

Apache MPM leader

Description:	An experimental variant of the standard <code>worker</code> MPM
Status:	MPM
Module Identifier:	<code>mpm_leader_module</code>
Source File:	<code>leader.c</code>

Summary

Warning

This MPM is experimental, so it may or may not work as expected.

This is an experimental variant of the standard `worker` MPM. It uses a Leader/Followers design pattern to coordinate work among threads. For more info, see <http://deuce.doc.wustl.edu/doc/pspdfs/lf.pdf>.

To use the `leader` MPM, add `--with-mpm=leader` to the configure script's arguments when building the httpd.

This MPM depends on APR's atomic compare-and-swap operations for thread synchronization. If you are compiling for an x86 target and you don't need to support 386s, or you are compiling for a SPARC and you don't need to run on pre-UltraSPARC chips, add `--enable-nonportable-atomics=yes` to the configure script's arguments. This will cause APR to implement atomic operations using efficient opcodes not available in older CPUs.

Directives

<code>AcceptMutex</code>	➡	<code>MinSpareThreads</code>	➡
<code>CoreDumpDirectory</code>	➡	<code>PidFile</code>	➡
<code>Group</code>	➡	<code>ScoreBoardFile</code>	➡
<code>Listen</code>	➡	<code>SendBufferSize</code>	➡
<code>ListenBacklog</code>	➡	<code>ServerLimit</code>	➡
<code>LockFile</code>	➡	<code>StartServers</code>	➡
<code>MaxClients</code>	➡	<code>ThreadLimit</code>	➡
<code>MaxMemFree</code>	➡	<code>ThreadsPerChild</code>	➡
<code>MaxRequestsPerChild</code>	➡	<code>User</code>	➡
<code>MaxSpareThreads</code>	➡		

(➡ This directive is defined elsewhere. See: `mpm_common`)