing Overview (cont.)

x86 Memory

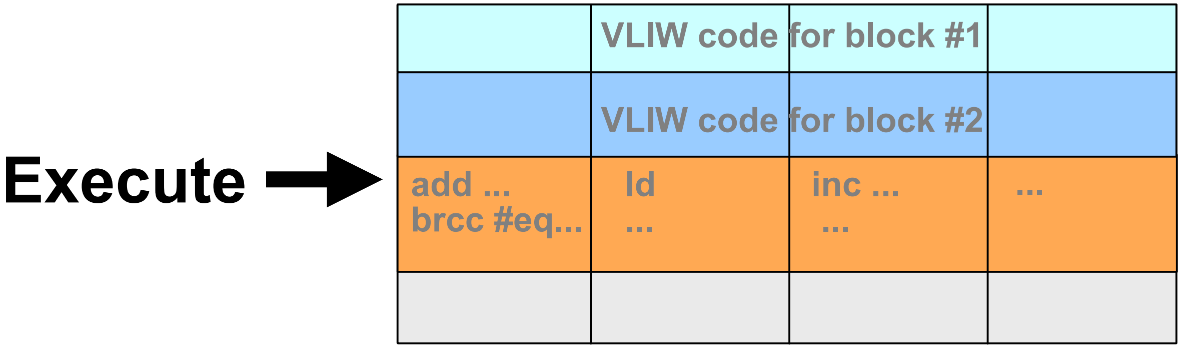


x86 code block #1

x86 code block #2

addl ...
movl ...
inc ...
je ...
...

Translation Cache



Execute

VLIW code for block #1

VLIW code for block #2

add ...
brcc #eq...

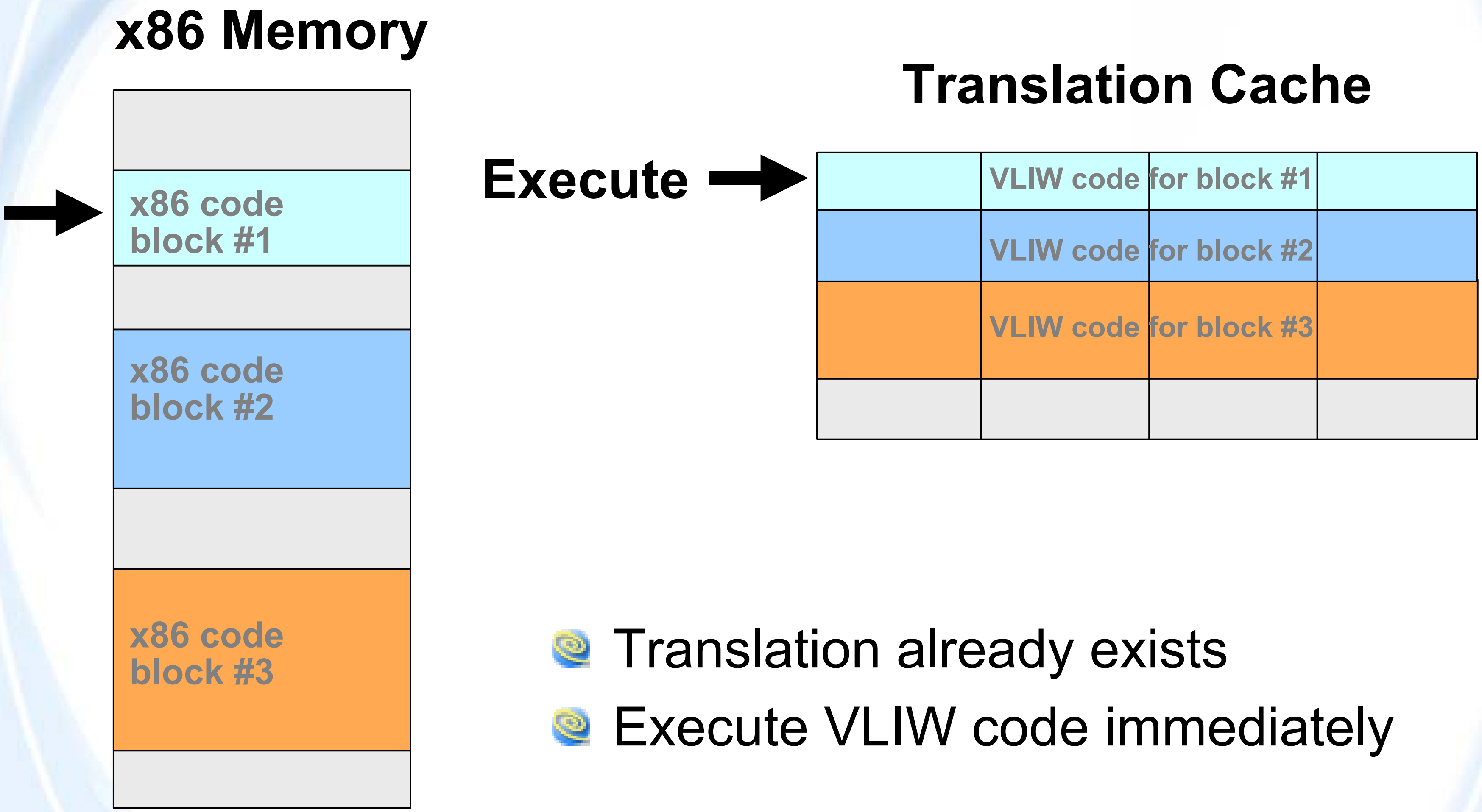
ld
...

inc ...
...

...

- New translation has been made
- Proceed with execution

Code Morphing Overview (cont.)



x86 Memory

x86 code block #1

x86 code block #2

x86 code block #3

Execute

Translation Cache

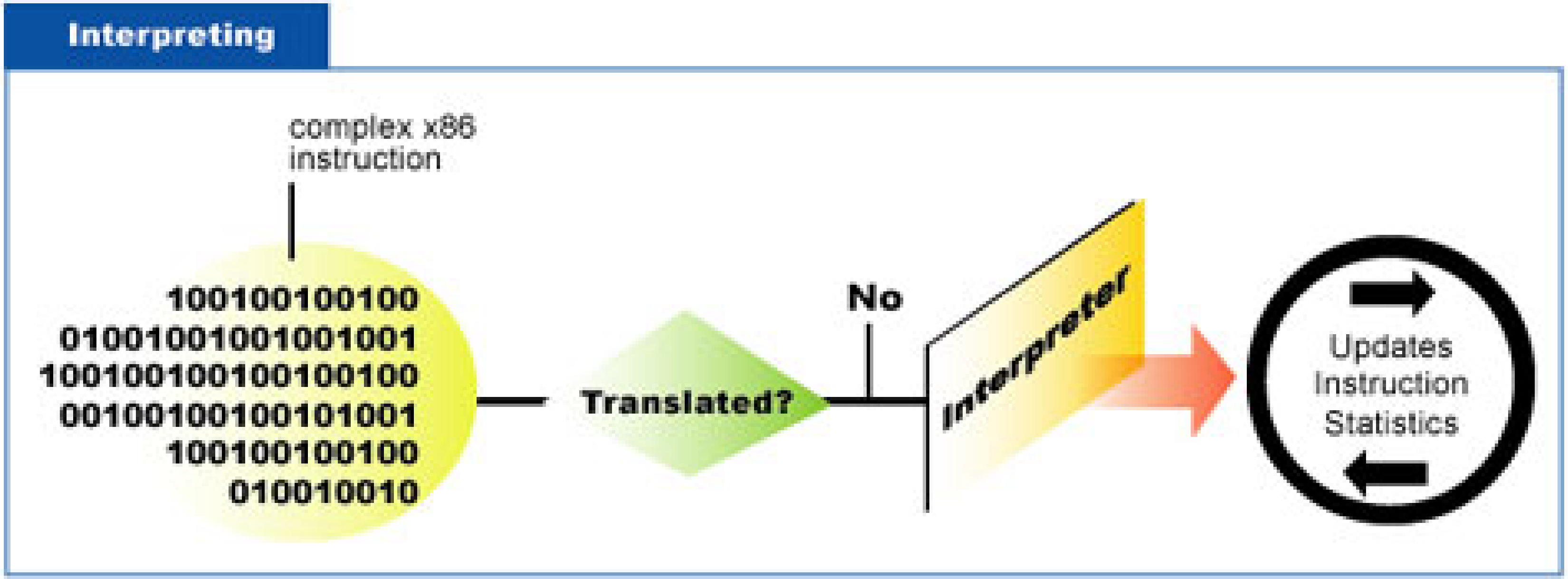
VLIW code for block #1

VLIW code for block #2

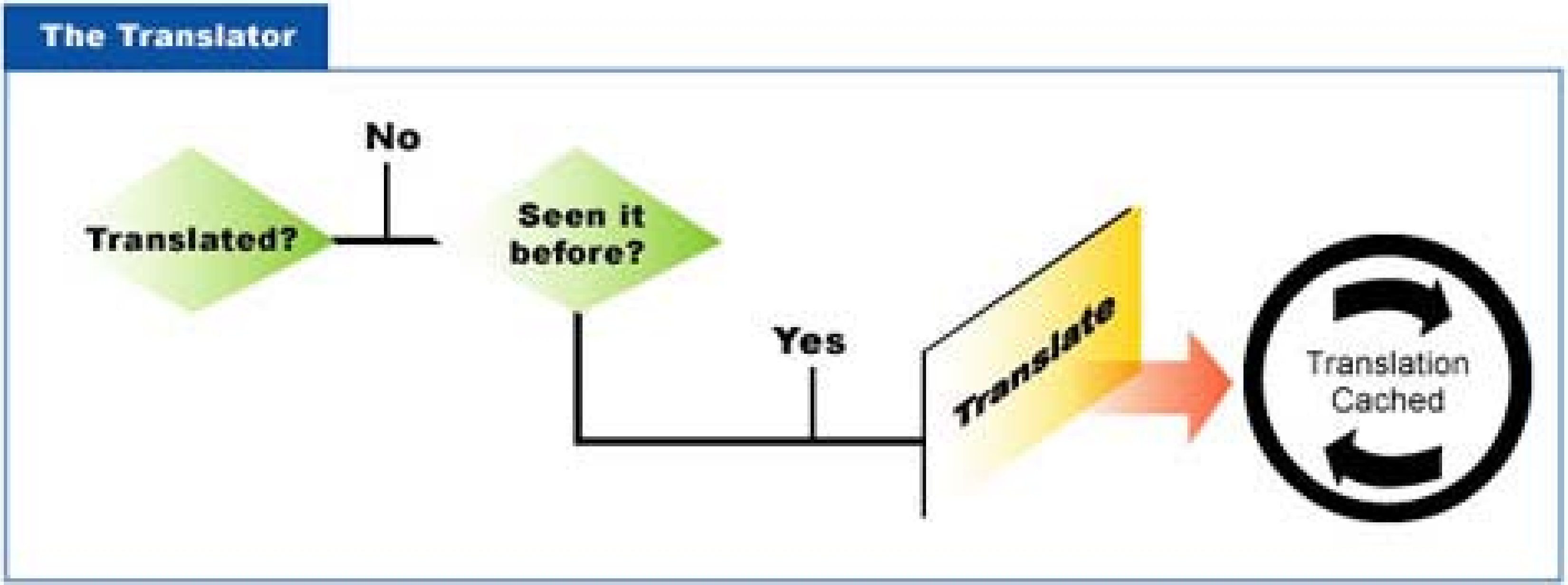
VLIW code for block #3

- Translation already exists
- Execute VLIW code immediately

Code Morphing Interpreter



Code Morphing Translator



Testing Challenges

- Testing two processors
  - x86 (Crusoe)
  - VLIW
- Variability in how Code Morphing Software treats application
  - Retrieves code from translation cache
  - Dynamically translates the code
  - Interprets the code
- VLIW processor may run code at different speeds
  - LongRun™ technology dynamically adjusts processor speed and voltage to match workload

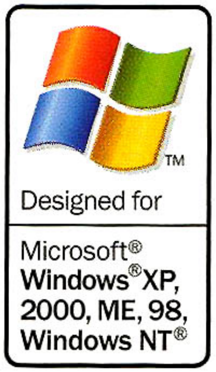
Many Layers of Testing

Before candidate product release enters TLC:

- VLIW Verification
- Code Morphing Software Verification
- x86 Instruction Set Verification
- Stress Testing
- Reliability Testing
- Silicon and Test Platform Screening

Product Certification

- Independent confirmation of functionality and PC compatibility



- Hardware compatibility test suites
  – System-level tests
  – Exercise CPU and its interfaces
  – Stress tests

Hardware Compatibility Tests

- **Microsoft Windows Hardware Quality Labs (WHQL)**
  - Ensure compliance with standards for compatibility with particular Microsoft OS
  - Extensive suite of system-level tests
    - BIOS, power management, functionality
  - Certification required by customers
- **Other HCT Suites: Linux, Solaris, OS/2**
  - Part of standard test repertoire
  - Pursue certification upon customer request

TLC Testing Philosophy

- Imagine you’re an end user, and exercise:
  - Applications bundled by customers
  - Typical applications
  - Popular applications

- Imagine you’re a processor, and exercise:
  - Hardware compatibility test suites
  - Games
  - Legacy software
  - Operating systems

Testing Policies

- Establish criteria to:
  - Start testing
  - Determine scope of testing
  - Continue testing
  - Warrant release of new candidate
  - Release for customer qualification
  - Release to production
- Decide how to regard:
  - Unreproducible errors
  - Problems that disappear

Testing Tips

- Establish methodical approach
- Use consistent approach (across product releases)
- Develop sufficient test matrix
- Build heterogeneity into test plan
- Revise test plan, to stay current and maximize payoff
- Develop regression test plans
- Track resources and results

Value of Tracking

- Problem reports
- Test results
- Logistical
  - Product inventory
  - Product configuration
  - Library (software, devices, tests)
- Analytical payoff:
  - Trend analysis
  - Historical questions

System is hung – it could be . . .

- Damaged silicon
- Manufacturing flaw in Crusoe silicon or package
- Bug or design flaw in Crusoe silicon
- Faulty mainboard, adapter, memory, power supply
- Bug or design flaw in system BIOS
- Bug or design flaw in CMS interpreter or translator
- Bug in operating system or driver
- Bug in application software
- Operator error

Isolation and Troubleshooting

- Repeatable?
  - Same test platform?
  - Different test platform?
  - Non-Crusoe system?
- Identify relevant factors
  - Based on product knowledge
  - Rely on experience
  - Consider probabilities

Debugging, Prior to Product Release

- Silicon Level
  - Logic analyzers, bus analyzers
  - Tools to examine VLIW state
- Code Morphing Level
  - Software techniques
  - Cross debugger
    - Connects from remote machine
    - Executes on VLIW
    - Examines x86 machine state

Example – The Dig

- System hung launching MS-DOS game "The Dig"
- Problem:
  - – CMS caught an inconsistency and halted
- Cause:
  - – Bug in Code Morphing Software
- Solution:
  - – Fixed Code Morphing Software, before release

Example – Microsoft Windows NT

- Windows NT installation aborts with message:
  STOP 0x0000003e MULTIPROCESSOR_CONFIGURATION_ NOT SUPPORTED
- Problem:

  CMPXCHG8B CPUs in Non-Intel/AMD x86 Compatibles Not Supported (Microsoft Knowledge Base Q189988)
- Cause:

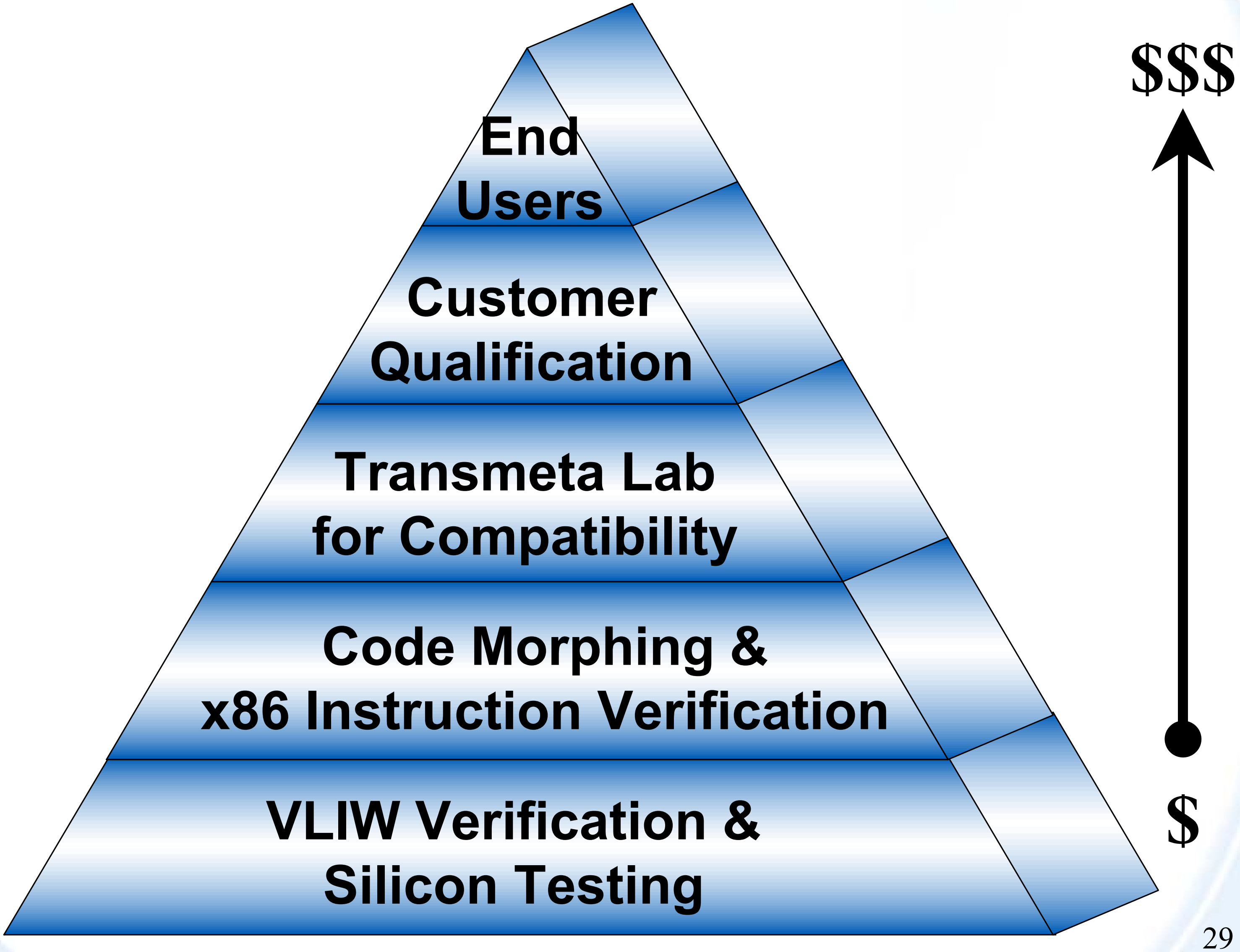
  OS checks vendor name instead of relying on feature flags
- Solution:

  Install Windows NT Service Pack 4 or above

Example – Caldera OpenLinux Server

- OpenLinux Server 3.1.1 installation hangs system
  - Workstation 3.1.1 installs and boots normally
- Problem:
  - Server 3.1.1 requires processors that support Page Address Extensions
- Cause:
  - OS does not check feature flags
- Solution:
  - OS should utilize CPUID to identify supported features

Ferreting Out Failures



End Users

Customer Qualification

Transmeta Lab for Compatibility

Code Morphing & x86 Instruction Verification

VLIW Verification & Silicon Testing

In Conclusion

- Apply our lessons to your products
  - Think about your testing challenges
  - Improve your test process
  - Increase satisfaction of your customers
- Never doubt that breaking things can be fun!