

+ 25

To Err is Human

Robust Error Handling in C++26

SEBASTIAN THEOPHIL



Cppcon
The C++ Conference

20
25



September 13 - 19

ABOUT ME

- Programming C++ Professionally Since 2003
- think-cell employee No 1
- Small team with over 1 Mio customers
- In hostile desktop environment
- Stability and efficiency were key



RELIABILITY VS ERROR HANDLING



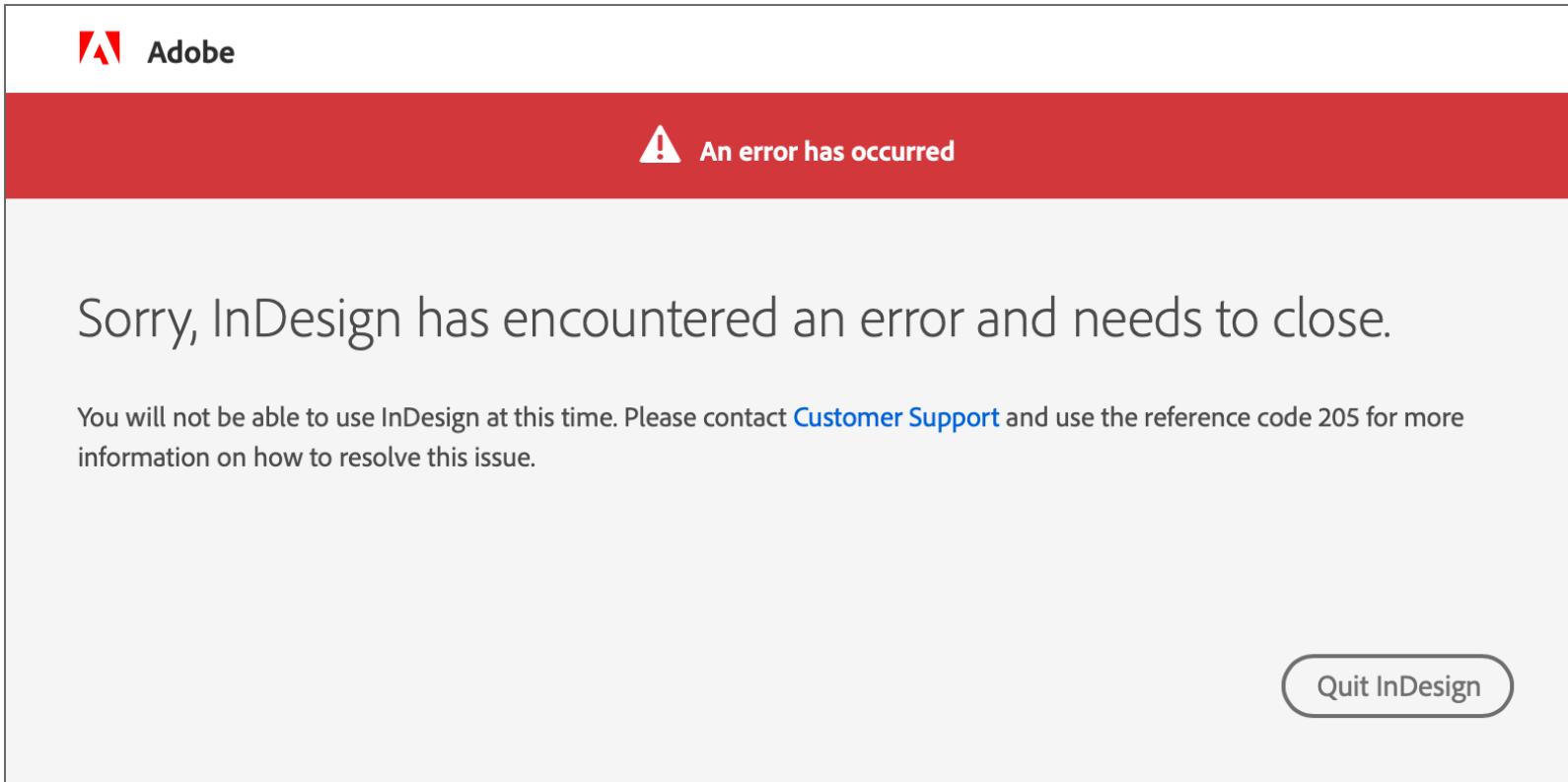
RELIABILITY VS ERROR HANDLING

- Redundant systems (3 + backup)
- In separate locations
- Diversified HW/SW (Intel vs Motorola)
- Must survive RAM faults, engine explosions...

RELIABILITY VS ERROR HANDLING

- Error Handling is "easier"
- We assume your RAM works
- The computer does not explode

RELIABILITY VS ERROR HANDLING



(No Word error message this time!)

WHY BOTHER?

- Environment behaves unexpectedly
- Unforeseen usage scenarios
- Programmer error
- Bad error handling breeds bugs
- Can sink enormous effort into this
- Deliver customer value, not perfect error handlers
- Goal: **stable product, fast!**

GOOD ERROR HANDLING IS THEREFORE KEY!

STRUCTURE OF THIS TALK

1. C-style error handling
2. C++ exceptions
3. `std::expected`
4. Contracts
5. Library Hardening
6. What & how to handle?

C-STYLE ERROR HANDLING

```
int open(const char *path, int oflag, ...);
```

Returns -1 on failure, and sets errno to indicate the error.

- 😊 ENOENT file does not exist
- 🤔 EINTR interrupted by signal
- 🤯 EFAULT path outside address space

C-STYLE ERROR HANDLING

```
HANDLE CreateFileA(  
    [in]          LPCSTR           lpFileName,  
    [in]          DWORD            dwDesiredAccess,  
    [in]          DWORD            dwShareMode,  
    [in, optional] LPSECURITY_ATTRIBUTES lpSecurityAttributes,  
    [in]          DWORD            dwCreationDisposition,  
    [in]          DWORD            dwFlagsAndAttributes,  
    [in, optional] HANDLE           hTemplateFile  
) ;
```

If the function fails, the return value is INVALID_HANDLE_VALUE.

To get extended error information, call GetLastError

C-STYLE ERROR HANDLING

```
std::string str = "10x";
char* end = str.begin();
long v = std::strtol(str.c_str(), /*in/out*/ &end, 10);
```

Success: Returns parsed value

No conversion: Returns 0, sets end, sets errno = EINVAL

Overflow: Clamps value, sets errno = ERANGE

😱 So many things to check! 😱

C-STYLE ERROR HANDLING

Is never going to die.

EXCEPTIONS

- Have a bad reputation
- They are a glorified goto statement
- Difficult to reason about control flow
- Exceptions are horribly slow
- They are banned in our codebase!
- Require writing exception-safe code

EXCEPTIONS

```
1 try {
2     auto json = do_get_request(
3         "https://api.weatherstack.com/current&query=Folkestone"
4     );
5     auto temp = parse_current_temp(json);
6     write_to_log(temp);
7 } catch(std::exception const&) {}
```

- Every line could realistically throw!
- Maybe from a deep stack, e.g., the parser.
- Unwinding may be a feature
- But catch as locally as possible.

EXCEPTIONS

```
1 try {
2     auto json = do_get_request(
3         "https://api.weatherstack.com/current&query=Folkestone"
4     );
5     auto temp = parse_current_temp(json);
6     write_to_log(temp);
7 } catch(decltype(error) const&) {}
8 catch(parser_error const&) {}
9 catch(file_error const&) {}
```

- Semantic error information
- Clean code on the happy path
- Bjarne likes exceptions

EXCEPTIONS

```
1 try {
2     auto json = do_get_request(
3         "https://api.weatherstack.com/current&query=Folkestone"
4     );
5     auto temp = parse_current_temp(json);
6     write_to_log(temp);
7 } catch(decltype(error::http_error) const&) {}
8 catch(decltype(error::parser_error) const&) {}
9 catch(decltype(error::file_error) const&) {}
```

- Slow on the error path
- Exceptions are serial
- Code must be exception-safe

EXCEPTIONS

Exceptionally Bad: The Misuse of Exceptions in C++ & How...



**Exceptionally Bad: The Misuse of Exceptions in C++
& How to Do Better**

Peter Muldoon — CppCon2023

std::expected

C++ and Beyond 2012: Andrei Alexandrescu - Systematic E...



Systematic Error Handling in C++

Andrei Alexandrescu — C++ and Beyond 2012

std::expected

- Conceptional simplicity of return values
- Richness of exceptions
- "*Make exceptions the error code*" (Alexandrescu)

std::expected

```
1 enum class parse_error { out_of_bounds, invalid_input };
2 expected<int, parse_error> parse_int(string_view sv);
```

<https://godbolt.org/z/6joosTzqd>

std::expected

```
1 enum class parse_error { out_of_bounds, invalid_input };
2 expected<int, parse_error> parse_int(string_view sv);
3
4 expected<double, string> parse_and_invert(string_view sv) {
5     auto ex = parse_int(sv);
6     if(ex) {
7         auto n = ex.value() * 2;
8         if(0==n) {
9             return std::unexpected("division by zero");
10        } else {
11            return std::expected<double, std::string>(1.0/n);
12        }
13    } else {
14        switch(ex.error()) {
15            case parse_error::out_of_bounds:
16                return std::unexpected("out of bounds");
17            case parse_error::invalid_input:
18                return std::unexpected("invalid_input");
19        }
20    }
21 }
```

<https://godbolt.org/z/6joosTzqd>

std::expected

```
1 expected<double, string> parse_and_invert(string_view sv) {
2     return parse_int(sv).transform(
3         [](int n) { return n*2; })
4     .transform_error([](parse_error p) -> string {
5         switch(p) {
6             case parse_error::out_of_bounds:
7                 return "out of bounds";
8             case parse_error::invalid_input:
9                 return "invalid input";
10        }
11    }).and_then(
12        [](int n) -> std::expected<double, std::string> {
13            if(0==n) {
14                return std::unexpected("division by zero");
15            } else {
16                return std::expected<double, std::string>(1.0/n);
17            }
18        });
19 }
```

<https://godbolt.org/z/6joosTzqd>

CONTRACTS (C++26)

Cpp Contracts - How Contracts in C++26 Can Improve C++ ...



How Contracts in C++26 Can Improve C++ Code Safety and Correctness

Timur Doumler — CppCon 2024

<https://wg21.link/p2900>

[https://timur.audio/contracts explained in 5 mins](https://timur.audio/contracts_explained_in_5_mins)

CONTRACTS (C++26)

- More flexible, standard assert mechanism
- Not for checking environment behavior
- Express invariants, pre/postconditions
- Behavior customizable at build time

CONTRACTS (C++26)

```
1 int f(const int x)
2   pre(x != 0) // x implicitly const
3   post(r : r != x) // r is const, x must be const
4 {
5   contract_assert(0<=x);
6   return x + 1;
7 }
8
9 int f(const int x); // redecl. without contract is ok
```

CONTRACT EVALUATION

- **pre:** after initialization of arguments, before function body
- **post:** after return value initialization & destructor of local variables
- **exception:** leads to contract violation
- Supported in `constexpr` contexts

CONTRACT SEMANTICS (BUILD TIME)

- **ignore**: do nothing
- **observe**: call contract violation handler
- **enforce**: call contract violation handler and terminate
- **quick-enforce**: terminate

CONTRACT CUSTOMIZATION

```
1 void handle_contractViolation(
2     contracts::contractViolation const& violation
3 ) {
4     if (violation.semantic() ==
5         contracts::evaluation_semantic::observe
6     ) {
7         CallMyErrorHandler(
8             violation.location().function_name(),
9             violation.location().line(),
10            violation.comment(),
11        );
12    }
13    return;
14 } contracts::invoke_default_contract_violation_handler(
15     violation
16 );
17 }
```

CONTRACTS

- Contract assertions may have side effects (e.g. logging), **but**
- No *destructive* side effects that change correctness
- Assertions should be complete and independent

CONTRACTS

```
1 #ifdef BOOST_ENABLE_ASSERT_HANDLER
2 namespace boost {
3 void assertion_failed(char const* expr, char const* fn, char const*
4   file, long line) {
5   MYASSERT(expr)
6 }
7#endif
```

No need for custom hooks once contracts land.

LIBRARY HARDENING (C++26)

- Enforced pre- and post-conditions in the standard library
- Microsoft had debug iterators for a long time
- Based on contracts now
- <http://wg21.link/P3471>

LIBRARY HARDENING (C++26)

- Clang: <https://libcxx.llvm.org/Hardening.html>
- GCC: -fhardened
- MSVC (WIP): <https://github.com/microsoft/STL/wiki/STL-Hardening>

SO MANY OPTIONS ...

SO LITTLE TIME.

WHAT & HOW TO HANDLE?

- We are not paid for handling errors
- but paid for solving customer problems.
- Goal is not great error handling, goal is stable product!
- Bad error handling leads to errors.

WHAT & HOW TO HANDLE?

1. Error handling must be easy
2. Error handling must be tested
3. Error handling must be focused where it matters

1. CHECK EVERYTHING

```
1 char ach[1024];
2 auto dwSize = 1024;
3
4 // Posix:
5 ERRNO( gethostname(ach, dwSize), tc::err::returned(0) );
6
7 // Windows:
8 APIERR( ::GetComputerName(ach, &dwSize) );
```

Check every API call And make it very easy

1. CHECK EVERYTHING

- Check every API call
- One wrapper per error reporting method
(e.g. `errno`, `GetLastError`, `HRESULT`)
- `RETURNS_VOID(x)` to annotate functions returning `void`
- `assert` aggressively, stay in release
- `noexcept` by default
 - May call `std::terminate`, unexpected exceptions will terminate anyway
 - Install `std::set_terminate` handler

2. ASSUME EVERYTHING WORKS

GOAL:

- Keep set of code paths small
- Keep set of program states small

(UNLESS YOU KNOW IT WON'T)

```
1 ERRNORETRYIGNORE(  
2     open("test.txt", O_CREAT | O_EXCL),  
3     tc::err::returned_nongative_value(),  
4     tc::err::returned(invalid_filehandle, EINTR),  
5     tc::err::returned(invalid_filehandle,  
6                         {ENOENT, EACCES, EPERM, ENOSPC}  
7     )  
8 )
```

Only if you can reproduce this!

(UNLESS YOU KNOW IT WON'T)

```
1 APIERIGNORE(
2     RemoveDirectory(szPath),
3     {
4         ERROR_FILE_NOT_FOUND,
5         ERROR_PATH_NOT_FOUND;
6         ERROR_DIR_NOT_EMPTY
7     }
8 )
```

Only if you can reproduce this!

3. WHAT IF CHECKS FAIL (UNEXPECTEDLY)?

PRIO 1: COLLECT AS MUCH INFO AS POSSIBLE

- Client: Send error report with memory dump home
- Server: Halt thread and notify administrator

3. WHAT IF CHECKS FAIL (UNEXPECTEDLY)?

PRIO 2: CARRY ON SOMEHOW

- Behavior now undefined, disable reports
- Do not terminate after asserts, they are code and can be wrong
- This is contracts observe semantics

4. FOCUS

think-cell BugReporter

Search in text fields (case-sensitive)

Unfixed (macOS only) Minimum version (inclusive) Maximum version (inclusive) Occurrences ↴

Matched 191916 occurrences at 1772 locations from 35796 systems and 5739 accounts. Took 571 ms.

Builds fixed	If you are using	Best hypothesis (score)	Entry
	(anonymous namespace)::RunSetup [tcaddin/tcupdate/tcupdate.cpp(362)]	stheophil	tcaddin/tcupdate/tcupdate.cpp(1659) R
32665	User message	Restricted to Gatekeeper installations!	Seen by
	tc::no_adl::com_item_ns::com_item_adl::com_itemImpl(13) [tcaddin/PpAddIn/LayoutAddIn/PowerPointAPI.cpp(79)]	PPCORE.DLL (4169.86)	tcaddin/PpAddIn/LayoutAddIn/PPTCont
Builds fixed	User message	Note	stheophil
	tcaddin/OfficeUtilities/MemoryFind.cpp(404)		

Installing update blocked by Gatekeeper. Go to "System Preferences" -> "Security & Privacy" -> "General" and allow apps downloaded from "App Store and ide

Verifying update failed with code 1: objc[10069]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1019]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1026]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1060]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10718]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10719]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10736]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10818]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1088]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

35679	W	2025-04-26T11:36:09	tcaddin/tcupdate/tcupda
35679	W	2025-04-25T02:54:17	tcaddin/tcupdate/tcupda
35679	W	2025-04-24T05:35:41	tcaddin/tcupdate/tcupda
35679	V	2025-04-23T13:42:41	tcaddin/tcupdate/tcupda
35679	V	2025-04-23T07:07:27	tcaddin/tcupdate/tcupda
35679	V	2025-04-22T13:15:27	tcaddin/tcupdate/tcupda
35679	V	2025-04-21T13:35:26	tcaddin/tcupdate/tcupda
35679	V	2025-04-09T05:41:00	tcaddin/tcupdate/tcupda
35679	V	2025-04-06T06:25:47	tcaddin/tcupdate/tcupda
35679	V	2025-04-04T06:36:11	tcaddin/tcupdate/tcupda

4. FOCUS

think-cell BugReporter

Search in text fields (case-sensitive)

Unfixed (macOS only) Minimum version (inclusive) Maximum version (inclusive) Occurrences

Matched 191916 occurrences at 1772 locations from 35796 systems and 5739 accounts. Took 571 ms.

Builds fixed	If you are using	Best hypothesis (score)	Entry
(anonymous namespace)::RunSetup [tcaddin/tcupdate/tcupdate.cpp(362)]			stheophil
32665	User message	Restricted to Gatekeeper installations!	tcaddin/tcupdate/tcupdate.cpp(1659) R
tc::no_adl::com_item_ns::com_item_adl::com_itemImpl(13)	[tcaddin/PpAddIn/LayoutAddIn/PowerPointAPI.cpp(79)]	PPCORE.DLL (4169.86)	Seen by
Builds fixed	User message	Note	tcaddin/PpAddIn/LayoutAddIn/PPTCont stheophil
tcaddin/OfficeUtilities/MemoryFind.cpp(404)			

Installing update blocked by Gatekeeper. Go to "System Preferences" -> "Security & Privacy" -> "General" and allow apps downloaded from "App Store and ide

Verifying update failed with code 1: objc[10069]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1019]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1026]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1060]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10718]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10719]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10736]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10818]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1088]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

35679	W	2025-04-26T11:36:09	tcaddin/tcupdate/tcupda
35679	W	2025-04-25T02:54:17	tcaddin/tcupdate/tcupda
35679	W	2025-04-24T05:35:41	tcaddin/tcupdate/tcupda
35679	V	2025-04-23T13:42:41	tcaddin/tcupdate/tcupda
35679	V	2025-04-23T07:07:27	tcaddin/tcupdate/tcupda
35679	V	2025-04-22T13:15:27	tcaddin/tcupdate/tcupda
35679	V	2025-04-21T13:35:26	tcaddin/tcupdate/tcupda
35679	V	2025-04-09T05:41:00	tcaddin/tcupdate/tcupda
35679	V	2025-04-06T06:25:47	tcaddin/tcupdate/tcupda
35679	V	2025-04-04T06:36:11	tcaddin/tcupdate/tcupda

4. FOCUS

think-cell BugReporter

Search in text fields (case-sensitive)

Unfixed (macOS only) Minimum version (inclusive) Maximum version (inclusive) Occurrences ↴

Matched 191916 occurrences at 1772 locations from 35796 systems and 5739 accounts. Took 571 ms.

-[OCPpSlideView updateLayer](13) [tcaddin/PpAddIn/LayoutAddIn/PpSlideView.mm(106)]
Builds fixed if you are using Best hypothesis (score) Entry
(anonymous namespace)::RunSetup [tcaddin/tcupdate/tcupdate.cpp(362)]
32665 User message Best hypothesis (score) tcaddin/tcupdate/tcupdate.cpp(1659) R
Seen by
tc::no_aon::com_item_ns::com_item_aon::com_item_impi(13) [tcaddin/PpAddIn/LayoutAddIn/PowerPointAPI.cpp(79)]
Builds fixed User message PPCORE.DLL (4169.86)
Note
tcaddin/OfficeUtilities/MemoryFind.cpp(404)

Installing update blocked by Gatekeeper. Go to "System Preferences" -> "Security & Privacy" -> "General" and allow apps downloaded from "App Store and ide
Verifying update failed with code 1: objc[10069]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[1019]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[1026]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[1060]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[10718]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[10719]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[10736]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[10818]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V
Verifying update failed with code 1: objc[1088]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

ID	File	Date	Category
35679	W...	2025-04-26T11:36:09	tcaddin/tcupdate/tcupda
35679	W...	2025-04-25T02:54:17	tcaddin/tcupdate/tcupda
35679	W...	2025-04-24T05:35:41	tcaddin/tcupdate/tcupda
35679	W...	2025-04-23T13:42:41	tcaddin/tcupdate/tcupda
35679	V...	2025-04-23T07:07:27	tcaddin/tcupdate/tcupda
35679	V...	2025-04-22T13:15:27	tcaddin/tcupdate/tcupda
35679	V...	2025-04-21T13:35:26	tcaddin/tcupdate/tcupda
35679	V...	2025-04-09T05:41:00	tcaddin/tcupdate/tcupda
35679	V...	2025-04-06T06:25:47	tcaddin/tcupdate/tcupda
35679	V...	2025-04-04T06:36:11	tcaddin/tcupdate/tcupda

4. FOCUS

think-cell BugReporter

Search in text fields (case-sensitive)

Unfixed (macOS only) Minimum version (inclusive) Maximum version (inclusive) Occurrences ↴

Matched 191916 occurrences at 1772 locations from 35796 systems and 5739 accounts. Took 571 ms.

Builds fixed	If you are using	Best hypothesis (score)	Entry
(anonymous namespace)::RunSetup [tcaddin/tcupdate/tcupdate.cpp(362)]			stheophil
32665	User message	Best hypothesis (score) Restricted to Gatekeeper installations!	tcaddin/tcupdate/tcupdate.cpp(1659) R Seen by
tc::no_adl::com_item_ns::com_item_adl::com_itemImpl(13) [tcaddin/PpAddIn/LayoutAddIn/PowerPointAPI.cpp(79)]		PPCORE.DLL (4169.86)	tcaddin/PpAddIn/LayoutAddIn/PPTCont stheophil
Builds fixed	User message	Note	
tcaddin/OfficeUtilities/MemoryFind.cpp(404)			

Installing update blocked by Gatekeeper. Go to "System Preferences" -> "Security & Privacy" -> "General" and allow apps downloaded from "App Store and ide

Verifying update failed with code 1: objc[10069]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1019]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1026]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1060]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[10718]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10719]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10736]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10818]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1088]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

ID	File	Date	Entry
35679	W...	2025-04-26T11:36:09	tcaddin/tcupdate/tcupda
35679	W...	2025-04-25T02:54:17	tcaddin/tcupdate/tcupda
35679	W...	2025-04-24T05:35:41	tcaddin/tcupdate/tcupda
35679	W...	2025-04-23T13:42:41	tcaddin/tcupdate/tcupda
35679	V...	2025-04-23T07:07:27	tcaddin/tcupdate/tcupda
35679	V...	2025-04-22T13:15:27	tcaddin/tcupdate/tcupda
35679	V...	2025-04-21T13:35:26	tcaddin/tcupdate/tcupda
35679	V...	2025-04-09T05:41:00	tcaddin/tcupdate/tcupda
35679	V...	2025-04-06T06:25:47	tcaddin/tcupdate/tcupda
35679	V...	2025-04-04T06:36:11	tcaddin/tcupdate/tcupda

5. HOMEWORK

1. Reproduce error at home
2. Add handling only for reproducible errors!
3. That means error handling is testable!

5. HOMEWORK

- 5% of handlers handle 95% of errors
- Write high quality error handlers
- **Bad:** Show message box
- **Good:** Fix the problem

6. CATEGORY: CRITICAL

- nullptr access
- API calls failing unexpectedly
- Assertion failure
- Should never happen, so no handler
- Program becomes invalid

6. CATEGORY: CRITICAL

- Client: Send error report, disable reports
- Server: Notify admin, enter infinite loop
- User message only if false alarm unlikely
- i.e. not for asserts

6. CATEGORY: UNTESTED

```
1 std::optional<char32_t> codepoint_value(auto const& rng);  
2 if (auto och=VERIFYNOTIFY(tc::codepoint_value(rng))) {  
3     return *och;  
4 } else {  
5     return U'\uFFFD'; // REPLACEMENT CHARACTER  
6 }  
7 }
```

- No reproduction, no error handling needed
- Could theoretically happen
- Client: Send error report, throttle reports
- Server: Send error report

6. CATEGORY: BAD EXPERIENCE

- 3rd party bug
- Reproducible, supported and tested
- Degrades user experience
- Users may complain
- Always log, don't report
- Can explain behavior if user calls support

6. CATEGORY: BROKEN ENVIRONMENT

- Weird configuration
- e.g. space configured as decimal separator
- Reproducible, supported and tested
- Could still be cause of problem
- Always logged in debug builds
- Support engineers can set flag to make this show error message in release builds
- Might be reason for support call

7. ERROR REPORTING

- Similar to Google Crashpad, but for all errors
- Out-of-process handling
- Suspend error process
- Create minidump in OS-specific format
- macOS: <https://github.com/think-cell/minidump>
- Upload dump (if customers agree)

7. ERROR REPORTING

- Server-side analysis
- Symbolication and loading in debugger
- Identify last relevant stack entry
- Group errors by line inside last method
- Cause analysis by correlating loaded libraries

7. ERROR REPORTING

think-cell BugReporter

Search in text fields (case-sensitive)

Unfixed (macOS only) Minimum version (inclusive) Maximum version (inclusive) Occurrences ↴

Matched 191916 occurrences at 1772 locations from 35796 systems and 5739 accounts. Took 571 ms.

Builds fixed	If you are using	Best hypothesis (score)	Entry
	(anonymous namespace)::RunSetup [tcaddin/tcupdate/tcupdate.cpp(362)]	13	stheophil
32665	User message	32665	tcaddin/tcupdate/tcupdate.cpp(1659) R
	tc::no_adl::com_item_ns::com_item_adl::com_itemImpl(13) [tcaddin/PpAddIn/LayoutAddIn/PowerPointAPI.cpp(79)]	13	Seen by
		PPCORE.DLL (4169.86)	tcaddin/PpAddIn/LayoutAddIn/PPTCont
Builds fixed	User message	Note	stheophil
	tcaddin/OfficeUtilities/MemoryFind.cpp(404)		

Installing update blocked by Gatekeeper. Go to "System Preferences" -> "Security & Privacy" -> "General" and allow apps downloaded from "App Store and ide

Verifying update failed with code 1: objc[10069]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1019]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1026]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[1060]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/Ve

Verifying update failed with code 1: objc[10718]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10719]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10736]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[10818]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

Verifying update failed with code 1: objc[1088]: Class SPExecutionPolicy is implemented in both /System/Library/PrivateFrameworks/SystemPolicy.framework/V

ID	Timestamp	File Path
35679	2025-04-26T11:36:09	tcaddin/tcupdate/tcupda
35679	2025-04-25T02:54:17	tcaddin/tcupdate/tcupda
35679	2025-04-24T05:35:41	tcaddin/tcupdate/tcupda
35679	2025-04-23T13:42:41	tcaddin/tcupdate/tcupda
35679	2025-04-23T07:07:27	tcaddin/tcupdate/tcupda
35679	2025-04-22T13:15:27	tcaddin/tcupdate/tcupda
35679	2025-04-21T13:35:26	tcaddin/tcupdate/tcupda
35679	2025-04-09T05:41:00	tcaddin/tcupdate/tcupda
35679	2025-04-06T06:25:47	tcaddin/tcupdate/tcupda
35679	2025-04-04T06:36:11	tcaddin/tcupdate/tcupda

THANK YOU

stheophil@think-cell.com

Speaker notes