OpenMP Implementation-Defined Behaviors for OMPi v2.0.0

This document enlists behaviors of the OMP*i compiler, which are described as implementation-defined in the OpenMP specifications.*

1 Internal Control Variables (ICVs)

Specific ICVs and their initial values are given in the table below. Please notice that OMPi can utilize multiple and different threading libraries, each of which may have its own initial / default values for some of these ICVs; library-specific ICVs are marked by an asterisk (*). Values in the table below refer to the default threading library (pthreads).

ICV name	Initial value
nthreads-var	# available cores (*)
dyn-var	`true` (*)
run-sched-var	`static`
def-sched-var	`static`
bind-var	`true` (*)
stacksize-var	OS default value (*)
wait-policy-var	`active` (*)
thread-limit-var	no limit (*)
max-active-levels-var	no limit (*)
place-partition-var	`threads`
default-device-var	1 if at least 1 device attached, 0 (host) if none attached

(*) threading-library specific.

2 Dynamic Adjustment of Threads

When the dynamic adjustment of threads is disabled and the threading library cannot provide the requested number of threads, the application is aborted with an advice to enable dynamic adjustment either programmatically or using the corresponding environmental variable.

3 Loop Directive

- If the runtime schedule has been selected and the *run-sched-var* ICV is set to `auto`, then the iterations are distributed using a `static` schedule.
- For a collapsed loop, the variable used to compute the iteration count is of type `int`.

4 Sections Construct

Scheduling of the structured `section` blocks among threads is competitive; threads are assigned `section` blocks on a first-come first-serve basis.

5 Single Construct

The selection of a thread to execute a structured `single` block is competitive; the first thread to ask for it, gets to execute it.

6 Processor

A 'processor' is whatever the OS system calls report.

7 Device

Devices are the host and any additional attached compute units. In this version, OMPi can support the Adapteva Epiphany accelerator as a device.

8 Runtime Routines

- *omp_set_num_threads()* : if the argument is not a positive integer, then it is assumed to be equal to 1.
- *omp_set_schedule()* : there are no additional loop schedules defined.
- *omp_set_max_active_levels()* : if the argument is not a positive integer, then the routine simply returns. If it is called from within a explicit parallel regiion, the binding thread set is all threads.
- *omp_get_max_active_levels()* : if it is called from within a explicit parallel regiion, the binding thread set is all threads.

9 Environmental Variables

- `OMP_SCHEDULE` : if the value of the variable does not conform to the specified format, it is considered to be equal to `auto`.
- `OMP_NUM_THREADS` : if any list value is not a positive integer, the default number of threads is used (which is threading-library specific; see the table above). It the requested number of threads exceeds the capabilities of the threading library (which cannot happen when using the default threading library) the program is aborted with an informative message.
- `OMP_PROC_BIND` : if the value of the variable is not `true`, `false` or a comma-seperated list of `master`, `close` or `spread`, then it is considered to be `true`.
- `OMP_PLACES` : the only abstract names recognized are `threads`, `cores` and `sockets`. If creating a place by appending the number n to an abstract name, then a) if n is greater than the available resourses, the default places are used and b) if n is smaller than the available resources, the first n resources are used.
- `OMP_DYNAMIC, OMP_NESTED` : if the value of the variable is neither `true` nor `false`, then it is considered to be `false`.

- `OMP_STACKSIZE` : if the value of the variable does not conform to the specified format, it is considered to be equal to 256KB.
- `OMP_WAIT_POLICY` : if the value of the variable is neither active nor passive, it is ignored. The details of the wait policy behavior are threading-library specific. For the default threading library, the wait policy is always active irrespectively of the value of this environmental variable.
- `OMP_MAX_ACTIVE_LEVELS` : if the value is not a positive integer, it is ignored. The default behavior, for the case where the requested number of levels is larger than what can be supported, is threading-library specific (does not concern the default threading library).
- `OMP_THREAD_LIMIT` : if the value is not a positive integer, it is ignored. The default behavior, for the case where the requested number of threads is larger than what can be supported, is threading-library specific (does not concern the default threading library).