

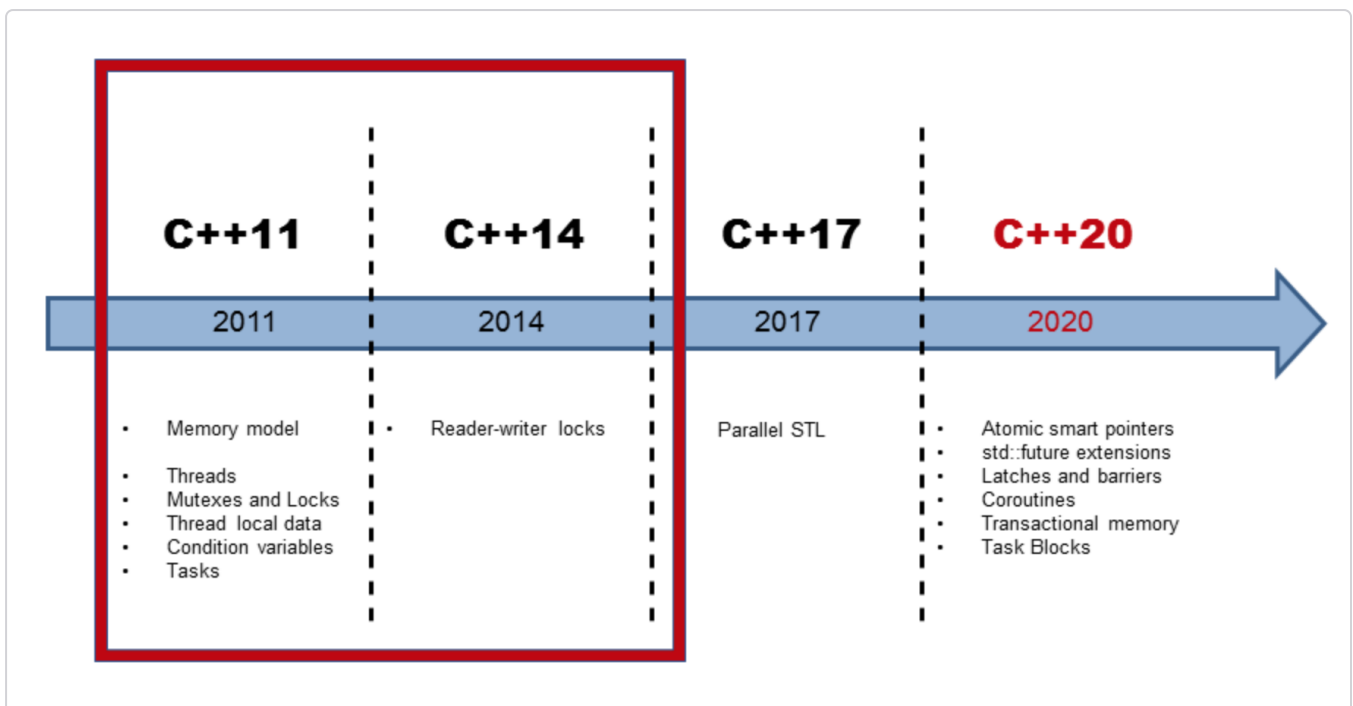
Introduction to Threads

This lesson gives an introduction to threads in C++.

WE'LL COVER THE FOLLOWING ^

- Further information

C++ has had a multithreading interface since C++11. This interface has all the basic building blocks for creating multithreaded programs: threads, synchronization primitives for shared data (e.g. [mutexes](#) and [locks](#)), thread-local data, synchronization mechanism for threads (e.g. condition variables), and [tasks](#). Tasks are usually called promises, and they provide a higher level of abstraction than native threads. It is okay if we do not understand the terms discussed here, as all of them will be discussed in depth in the following lessons.



Further information

- [mutexes](#)

- [locks](#)
- [tasks](#)

In the next lesson, we'll talk about how we can create threads in C++ using callable units.