What is the Difference Between Static and Dynamic Linking

July 13, 2019 • by Lithmee • 4 min read

1

The **main difference** between static and dynamic linking is that **static linking copies all library modules used in the program into the final executable file at the final step of the compilation while, in dynamic linking, the linking occurs at run time when both executable files and libraries are placed in the memory.**

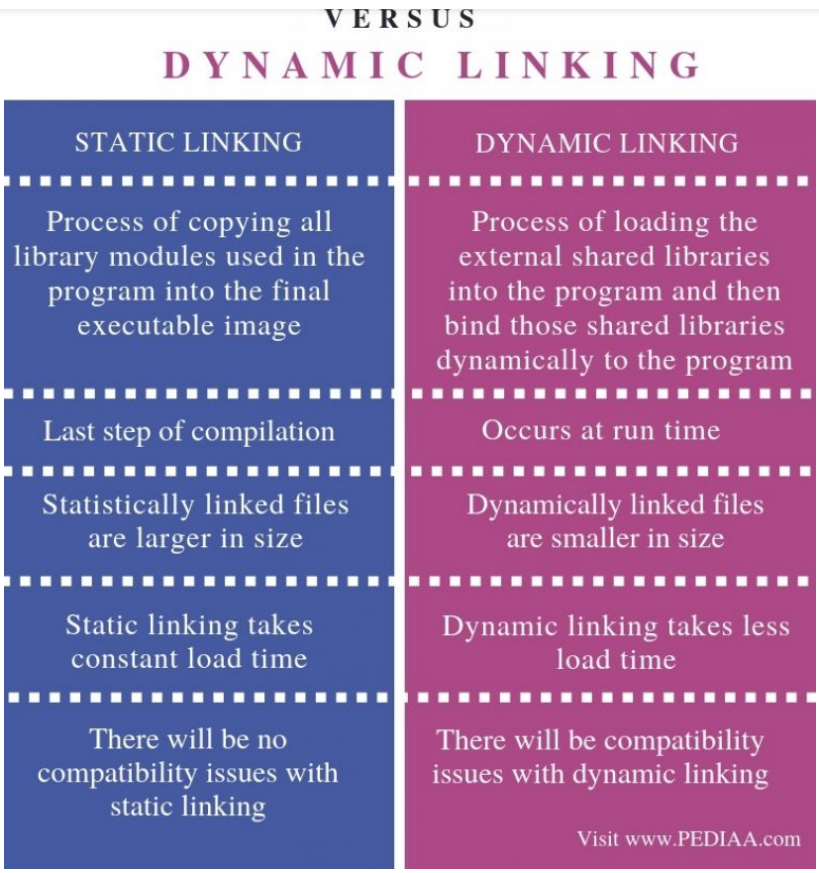
Generally, a computer program is a sequence of steps in a programming language that instructs the computer or the CPU to perform a certain task. Even though the programmer understands this program, the computer does not. Therefore, it is necessary to convert the source code to machine code. Also, this program might require other programs or libraries. In those cases, it is necessary to bring those programs or libraries together with this program to execute it. Thus, linking is the process of combining external programs with the programmer's program to execute it successfully. In overall, there are two linking mechanisms as the static and dynamic linking.

Key Areas Covered

1. **What is Static Linking**
   -Definition, Functionality
2. **What is Dynamic Linking**
   -Definition, Functionality
3. **Difference Between Static and Dynamic Linking**
   -Comparison of key differences

Key Terms

Dynamic Linking, Library, Linker, Static Linking



What is Static Linking

Static linking copies all the libraries required for the program into the final executable file. The linker performs this task, and it is the last step of compilation. The linker combines the relevant libraries with the program code to resolve external references. Finally, the linker generates an executable file suitable for loading into memory. The final statistically linked file contains the calling program and called programs. Generally, these files are large because they are connected with other files.

Assume that there are some changes in the external programs. In that case, it is necessary to recompile and re-link again. Otherwise, the existing executable file does not reflect these changes. Furthermore, in statistical linking, all the modules and libraries are available as a single executable module. Therefore, this linking is faster and does not cause compatibility issues.

What is Dynamic Linking

In dynamic linking, the names of the external libraries /shared libraries are copied into the final executable; thus, the real linking occurs at run time when the executable file and libraries load to the memory. The operating system performs dynamic linking. Generally, there is only one copy of a shared library is in the memory. Therefore, the size of the executable file is lower. It is possible to update and recompile the external libraries. Moreover, if the shared library code is already available in memory, there will be less load time.

Usually, in dynamic linking, it is better to have a compatible library. If there is a modification in the library, the application has to find a way to make it compatible with the new version of the library. Additionally, removing the library can cause the program not to work further.

Difference Between Static and Dynamic Linking

Definition

Static linking is the process of copying all library modules used in the program into the final executable image. In contrast, dynamic linking is the process of loading the external shared libraries into the program and then binds those shared libraries dynamically to the program. Thus, this is the main difference between static linking and dynamic linking.

Moreover, static linking is the last step of compilation, while dynamic linking occurs at run time.

File size

While statistically linked files are larger in size, dynamically linked files are smaller in size.

Load time

Besides, static linking takes constant load time while dynamic linking takes less load time. Hence, this is another difference between static linking and dynamic linking.

Compatibility

Furthermore, there will be no compatibility issues with static linking. On the other hand, there will be compatibility issues with dynamic linking.

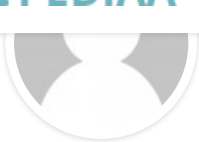
Conclusion

In brief, static and dynamic linking are two linking mechanisms. The main difference between static and dynamic linking is that static linking copies all library modules used in the program into the final executable file at the final step of the compilation while in dynamic linking, the linking occurs at run time when both executable files and libraries are placed in the memory.

References:

1.“Dynamic Linker.” Wikipedia, Wikimedia Foundation, 25 Mar. 2019, Available here.

About the Author: Lithmee



sharing her knowldge in the areas of programming, data science, and computer systems.

View all posts

You May Also Like These

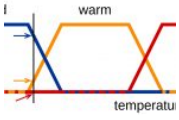
Leave a Reply

You May Like

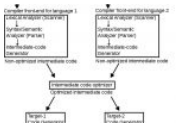
What is the Difference Between Agile and Iterative

What is the Difference Between Object Code and...

What is the Difference Between Source Program and...



What is the Difference Between Fuzzy Logic and...



What is the Difference Between Syntax Analysis and...

What is the Difference Between Millet and Sorghum

What is the Difference Between Common Sense and Critical Thinking

What is the Difference Between Grains and Cereals

What is the Difference Between Social Exclusion and Vulnerability

What is the Difference Between Proactive and Retroactive Interference

What is the Difference Between Etching and Engraving