

The Next Big Thing

Prepared for Meeting C++ 2018

Andrei Alexandrescu, Ph.D.

`andrei@erdani.com`

November 15, 2018

“Squeaky Wheel Gets the Grease”

(“Those were the most cringey minutes of the whole conference!”

— Meeting C++ 2018 Attendee)



2001: “Templates Are Important!”

- 2001: “Modern C++ Design”
- Currently: “modern” is a common noun in “modern C++”

2004: “Threads Are Important!”

comp.std.c++ >

Multithreaded programming: is the C++ standardization committee listening?

249 posts by 37 authors  



Andrei Alexandrescu (See Website for Email)



Years ago, Java decided to add support for multithreading to the core language. Many MT experts said that the support was not very good. Standard C++, as we all know, chose (and as far as I understand, is undecided as of the next standardization iteration) to stay away from MT altogether.

Since then, a predictable sequence of events has happened.

The Java community got better at what they were doing, and the C++ community continued to segregate deeper and deeper among standard-conforming programmers who don't use MT, multi-standard programmers who use, say, pthreads and its C binding (and tread carefully as far as mixing C++ with pthreads go), and hard-core MT programmers who use nonportable documentation provided by their own platform and compiler.

Once it became evident that threads must be part of the language, the pursuit of better language-level abstractions for multithreading naturally continued within the Java research and industrial communities. Also, hard-core C and C++ MT programmers became a more and more isolated clique (occasionally perceived as snooty by the standard-compliant programmers who don't understand the complexities that are at stake).

As a result, much of the research on the recent lock-free data structures

2007: “Online Voting Is Problematic!”

The Wisdom of Crowds

- Anyone can publish anything on the Web
- People select the good from the bad through voting
- Without votes: an amorphous anarchy of good and bad content
- With votes: a hierarchy of quality through popularity

- Issue: not everything popular is good
- Issue: machines and humans cannot easily “cooperate” over the same data

2007: “NLP+Web=Loss of Privacy!”

Natural Language Processing (II)

- Amazon 1.0: “This book is ranked 3.5” (direct integration of user scores)
- Amazon 2.0: “Because of your and others’ shopping history, you may like. . .” (simple data mining)
- Future: “This book’s style may appeal to you, and your blog reveals your keen interest in dystopic sci-fi. Here’s a synopsis. . .”
 - ◆ Issue: resolve the fear that the computer follows every step we make

2007: “Phone+Net+ASR=FUN!”

Speech (I)

- Speech 1.0: Person yelling in front of a huge computer
- Speech 2.0: Person crisply dictating to a desktop computer
- Future: *Dialog with the Web over the cell phone*
 - ◆ Dialog systems are already usable and will be commoditized
 - ◆ Cell phones become more wearable
 - ◆ All buttons in cell phones become unnecessary

2009: “Ranges Are Important!”

- 2009: “Iterators Must Go” Boostcon 2009 keynote
- Currently: std ranges

Today: Disclaimer

Past performance may not be indicative of future results.

Nunc in turpis eu elit dignissim ornare. Vivamus pretium justo a lectus auctor laoreet. Quisque maximus scelerisque cursus. Praesent suscipit accumsan imperdiet. Etiam ultricies nunc ut lorem pellentesque semper. Vivamus nec quam nec mauris aliquam iaculis. Proin eu metus vitae nunc iaculis gravida quis in erat. In rhoncus lectus nec euismod consectetur. Etiam fermentum nulla at vestibulum rutrum. Nullam lorem nulla, posere eget ligula a, faucibus maximus purus. Mauris mollis massa lorem, ad efficitur nisi faucibus accumsan. Mauris consequat, velit vel iaculis ornare, nibh ante rutrum mauris, et auctor leo elit quis turpis. Nam ut velit urna. Ut nec efficitur ante. Integer et quam facilisis, fringilla leo et, feugiat arcu. Phasellus mollis facilisis erat, sit amet ullamcorper mi pulvinar ut. Donec nisi orci, laoreet vel placerat non, pellentesque varius lectus. Maecenas malesuada sapien nisl, et feugiat arcu porttitor. Sed quis dapibus neque, a ultricies lorem. Pellentesque rhoncus leo aliquet odio tempus dictum. Donec nibh metus, pellentesque quis urna ut, porttitor efficitur arcu. Integer vel sem sagittis, dignissim sapien eu, facilisis magna. Phasellus id nulla a nisl vulpate pharetra. Sed aliquet diam non tortor consequat tincidunt. In viverra sapien sed nibh suscipit hendrerit. Ut molestie volutpat ex, dapibus semper dolor euismod eu. Aenean accumsan convallis nulla at feugiat. Nullam in leo commodo erat euismod pellentesque vitae a leo. Curabitur tempus a eros in eleifend. Donec faucibus et ipsum ac congue. Aliquam aliquet dui sit amet turpis mollis rhoncus. Sed faucibus neque at condimentum vulpate. Vestibulum lorem neque, cursus sit amet neque vel, feugiat blandit mi. Aenean ut erat tortor. Phasellus mollis risus non magna aliquam, et lacinia leo rhoncus. Vestibulum in elit a odio condimentum hendrerit ut a est. Mauris eget metus leo. Vestibulum porta maximus luctus. Aenean bibendum eu vel laoreet pretium. Nam vehicula turpis pellentesque vulpate tempus. Praesent ullamcorper est non purus molestie lacinia. Maecenas at dapibus tellus. In a mauris efficitur, ultrices felis non, dictum nulla. Proin ac ultrices massa, eu hendrerit nulla. Nulla venenatis, dolor nec dictum porta, nisi turpis pharetra massa, in laoreet augue mauris non lacus. Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam laoreet ex quis dapibus vehicula. Aliquam libero nulla, fringilla non felis vitae, semper imperdiet dui. Praesent mollis vitae libero non efficitur. Etiam est sem, condimentum tempor condimentum vel, ullamcorper pharetra felis. Fusce tristique semper vulpate. Duis id imperdiet nibh, vitae tincidunt orci. Integer posuere eleifend magna eu vestibulum. Pellentesque magna urna, sodales ut tincidunt vel, facilisis ac neque. Donec aliquam tellus malesuada eleifend consequat. Donec eu volutpat leo, pellentesque placerat orci. In justo ligula, condimentum vel lobortis et, mattis eget neque. Suspendisse tincidunt in lacus nec bibendum. Suspendisse vehicula mi eget scelerisque euismod. Aliquam a interdum nisl. In aliquam, dolor vel auctor ultrices, nunc augue hendrerit nunc, sit amet pellentesque nunc tortor vitae leo. Integer in volutpat nunc. Sed ornare luctus felis quis laoreet. Morbi fermentum, turpis sit amet ornare facilisis, enim mi aliquam ante, non tempus dui massa eget elit. Class aptent taciti sociosq ad litora torquent per conubia nostra, per inceptos himenaeos. Nam nec aliquam ante, quis aliquam ante. Cras odio odio, sodales quis tempor vitae, mattis quis nisl. Aliquam leo lacus, ultrices vel lacinia at, finibus non enim. Suspendisse potenti. Aliquam id sagittis ex. Nulla consectetur dui felis, vel interdum tellus aliquam nec. Proin quis libero sed ipsum elementum laoreet. Quisque porta, urna at tempor sodales, arcu metus maximus nulla, in placerat leo lacus eget nisi. Integer nec nulla mauris. Pellentesque auctor tellus nisi, vitae ultrices est malesuada et. Nunc et arcu lorem. Pellentesque fringilla ligula in diam aliquet ornare. In eros nibh, pharetra at lorem ut, sollicitudin congue ante. Pellentesque interdum tortor luctus neque aliquam blandit. Vivamus quis odio mattis, convallis orci at, pharetra nunc. Maecenas tristique feugiat pulvinar. Suspendisse consequat sagittis quam, nec sodales eros auctor at. Cras vitae fringilla nisi. Phasellus venenatis massa a quam sollicitudin tincidunt. Nulla nisl arcu, egestas nec nulla non, convallis placerat sapien. Etiam in metus vitae nibh ultrices consequat. Donec dolor dui, convallis nec volutpat sit amet, iaculis tincidunt ex. Proin convallis vestibulum est, id gravida ipsum accumsan in. Maecenas egestas tincidunt lorem, vitae tempus ex eleifend quis. Sed est metus, feugiat eget pretium at, molestie in libero. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed tempor nulla et commodo posuere. Maecenas egestas diam at ante lobortis venenatis. Nullam a commodo metus. Nulla placerat arcu et erat pulvinar, eu suscipit lectus elementum. Donec mollis lacus at sagittis dignissim. Phasellus fringilla orci mauris, nec feugiat est faucibus sit amet. Donec ultrices cursus fringilla. Praesent nec libero at quam congue laoreet at sodales dui. Sed vitae ultrices turpis. Mauris efficitur sollicitudin risus eget pretium. Maecenas nec imperdiet tortor, quis viverra dolor. Sed a pretium purus. Integer non augue at felis dictum faucibus. Nunc tellus nunc, interdum eget sollicitudin at, fermentum eu nunc. Nam consectetur nisl elit, in dictum quam facilisis quis. Praesent a ex tellus. Maecenas maximus nunc a lectus pulvinar viverra. Fusce efficitur tempor pellentesque. Donec eu suscipit diam, ac consectetur justo. Aenean rhoncus, est et tempus auctor, odio risus facilisis erat, in dignissim magna velit eu purus. Nunc et luctus enim, a tempus nulla. Vestibulum tempus ante eget rhoncus porttitor. Pellentesque tincidunt justo id consequat pellentesque. Morbi ultrices euismod ipsum, vel consectetur orci finibus ac. Nam in velit ullamcorper, mattis diam eu, lobortis nunc. Nam lobortis neque nibh, id porttitor enim sollicitudin non. Donec mauris tortor, tempus ut velit maximus, lacinia mattis urna. Aliquam erat volutpat. Ut turpis risus, ornare a porttitor vel, euismod quis odio. Fusce in eleifend mi. Donec ex pharetra quam, non suscipit velit. Vivamus sit amet pellentesque mauris, fermentum placerat diam. Quisque pellentesque volutpat ex, non porta quam pharetra ut. Integer enim magna, volutpat et pellentesque in, finibus in urna. Quisque vel ipsum varius, ullamcorper velit vel, dapibus diam. Phasellus at elit purus. Aenean consectetur eros magna, in aliquet tortor volutpat sed. Pellentesque sagittis lacus sed feugiat fermentum. Integer a tincidunt mi, eu pharetra lorem. Nunc ornare quis diam luctus laoreet. Duis bibendum diam ac euismod euismod. Nullam rhoncus consectetur nunc, ac aliquam purus blandit non. Phasellus laoreet rutrum odio, a sodales lacus placerat nec. Maecenas sem urna, luctus vitae erat sed, feugiat pellentesque lectus. Duis ut purus nisl. Integer auctor efficitur libero, nec tincidunt sapien tempor in. Proin commodo nulla velit, ac eleifend ex viverra sed. Praesent quis nisl dapibus, finibus felis eget, tempus sem. Nunc laoreet dapibus orci, id cursus elit. Nunc vitae ex suscipit arcu porta dictum nec at sapien. Curabitur sagittis sed elit ac viverra. Fusce consequat rutrum nisl, sit amet sagittis quam ultrices quis. Nunc porttitor pellentesque magna eget tristique. Aliquam varius at dui vel placerat. Donec sit amet leo luctus, lobortis dolor ut, varius eros. Nulla dolor ante, rhoncus et neque vitae, mattis vulpate nibh. Etiam pretium augue scelerisque mauris blandit, ut lobortis nunc consectetur. Vestibulum consectetur blandit eros sit amet scelerisque. Nullam odio arcu, rhoncus volutpat ante vitae, blandit tristique augue. In hac habitasse platea dictumst. Nam et tellus nisl. Fusce rhoncus ullamcorper turpis, efficitur porta tellus tristique id. In ac mauris molestie, convallis ipsum id, suscipit orci. Vivamus cursus condimentum ligula, at mollis dolor aliquet a. Duis eget tincidunt orci. Nam pharetra, velit vel consectetur dignissim, massa lorem scelerisque diam, sed elementum libero metus id nulla. Fusce id mattis felis. Etiam nisi enim, facilisis at lobortis a, vehicula sed risus. Aenean tristique urna in felis tincidunt, in ultrices ex vehicula. Praesent dictum magna semper, ullamcorper ipsum id, mollis ex. Pellentesque vitae tempus dolor. Sed id nunc vulpate, scelerisque risus congue, lacinia leo. Mauris eu varius justo. Ut congue interdum pharetra. Praesent diam lorem, venenatis vulpate luctus in, ornare sed justo. Maecenas consectetur arcu magna, et porta nunc mattis sed. Donec ac quam eu elit sagittis posuere a eu nisia. Fusce consectetur purus lorem, eget euismod tellus accumsan eget. Ut condimentum arcu eu mauris ullamcorper, rutrum rhoncus sapien viverra. Sed finibus, justo eu egestas viverra, ex ex sagittis lectus, non fermentum lectus enim ac leo. Vestibulum at augue hendrerit, mattis lorem ac, mollis purus. Donec pharetra ac augue vel vulpate. Donec vel lacinia elit. Donec sodales sollicitudin lorem a facilisis. Duis dolor felis, lacinia nec accumsan quis, molestie mollis ante. Nunc faucibus lorem augue, id aliquet arcu condimentum vel. Nulla sollicitudin eget libero pretium vehicula. Pellentesque quis ligula maximus, tincidunt lorem id, finibus urna. Proin semper, purus quis pretium vestibulum, erat elit posuere purus, at aliquet purus est vitae ligula. Duis maximus sem ut velit auctor, et placerat quam auctor. Ut pharetra purus eget dui volutpat sagittis. Integer eu ex auctor, pretium metus sed, egestas orci. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean mattis, sem eu dictum sodales, ipsum lorem molestie lorem, a porta justo odio eu ipsum. Suspendisse suscipit odio nec metus pellentesque auctor. Aliquam at vehicula lectus. Curabitur purus est, efficitur non molestie ut, semper nec risus. Cras accumsan, dui nec rutrum facilisis, enim sapien rutrum ante, vitae tempus mauris velit non libero. Fusce efficitur lacus a purus scelerisque semper. Maecenas ultrices fermentum sollicitudin. Pellentesque egestas dui et lacinia malesuada. Quisque eu leo eget ipsum interdum dignissim. Vestibulum accumsan lorem eu ex molestie, a vehicula turpis lacinia. Cras et ligula quis magna vestibulum rhoncus a sit amet quam. Vestibulum ut ultrices felis. Donec ac nunc ac justo gravida elementum. Vivamus in lacus erat. Suspendisse tincidunt nisi scelerisque lorem ornare mattis. Ut tempus porttitor odio, quis volutpat ex tincidunt ac. Duis vestibulum egestas diam, eget faucibus libero sodales vel. Proin semper malesuada ligula, sed pharetra elit mattis at. Mauris vitae metus sem. Praesent in justo a nisi convallis luctus at ut nisi. Aliquam in diam efficitur, pharetra ipsum quis, ultrices lorem. Sed sollicitudin varius orci, aliquam varius est dapibus et. Vivamus congue, sem non sollicitudin facilisis, leo felis dignissim elit, id placerat mi dolor vel libero. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Pellentesque gravida imperdiet urna, in euismod lacus viverra eu. Quisque elementum tristique purus, nec blandit est accumsan lobortis. Integer sagittis metus nec risus facilisis consequat. Vivamus ullamcorper ante id diam efficitur dictum. Nunc iaculis nisi ac gravida porta. Fusce dictum cursus ante non lobortis. Pellentesque eget aliquet lectus, ut condimentum sapien. Vivamus ut tristique arcu. Proin tincidunt interdum lacus in pharetra. Donec vestibulum lorem mi, quis molestie ipsum aliquet sed. Donec ornare tristique auctor. Nullam euismod augue sed ante feugiat egestas. Nam risus metus, faucibus vitae purus sit amet, efficitur blandit augue. Etiam ac lorem sed libero scelerisque congue quis non massa. In at aliquet neque. Proin consequat, velit ut ullamcorper posuere, mi erat sodales lacus, vitae tempus dui eros ac sem. Ut nibh tortor, facilisis vitae euismod et, aliquet id nisl. Nullam in velit eu leo lobortis scelerisque ut vitae nibh. Maecenas mattis est vel tincidunt aliquam. Pellentesque tempor aliquam malesuada. Maecenas porttitor tortor eu nunc finibus semper. Maecenas fermentum eget velit nec bibendum. Praesent blandit sit amet leo ac condimentum. Nunc eget sem feugiat, interdum ante id, tempus ante. Nunc rutrum nisl eget luctus ornare. Nunc gravida, nisi non ultrices interdum, justo ligula gravida massa, tincidunt dictum nisl dolor ut nulla. Praesent blandit velit id erat ornare condimentum ac at risus. Nulla facilisi. Praesent ut dignissim velit. Maecenas vitae turpis nec leo viverra dignissim elementum ut diam. Etiam ac ullamcorper velit. Class aptent taciti sociosq ad litora torquent per conubia nostra, per inceptos himenaeos. Vestibulum mattis faucibus nisl, ac congue nisl euismod vitae. In bibendum libero mollis malesuada congue. Donec placerat commodo euismod. Aenean tempor erat libero, non tempus mi facilisis eu. Mauris nec malesuada neque. Ut vulpate elementum diam, quis hendrerit diam gravida vel. Ut elit leo, lacinia eget sodales consequat, lacinia sit amet mauris. Pellentesque sed est id aliquet. Sed porta viverra arcu, ut sagittis metus tincidunt rhoncus. Nunc ac mattis felis. Maecenas scelerisque ex magna, sit amet commodo erat molestie sit amet. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec interdum convallis ligula, et cursus diam vulpate vitae. Praesent vulpate a elit quis vulpate. Donec sapien odio, vulpate et metus diam, interdum egestas tellus. Praesent convallis nisi non iaculis gravida. Mauris eros diam, rhoncus nec luctus in, ullamcorper vitae nunc. Suspendisse interdum tempor urna, vitae hendrerit erat commodo at. Aliquam erat volutpat. Morbi tristique eros vel molestie malesuada. Nunc sed magna dui. Nam porta scelerisque nisi, in finibus sapien consequat nec. Duis efficitur ex enim, non maximus orci consectetur quis. Sed vehicula magna nec justo dictum, nec dignissim nisi pharetra. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Sed tincidunt mauris at ni tincidunt, at fringilla nunc dignissim. Etiam eu ultricies augue. Morbi sapien velit, vulpate at gravida vitae, faucibus semper tellus. Phasellus dignissim vitae risus et euismod. Nullam tristique felis quis pulvinar suscipit. Proin dignissim, turpis vitae feugiat finibus, ligula erat dapibus nisl, sit amet consectetur ipsum ligula ac sapien. Praesent facilisis arcu ac elit accumsan molestie. Suspendisse hendrerit venenatis mollis. Morbi vulpate, dui vel hendrerit gravida, tellus nisl posuere massa, vel tempus elit nulla a enim. Aliquam convallis sit amet est sed mattis. Donec at justo aliquam, viverra leo luctus, interdum quam. Phasellus non tempus nisl, at congue magna. Fusce ut mi eu arcu pharetra ullamcorper nec ut urna. Proin turpis, erat rutrum in massa sed, pulvinar ornare leo. Sed varius, felis in fermentum malesuada, dui libero iaculis tortor, eu vestibulum lorem felis id eros. Nunc bibendum leo eu urna malesuada sagittis. Vivamus commodo a nibh non maximus. Curabitur ac leo nec quam efficitur volutpat et ut arcu. Aliquam erat volutpat. Praesent vel nulla auctor, fringilla libero tempus, eleifend nulla. In tempor purus ante, non malesuada tortor lacinia et. Nullam rutrum et lectus in aliquet. Duis at ante vitae sollicitudin luctus. Phasellus tincidunt eros et nulla dignissim aliquam. Curabitur nec pellentesque ipsum, vel dignissim leo. Sed faucibus, libero quis elementum imperdiet, urna elit faucibus neque, id rhoncus dolor nulla vel dui. Vivamus ac porta mi. Maecenas in nisl tempus, ornare ligula sit amet, gravida quam. Aliquam quis tempus augue. Maecenas ac luctus risus. Suspendisse id tincidunt metus. Morbi efficitur mauris consectetur, elementum velit eget, condimentum libero. Maecenas rhoncus tempus consequat laoreet. Fusce lorem tortor, ultricies at ultrices nec, fringilla sed enim. Etiam rhoncus lacus eget ipsum semper fringilla. Mauris porttitor metus eget vehicula consequat. Sed venenatis ligula vel velit imperdiet, vitae facilisis ante tempor. Donec pharetra orci ut porttitor tempus. Nam sed finibus diam. Morbi efficitur molestie elit eu volutpat. Cras condimentum lacus quis felis imperdiet porttitor. Aliquam et dolor sem. Suspendisse ut lorem a orci gravida viverra. Sed faucibus nisi et nec, convallis lorem viverra sed. Quisque facilisis sapien at ullamcorper elementum. Nullam pellentesque consectetur risus et feugiat. Pellentesque vitae varius lectus, non vestibulum ligula. Aenean commodo lacus nec libero tincidunt posuere. Donec imperdiet viverra quam, ac dictum dui ornare nec. Curabitur sollicitudin at erat non aliquam. Praesent ac neque nulla. Quisque at lacus iaculis, dapibus nunc ultrices, dapibus risus. Ut et tempor ex. Phasellus ut nisl eget nulla vehicula molestie. Cras diam orci, tempus in nibh a, bibendum convallis ex. Integer commodo justo id orci consequat aliquet. Fusce egestas lorem in nunc vulpate condimentum. Nulla interdum ex et risus tristique pretium vitae sed magna. Sed ut justo ut purus auctor egestas in in diam. Nulla pretium dictum consectetur. Cras quis bibendum lectus. Pellentesque tristique, purus vitae cursus porttitor, velit nunc elementum lacus, lobortis consectetur elit neque eget metus. Pellentesque ac enim rutrum, interdum magna sit amet, blandit velit. Ut nec fringilla felis. Sed vulpate ex a euismod consectetur. In hac habitasse platea dictumst.

You Know What's Big?

Code. That's What's Big.

- Project size growing
- “All simple programs have been written”
- How's the bug rate faring with code size?

Bug Rate

- 15–50 errors per 1000 lines of delivered code
 - Largely independent of language
 - Measured across a large class of projects
-
- (Source: Steve McConnell, *Code Complete*)

“The worst thing that can happen
to a code base is size.”

— Steve Yegge

How do we fight that?

- Concepts
- Metaclasses
- Introspection

How do we fight that?

- Concepts: add constraints
- Metaclasses
- Introspection

How do we fight that?

- Concepts: add constraints (bo-riing!)
- Metaclasses
- Introspection

How do we fight that?

- Concepts: add constraints (bo-riing!)
- Metaclasses: automate boilerplate
- Introspection

How do we fight that?

- Concepts: add constraints (bo-riing!)
- Metaclasses: automate boilerplate (bo-riing!)
- Introspection

How do we fight that?

- Concepts: add constraints (bo-riing!)
- Metaclasses: automate boilerplate (bo-riing!)
- Introspection: unbridled creativity

How do we fight that?

- Concepts: add constraints (bo-riing!)
- Metaclasses: automate boilerplate (bo-riing!)
- Introspection: unbridled creativity (WOW!)

We want to generate more *smart*
code, not more boilerplate!

Design by Introspection

Recap: Policy-Based Design

- Semi-Automatic use of Design Patterns
- Coined by “Modern C++ Design” in 2001
- Enjoys use in C++, D
- Inducted in Wikipedia’s “hall of fame” at http://en.wikipedia.org/wiki/Programming_paradigm (along with 75 others)

To Wit

“[...] the Design Patterns solution is to turn the programmer into a fancy macro processor.”

– Mark Dominus

The Point of PbD

*Fight combinatorics
with combinatorics*

Plenty of Room at the Bottom

“What would happen if we could arrange the atoms one by one the way we want them?”

– Richard P. Feynman

Core Idea

- Patterns: programmer “expands” mental macros
 - Total plasticity, no code reuse
- PBD: programmer assembles rigid macros
 - No plasticity, good code reuse
- DbI: programmer *molds* macros that communicate with, and adapt to, one another
 - Good plasticity, good code reuse

DbI Prerequisites

- DbI Input

DbI Prerequisites

- DbI Input
 - Introspect types: “What are your methods?”
 - Variant: “Do you support method "xyz"?”
 - Strongest: “Does this code compile?”

DbI Prerequisites

- DbI Input
 - Introspect types: “What are your methods?”
 - Variant: “Do you support method "xyz"?”
 - Strongest: “Does this code compile?”
- DbI Processing

DbI Prerequisites

- DbI Input
 - Introspect types: “What are your methods?”
 - Variant: “Do you support method "xyz"?”
 - Strongest: “Does this code compile?”
- DbI Processing
 - Arbitrary compile-time evaluation

DbI Prerequisites

- DbI Input
 - Introspect types: “What are your methods?”
 - Variant: “Do you support method "xyz"?”
 - Strongest: “Does this code compile?”
- DbI Processing
 - Arbitrary compile-time evaluation
- DbI Output

DbI Prerequisites

- DbI Input
 - Introspect types: “What are your methods?”
 - Variant: “Do you support method "xyz"?”
 - Strongest: “Does this code compile?”
- DbI Processing
 - Arbitrary compile-time evaluation
- DbI Output
 - Generate arbitrary code

Optional Interfaces

Optional Interfaces

- A DbI component typically prescribes:
 - n_r *required* primitives (may be 0)
 - n_o *optional* primitives
- Introspection queries for optionals
- What's missing as important as what's present

- Up to 2^{n_o} possible interfaces, in compact form!

Optional Interfaces: Aftermath

- Linear code for exponential behaviors
 - Includes state variations, too
- No penalty for fat interfaces
- Graceful degradation
 - Old: Less capable components \Rightarrow errors
 - New: Less capable components \Rightarrow reduced features

if constexpr Needs A Fix

Example: Hash table layout

```
template <class K, class V, size_t maxLength>
struct RobinHashTable {
    static if (maxLength < 0xFFFFE) {
        using CellIdx = uint16_t;
    } else {
        using CellIdx = uint32_t;
    }
    static if (sizeof(K) % 8 < 7) {
        alignas(8) struct KV {
            alignas(1)
                K k;
                uint8_t cellData;
            alignas(8)
                V v;
        }
    }
}
```

Example: Hash table layout

```
...
else {
    alignas(8) struct KV {
        K k;
        alignas(8)
        V v;
        alignas(1)
        uint8_t cellData;
    }
};
```


if constexpr

- Does not operate at run time

if constexpr

- Does not operate at run time
- Does not operate at compile time

if constexpr

- Does not operate at run time
- Does not operate at compile time
 - (We have regular `if` inside `constexpr` for that!)

if constexpr

- Does not operate at run time
- Does not operate at compile time
 - (We have regular `if` inside `constexpr` for that!)
- Operates at *design* time

if constexpr

- Does not operate at run time
- Does not operate at compile time
 - (We have regular `if` inside `constexpr` for that!)
- Operates at *design* time
 - That makes all the difference!

Each use of `if`
`constexpr` could double
the design space
covered

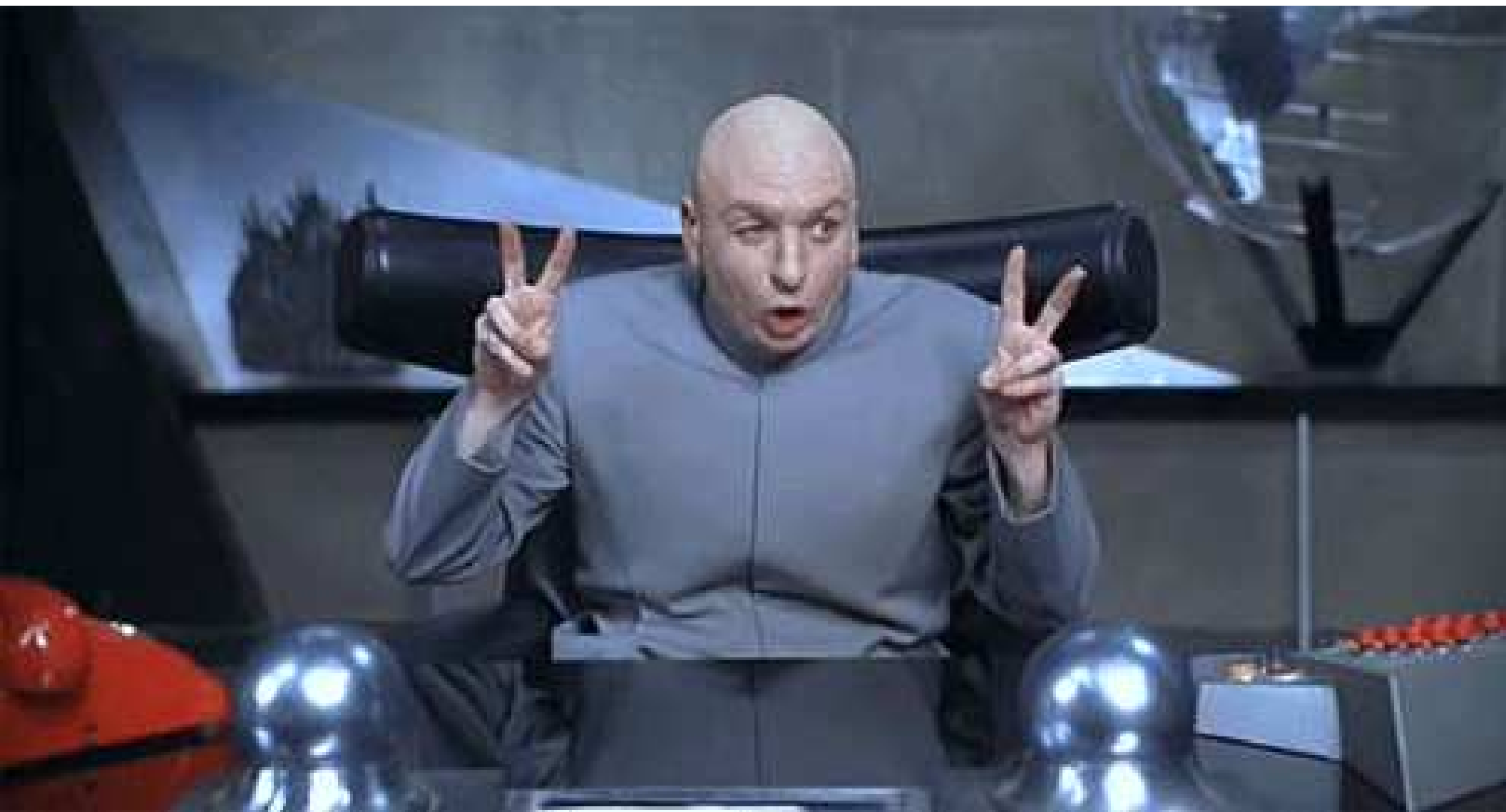
But!

**Must. Not! Introduce.
A. Scope!**

Design by Introspection Examples

Found in “another language”

- `std.experimental allocator`: unbounded allocator designs in 12 KLOC
 - `jemalloc`: 1 allocator in 45 KLOC
- Collections: work in progress
- `std.experimental.checkedint`: now



Checked Integrals

- `+`, `+=`, `-`, `-=`, `++`, `--`, `*`, `*=` may lose information
- Division by zero in `/`, `/=`
- Negating the most negative is negative for all signed types
- `-1 == UINT_MAX`, `-1 > 2u`

- That's pretty much it!

Design Challenges

- What gets checked: overflows? div0? negation? mixed-sign comparisons? conversions? some of the above—which?
- On violation: warn? abort? throw? log? fix/approximate?
- Type system integration: statically disallow some operators/conversions?
- Make it efficient (not easy!)
- Make it small
 - Proportional response
 - Not rocket surgery after all

Meta Design Challenges

- No trouble to implement *any given behavior*
- Much more difficult to allow behaviors that are *as of yet unspecified*
- Scaffolding scales poorly with behaviors
- “Sticker shock” of generic libraries
 - “You mean I need to use this 5 KLOC library coming with 20 pages of documentation to check a few overflows?”

Baselines (1/2)

- Mozilla's CheckedInt for C++
 - 0.8 KLOC (without docs, unittests)
 - Inefficient layout (“valid” bit with the integral)
 - Enforcement onus on user code
 - No configurability
 - Inefficient approach (checks separated from operations)
- Microsoft's SafeInt for C++
 - 7 KLOC
 - Lavish documentation

Baselines (2/2)

- `safe_numerics` for C++ by Robert Ramey
 - Policy-based Design in 5 KLOC (+ 5 KLOC tests)
 - Requires 6 Boost libs
- `checkedint` for D by T. S. Bockman
 - PbD in 5 KLOC, including docs

std.experimental.checkedint size

- 3 KLOC (code + unittests + documentation)
 - Code: 1200 LOC
 - Tests: 900 LOC
 - Documentation: 900 LOC
-
- Speed: comparable to hand-inserted checks
 - Flexibility: unbounded

Overall Design

- “Shell with hooks” approach
- Similar to PbD, but introspection-driven
- Shell: high-level language integration
- Hook: optional intercepts of ops/events
- Default hook: just abort on anything fishy

Stateless hook? No problem!

```
template <class T, class Hook = Abort>
struct Checked {
    private T payload;
    static if (state_size<Hook> > 0) Hook hook;
    else static Hook hook;
    ...
};
```

Use of Introspection

- Shell factors all commonalities
- Handles qualifiers
- Drives hooks
- Type system integration
- Composition mediation

- Uses introspection to “look” at hooks
 - What can you do?
 - What operation(s) are you interested in?
- One **static if** every 10.5 lines

Defined Hooks

- Abort
- Throw
- Warn: output issues to stderr
- ProperCompare: fix comparisons on the fly
- WithNaN: Reserve “not a number” value
- Saturate: sticky saturation instead of overflowing

- Your own
 - Average length: 50 lines

Design by Introspection

- Assembly with plastic, adaptable components
- Combine:
 - `static if`
 - Compile-time introspection
 - Compile-time evaluation
 - Code generation

Fix `if constexpr!`

Call The Destructors!