

# System Requirements

- [Windows](#)
  - [Web Servers](#)
- [Linux/Unix](#)
  - [Web Servers](#)
- [Mac OS](#)
  - [Web Servers](#)

The Service Provider is written in fairly portable C++ with dependencies on a mixture of C and C++ libraries, and is in principle buildable on most Operating Systems, but we provide official support for only the platforms below at this time, which also correspond to the exact platforms for which we produce some form of official packaging.

Generally we only support versions of an OS that are under general security maintenance by the vendor; that is, once you have to pay for updates privately, we stop officially supporting it and reserve the right to produce updated versions that depend on features or libraries only available in newer versions of the platform.

We do not have any particular resource requirements, as the SP generally supports whatever an appropriate load for a given web server would otherwise be to at least an approximation. For a very heavily loaded system or one relying on large metadata sources, reserving 1G of RAM or so for the service daemon is likely wise. The SP should scale fairly linearly with the number of cores available and works best with (almost to the point of requiring) web servers that rely on long-running processes with threads and not frequently cycled child processes.

## Windows

We support Windows Server 2008 and later. Use of Windows client versions that have been released since Windows Server 2008 is almost always possible but not officially supported.

We provide installers for both 32-bit and 64-bit Windows and that is the only supported means of installation.

We do not support Itanium or ARM based systems.

Building from source is possible (and somewhat easier than in the past) but this is still beyond the capabilities of most and not encouraged unless you have a need to produce extensions to the software (or need the FastCGI modules, see below).

## Web Servers

We support the versions of the IIS web server that are provided with the supported Windows versions. A legacy ISAPI module is provided for compatibility but will be discontinued in a future version, and the new IIS native module should be used instead.

We also support the use of Apache 2.2 and 2.4 as built by the [Apache Lounge](#) site, and it is not a requirement to obtain a version built with the exact same compiler as we use, as this has been tested fairly thoroughly and mixing them does not appear to cause any problems. We no longer provide pre-built modules for older Apache versions, as they are no longer supported by the ASF.

For legacy reasons, we continue to provide an NSAPI module for the old Netscape/Sun/Oracle web server but this will be discontinued in a future version and should be viewed as a temporary aid to migrating to something else.

We do not provide pre-built FastCGI support on Windows because the supporting library for that is not actively maintained and we have chosen not to own that responsibility. It is possible to build them from source but we will not officially support that at this time.

## Linux/Unix

We officially support the following Linux distributions:

- Red Hat Enterprise and CentOS 6, 7, 8
- SUSE Linux Enterprise Server 12SP4

The reason for this policy, aside from limiting our scope, has to do with packaging: these are the versions we explicitly provide packages for (in the form of RPMs built by the OpenSUSE build service under the oversight of the project team) and are therefore the versions for which we can provide security updates.

We may produce packages for older or other unsupported platforms at the discretion of the project team, but do so solely as a service to the community and do not officially support versions other than those above.

Note that support for OpenSUSE itself has essentially ended because a third party has produced packages for the SP that are included with that OS, so our support is limited to the older official versions for which we provided packages.

We are discussing with the Consortium's members how best to tackle some form of support for other distributions such as Debian that would delineate our role vs. the role of the packaging teams. In practice, we do not simply ignore bug reports or questions about other distributions since the majority are of general applicability, but there are cases where more esoteric bugs are in fact limited to a platform and that is something we take into consideration.

It is generally possible to build the software and all dependencies on most Unix, Linux, and BSD versions (with problems likely cropping up on non-x86/64 architectures or the more unusual stuff like AIX that tends to have poor open source support generally).

Note that Solaris is no longer officially supported as of V3, which is a change from V2. We don't deliberately seek to break the build there, but in practice most of our releases will tend to need minor patches to deal with fairly obscure C++ compiler differences, so a clean build is unlikely. The extra work necessary to keep the build working there is one of the reasons we dropped it.

## Web Servers

Officially we support the use of Apache 2.2 and 2.4 and FastCGI. Functionally, the source continues to include older Apache module support.

We do not recommend the use of the old prefork MPM and strongly encourage the worker MPM. The prefork option will fail under load in a variety of cases, with some limited workarounds possible.

On the supported Linux platforms, we provide the modules that are native to the version of Apache provided by the upstream OS vendor, so the version built depends on what happens to be available.

We provide the FastCGI modules when the requisite libraries are available on the OS.

## Mac OS

We maintain [MacPorts](#) of the SP and most of its dependencies and support the use of the SP when those ports are used. We have essentially no capability to test on non-current Mac OS releases so only the latest 10.X release at any given time is officially supported.

## Web Servers

As of this version, we now provide port variants that allow the SP to be built against either Apache as provided by Apple, which has been deprecated, or from the MacPorts version. For compatibility, the default port variant continues to assume Apple's.

We do not recommend the use of the old prefork MPM and strongly encourage the worker MPM. The prefork option will fail under load in a variety of cases, with some limited workarounds possible.