.. note::

It is assumed that the linker specified is fully compatible with the default

one the compiler would normally invoke. CMake will not do any option

translation.

Linker types are case-sensitive and may only contain letters, numbers and

underscores. Linker types defined in all uppercase are reserved for CMake's own

built-in types. The pre-defined linker types are:

``DEFAULT``

This type corresponds to standard linking, essentially equivalent to the

:prop\_tgt:`LINKER\_TYPE` target property not being set at all.

``SYSTEM``

Use the standard linker provided by the platform or toolchain. For example,

this implies the Microsoft linker for all MSVC-compatible compilers.

This type is supported for the following platform-compiler combinations:

\* Linux: ``GNU``, ``Clang``, ``LLVMFlang``, ``NVIDIA``, and ``Swift``

compilers.

\* Apple platforms: ``AppleClang``, ``Clang``, ``GNU``, and ``Swift``

compilers.

\* Windows: ``MSVC``, ``GNU``, ``Clang``, ``NVIDIA``, and ``Swift`` compilers.

``LLD``

Use the ``LLVM`` linker. This type is supported for the following

platform-compiler combinations:

\* Linux: ``GNU``, ``Clang``, ``LLVMFlang``, ``NVIDIA``, and ``Swift``

compilers.

\* Apple platforms: ``Clang``, ``AppleClang``, and ``Swift`` compilers.

\* Windows: ``GNU``, ``Clang`` with MSVC-like front-end, ``Clang`` with

GNU-like front-end, ``MSVC``, ``NVIDIA`` with MSVC-like front-end,

and ``Swift``.

``BFD``

Use the ``GNU`` linker. This type is supported for the following

platform-compiler combinations:

\* Linux: ``GNU``, ``Clang``, ``LLVMFlang``, and ``NVIDIA`` compilers.

\* Windows: ``GNU``, ``Clang`` with GNU-like front-end.

``GOLD``

Supported on Linux platform with ``GNU``, ``Clang``, ``LLVMFlang``,

``NVIDIA``, and ``Swift`` compilers.

``MOLD``

Use the `mold linker <https://github.com/rui314/mold>`\_. This type is

supported on the following platform-compiler combinations:

\* Linux: ``GNU``, ``Clang``, ``LLVMFlang``, and ``NVIDIA`` compilers.

\* Apple platforms: ``Clang`` and ``AppleClang`` compilers (acts as an

alias to the `sold linker`\_).

``SOLD``

Use the `sold linker`\_. This type is only supported on Apple platforms

with ``Clang`` and ``AppleClang`` compilers.

``APPLE\_CLASSIC``

Use the Apple linker in the classic behavior (i.e. before ``Xcode 15.0``).

This type is only supported on Apple platforms with ``GNU``, ``Clang``,

``AppleClang``, and ``Swift`` compilers.

``MSVC``

Use the Microsoft linker. This type is only supported on the Windows

platform with ``MSVC``, ``Clang`` with MSVC-like front-end, and ``Swift``

compilers.

.. \_sold linker: https://github.com/bluewhalesystems/sold