

## Menetrend szerint érkezik a C++20

**Bjarne Stroustrup, a programnyelv atyjával beszélgettünk a jövőre várható új kiadásról.**

**Az idei Craft konferencia ütős előadói gárdát sorakoztatott fel, közülük is kiemelkedett azonban Bjarne Stroustrup - a sztárfejlesztőt valószínűleg nem sokaknak kell bemutatni, hiszen a C++ megalkotójáról van szó, akivel a rendezvényen nekünk is alkalmunk nyílt néhány szót váltani, és akit nyelv múltjáról-jövőjéről faggattuk.**

Bjarne Stroustrup jelenleg a Morgan Stanley technológiai divíziójának igazgatójaként dolgozik, ahol elosztott rendszerek tervezésével foglalkozik - ami nem unalmas feladat, miután egy hasonló méretű bank hatalmas mennyiségű adatot mozgat rengeteg gép, ország között. Lényegében egy világméretű elosztott rendszerről van szó, amelyre ráadásul nagyon szigorú megbízhatósági elvárások vonatkoznak, egy tranzakció elvesztése nem csak komoly probléma, de törvénybe ütköző is lehet. A fejlesztő mindennapjait a hasonló, méretes elosztott rendszerek kihívásai mellett az egyetemi oktatás tölti ki, a Columbia Egyetemen tart előadásokat, emellett pedig a C++ szabványért felelős bizottság munkáját is felügyeli - miután a programnyelv megalkotójáról beszélünk, ez utóbbi nem különösebben meglepő szerepkör.

## Négy évtizede tartó munka

Magát a C++-t egy egészen konkrét probléma hívta életre 1979-ben: ahogy Dr. Stroustrup felidézte, egy UNIX klaszter építésén dolgozott, aminek során komoly nehézséget jelentett számára, hogy nem állt rendelkezésére egy olyan nyelv, amely képes kezelni a memória- és folyamatkezelőkhöz, illetve az illesztőprogramokhoz hasonló alacsony szintű komponenseket, valamint a magas szintű feladatokkal is megbirkóznán mint az absztrakció. Noha már akkor rendelkezésre álltak a hardverközeli feladatokra használható nyelvek és az absztrakcióra alkalmas programnyelvek is, mindkettővel egyszerre nem volt amelyik meg tudott volna küzdeni. Ennek a problémának az orvoslására született meg a C++, amely a GitHub tavalyi évre vonatkozó statisztikái szerint 2018-ra már globálisan az ötödik legnépszerűbb programozási nyelvvé nőtte ki magát.

## C++20 is arriving on schedule

**We spoke with Bjarne Stroustrup, the father of the programming language, about the new edition's launch next year**

**This year's Craft conference featured an exceptional list of presenters, but even among them Bjarne Stroustrup stood out. The star developer probably doesn't need to be introduced, since he is the creator of C++, and with whom we had the chance to talk about the past and the future of the programming language at the event.**

Bjarne Stroustrup is a Managing Director in the technology division of Morgan Stanley where he works on the development of distributed systems. It is a challenging task, because a bank of Morgan Stanley's size moves enormous amounts of data among a large number of computers and countries. Essentially it is a worldwide distributed system, which is subject to very strict reliability requirements; a loss of only one transaction could have significant consequences. In addition to the challenges of similarly large distributed systems, the developer's everyday life is completed by lecturing at Columbia University and supervising the work of the committee responsible for the C++ standard – it is not surprising, since we are talking about the creator of the programming language.

## The work of the past four decades

C++ itself was brought to life by a very specific problem in 1979. As Dr. Stroustrup remembers, he was working to build a UNIX cluster when he faced the problem of not having a programming language that could handle low-level codes like memory managers, process schedules and device drivers, as well as coping with high-level tasks like abstraction. There were already languages available for hardware-related tasks and programming languages for abstraction, but there were none that could cope with both at the same time. C++ was born to overcome this problem and, according to GitHub's statistics for last year, became the world's fifth most popular programming language by 2018.

A nyelv fejlődése persze azóta is folyamatos, és már a következő, C++20 néven emlegetett nagy kiadás is a csőben van, amely a jelenlegi C++17-et követi majd. Ahogy a nevezéktanból sejthető, a jelenleg használt főverzió 2017-ben látott napvilágot, a C++20 rajtja pedig 2020 végére várható. Dr. Stroustrup kérdéseinkre elmondta, az új kiadás legfontosabb újdonságai a modulok, a conceptek, vagy "konceptiók" bevezetése, a contractok, azaz szerződések megjelenése, valamint a korutinok lesznek - de több hasznos új library érkezése is várható.

Ahogy fogalmaz, a modulok egyrészt sokkal jobb "kódhigiénia" biztosítanak majd, miközben sokkal gyorsabb fordítást tesznek majd lehetővé, a korutinok pedig sokkal hatékonyabb pipeline-ok előtt nyitnak majd kaput, továbbá az aszinkron munkát is sokkal egyszerűbbé teszik. A nyelv atyjának személyes kedvencei ugyanakkor a konceptiók (concepts), amelyekkel a generikus programozás sokkal egyszerűbb és fenntarthatóbb lesz - azokkal ugyanis kiküszöbölhetővé válik a template-ek sokszor összetett szintaxisa. A megoldás további előnyei lehetnek az egyértelműbb hibaüzenetek is, a részletesebb típusellenőrzésnek hála.

A konceptiókat egyébként egyes újonnan érkező libraryk is használják, mint a range library - de a date library is hasznos frissítés, utóbbi nevéből sejthető módon a dátumok kezelését segíti - miután ez a feladat szinte minden vállalatnál, üzletnél előkerül valamilyen formában, sok fejlesztő veszi majd hasznát az egyszerűbb, elegánsabb dátumkezelésnek.

### **Késés nélkül jön az új kiadás**

Dr. Stroustrup kiemelte, hogy a C++20 nagyon komoly előrelépés lesz a jelenleg is használt C++17-hez képest - a szabvány leghamarabb 2020 végén érkezhetsz meg. Az új verzióval kapcsolatos technikai munka már jövő év tavaszán lezárulhat, ezután ugyanakkor a frissített szabványnak még hosszas jóváhagyási folyamatokon kell átesnie, egyebek mellett Genfben az ISO testülettől is zöld utat kell kapnia - a bizottság itt jellemzően kérvényez néhány formai változtatást, műszaki módosításokra azonban általában nincs már szükség. A vonatkozó dokumentumoknak tehát az élesedésig hosszú, bürokratikus utat kell bejárnia - ez ugyanakkor érthető, miután egy nemzetközi, számos iparágat

The development of the language has continued since then, and the release of the next revision called C++20, following C++17, is already in the pipeline. As you can predict from the names, the version presently used was launched in 2017 and the launch of C++20 is expected at the end of 2020. Dr. Stroustrup explained that the major improvements of the new addition are the modules, concepts, contracts and coroutines, but the arrival of new useful libraries can be expected as well.

As he explains, the modules will mean better "code-hygiene" while making translation a lot faster and coroutines will open lot more effective pipelines and will make asynchronous work much easier. The personal favorites of the father of the language are the concepts which make generic programming a lot easier and sustainable because their use can eliminate the often complex syntax of templates. Further advantages of the solution include clearer error messages, thanks to more detailed type checking.

The concepts are also used by some newly arriving libraries like the range library. The date library, helping the management of dates and calendars, is a useful update as well, since this task appears at almost all companies in some form and many developers will benefit from simpler, more elegant date management.

### **Next edition to be delivered on time**

Dr. Stroustrup emphasized that C++20 will have dramatic improvements compared to the presently used C++17. The standard may arrive at the end of 2020 the earliest. The technical work on the new version may be already completed by next spring, however, the updated version will undergo a long approval process, including getting the green light from the ISO body in Geneva—the board typically requests some formal change, but at this stage, usually, there is no need for technical changes. So the relevant documents need to undergo a long, bureaucratic process until they go live, but this is quite legitimate considering that the new standard affects many countries and industries.

érintő szabványról van szó, megvan a helye tehát a hasonló formalitásoknak.

A végleges szabvány közzétételére kitűzött, 2020 végi dátum egyébként korántsem elrugaszkodott terv, a C++17 és az azt megelőző C++14 is időben, menetrend szerint látott napvilágot. Persze a határidők betartása nem egyszerű feladat, annak érdekében sokszor egy-egy funkciót a következő kiadásra kell csúsztatni - a dolgot pedig tovább nehezíti, hogy a nyelv gondozásáért és fejlesztéséért felelős, nagyjából 350 tagot számláló bizottság a különböző vitás kérdéseket nem igazán szereti úgy eldönteni, hogy azokban ne lenne legalább 90 százalékos egyetértés. Ahogy Dr. Stroustrup elmondta, még a 80 százalék is marginálisnak számít. Az átfogó egyetértés ráadásul iparágakon, és nemzeteken is át kell hogyíveljen - ennek fényében nem meglepő, hogy egy-egy új nyelvverzió kiadása három évet vesz igénybe. Dr. Stroustrup egyébként már a koncepciókat is a C++17-ben akarta elérhetővé tenni, miután azonban akkor nem tudott elég embert meggyőzni az ügyről, a bevezetés végül a 2020-as kiadásra csúszott - ahogy fogalmaz, könnyű így három évet elveszíteni. A határidőket ugyanakkor kőkeményen tartani kell, hiszen ha egy elcsúszik, jó eséllyel a következő is el fog, ami hosszú távon komoly problémákhoz vezetne. Ezért egyszerűbb inkább az egyes funkciókat elhalasztani, ha az adott C++ kiadás vonata elment, meg kell várni a következőt.

A C++20 tehát sok fontos változást hoz majd, magában a nyelvben, illetve ebből adódóan az azzal kapcsolatos munkában, a fejlesztési folyamatokban is. Annak kapcsán, hogy pontosan hogyan fognak változni a C++ fejlesztők mindennapjai az új kiadás megjelenésével, Dr. Stroustrup szerint korai lenne találgatni, mindenesetre sok feladat könnyebbé válik majd. A modulokkal a fordítási-debugging ciklusok sokkal gyorsabbak lesznek, a fordítási idők lecsökkenésével pedig jó eséllyel a fejlesztők is bátrabban próbálhatják ki új funkciók beépítését az adott szoftverbe, illetve a debuggolási szokások is megváltozhatnak. A koncepciók bevezetése átrajzolja majd, hogy az emberek általában hogyan gondolkoznak a dizájnról a generikus programozásban - hogy pontosan hogy festenek majd ezek a változások, azt leghamarabb jövő év végén láthatjuk majd.

The deadline to publish the new edition by the end of 2020 seems feasible: C++17 and the earlier C++14 were delivered on schedule as well. Meeting deadlines is not an easy task, and in order to do so, it is often necessary to postpone the introduction of some new features for the next release. It is further complicated by the fact that the committee of approximately 350 members responsible for the maintenance and development of the language prefers to decide on a topic with at least an 80 to 90 percent majority. Dr. Stroustrup said that 80 percent was marginal. In light of the fact that a broad consensus needs to span across industries as well as countries, it is not surprising that the release of a new version takes three years. Dr. Stroustrup wanted to make the concepts available already in the C++17, but since he couldn't convince enough people to stand for the case, the introduction slipped to the 2020 launch. It is easy to waste 3 years like this, he says.

The deadlines, however, need to be kept, because if there is one delay, then there is a good chance that the next one will be late as well, which can cause serious problems in the long run. So it's easier to postpone certain functions if the given C++ revision is already on the way out and wait for the next one.

C++20 will bring many important changes in the language itself and in the work and development process related to it as well. According to Dr. Stroustrup, it is too early to tell how the everyday life of C++ developers will change, but many tasks will become easier for sure. Translating-debugging cycles will become a lot faster with these modules, and by the reduction of the translation time, developers are more likely to try to incorporate new features into the software and it will change debugging habits as well. Introducing concepts will redefine how people think about design in generic programming. To see what these changes look like, we don't need to wait until the end of next year; concepts are already shipping in GCC.

### **Már a 2023-as tervek is körvonalazódnak**

De természetesen már a későbbi verziókra vonatkozóan is vannak tervek, Dr.Stroustrup a potenciálisan a C++23-ra vagy akár 2026-os kiadásra csúszó funkciók között említette az executorokat, amelyek lényegében általános concurrency modulokként működnek, egyebek mellett a networking funkciók támogatására. Ha minden gördülékenyen halad, a statikus reflexiók (static reflection) is bekerülhetnek a C++23-ba - ugyanakkor a nyelv fejlődését rengeteg tényező befolyásolja, így egyelőre nehéz lenne megmondani, mi kerül be biztosan a következő kiadásba.

A C++20 újdonságai mellett a nyelv megalkotóját, akárcsak nemrég Kevlin Henney-t, megkérdeztük arról is, mit gondol a fejlesztői pálya népszerűsödéséről, illetve mit javasol azoknak a felnőtteknek, akik akár valamilyen egészen más iparágból érkeznek a területre, valamely manapság népszerű felnőttképző szolgáltatáson keresztül, vagy akár autodidakta módszerekkel. Noha Dr. Stroustrup saját tapasztalatokkal nem rendelkezik a hasonló váltásról, ahogy fogalmazott, az alapok megfelelő elsajátítása nélkülözhetetlen, sokan beleesnek abba a hibába, hogy túl gyorsan akarnak fejlesztők lenni, és úgy vágnak bele a gyakorlatba, hogy nem látják át a megcélzott rendszerek működését. Amennyi időt csak lehet rá kell tehát szánni az alapok bevésésére, persze sokaknak nehéz lehet megfogadni ezt a tanácsot, akik sietnek az új, vonzó területre váltani - de mindenképp jó tanácsról van szó, amelyet nem kevés tapasztalat nyomtatékosít.

### **The plans for 2023 are already under way**

But, of course, there are plans for later versions as well. Dr. Stroustrup mentioned functions that may potentially be postponed to the C++23 or the 2026 version: for instance executors, a general model for concurrency in libraries which support networking functions among others. If everything goes smoothly, static reflections may be included in C++23, however, the development of the language is influenced by a lot of factors, so it would be difficult to tell for now what is going to be included in the next release for certain.

In addition to the new features of C++20, we also asked the author of the language, just as we asked Kevlin Henney recently, about what he thinks about the growing popularity of the software development career, and what he suggests to those who come from a completely different industry, through a now popular adult education service or used an autodidactic approach to learn programming. Although Dr. Stroustrup did not have any experience like that in his own career, he said, knowledge of the basics is a must. Many people fall into the trap of wanting to be developers too quickly and start practicing, without having a solid foundation of the relevant computing fields. You need to devote as much time as possible to gain such a foundation. This piece of advice may be hard to follow for those who hurry to shift to this new, appealing profession, however, it is an idea, supported by considerate evidence that is worth considering.