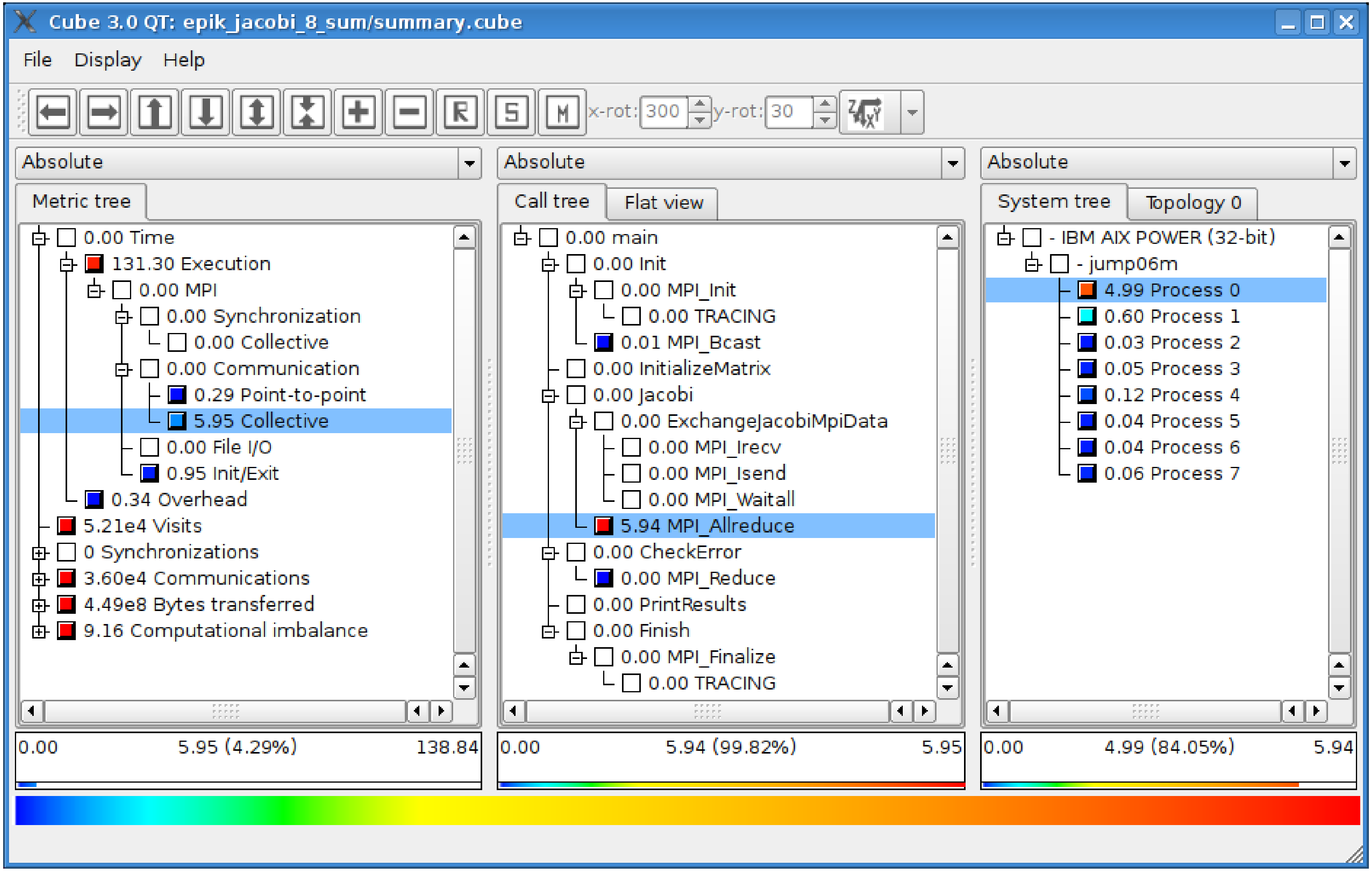
Scalasca performance properties

“The metrics tour“

Markus Geimer
m.geimer@fz-juelich.de

Scalasca analysis result



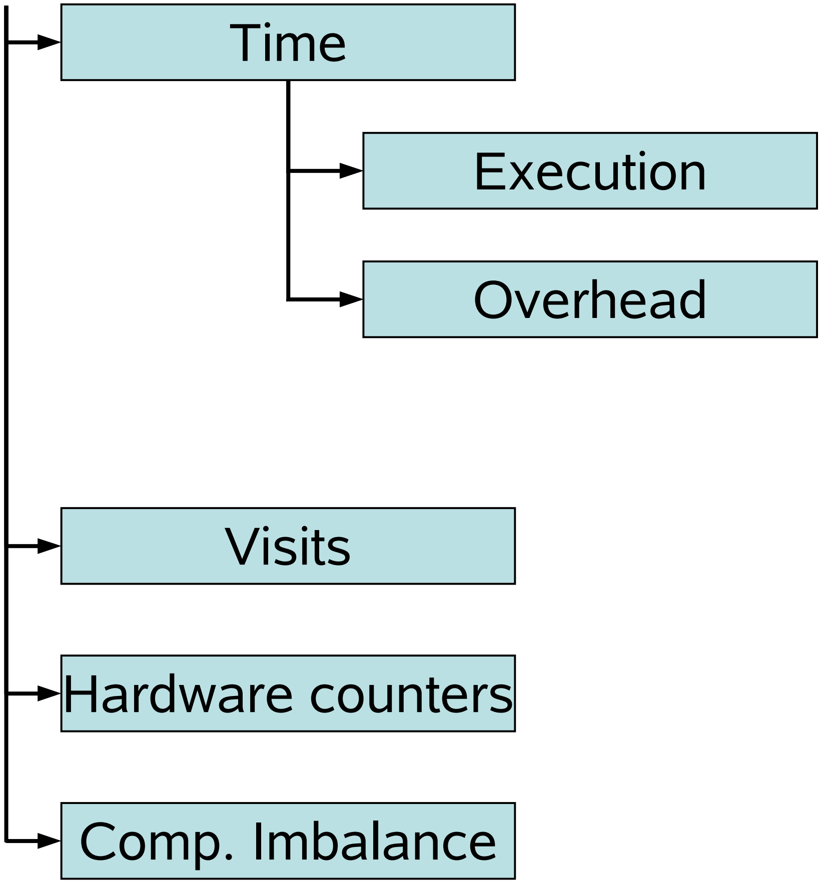
```
Metric tree
⊟ □ 0.00 Time
   ⊟ 🟥 131.30 Execution
      ⊟ □ 0.00 MPI
         ⊟ □ 0.00 Synchronization
            └ □ 0.00 Collective
         ⊟ □ 0.00 Communication
            ├ 🟦 0.29 Point-to-point
            └ 🟦 5.95 Collective
         ├ □ 0.00 File I/O
         └ 🟦 0.95 Init/Exit
      └ 🟦 0.34 Overhead
├ 🟥 5.21e4 Visits
⊞ □ 0 Synchronizations
⊞ 🟥 3.60e4 Communications
⊞ 🟥 4.49e8 Bytes transferred
⊞ 🟥 9.16 Computational imbalance

0.00          5.95 (4.29%)          138.84
```

```
Call tree
⊟ □ 0.00 main
   ⊟ □ 0.00 Init
      ⊟ □ 0.00 MPI_Init
         └ □ 0.00 TRACING
      └ 🟦 0.01 MPI_Bcast
   ├ □ 0.00 InitializeMatrix
   ⊟ □ 0.00 Jacobi
      ⊟ □ 0.00 ExchangeJacobiMpiData
         ├ □ 0.00 MPI_Irecv
         ├ □ 0.00 MPI_Isend
         └ □ 0.00 MPI_Waitall
      └ 🟥 5.94 MPI_Allreduce
   ⊟ □ 0.00 CheckError
      └ 🟦 0.00 MPI_Reduce
   ├ □ 0.00 PrintResults
   ⊟ □ 0.00 Finish
      ⊟ □ 0.00 MPI_Finalize
         └ □ 0.00 TRACING

0.00          5.94 (99.82%)          5.95
```

```
System tree
⊟ □ - IBM AIX POWER (32-bit)
   ⊟ □ - jump06m
      ├ 🟧 4.99 Process 0
      ├ 🟦 0.60 Process 1
      ├ 🟦 0.03 Process 2
      ├ 🟦 0.05 Process 3
      ├ 🟦 0.12 Process 4
      ├ 🟦 0.04 Process 5
      ├ 🟦 0.04 Process 6
      └ 🟦 0.06 Process 7

0.00          4.99 (84.05%)          5.94
```

Generic metrics

Generic metrics



Time

Total CPU allocation time

Execution

Execution time without overhead

Overhead

Time spent in tasks related to measurement (Does not include per-function perturbation!)

Visits

Number of times a function/region was executed

Hardware counters

Aggregated counter values for each function/region

Comp. Imbalance

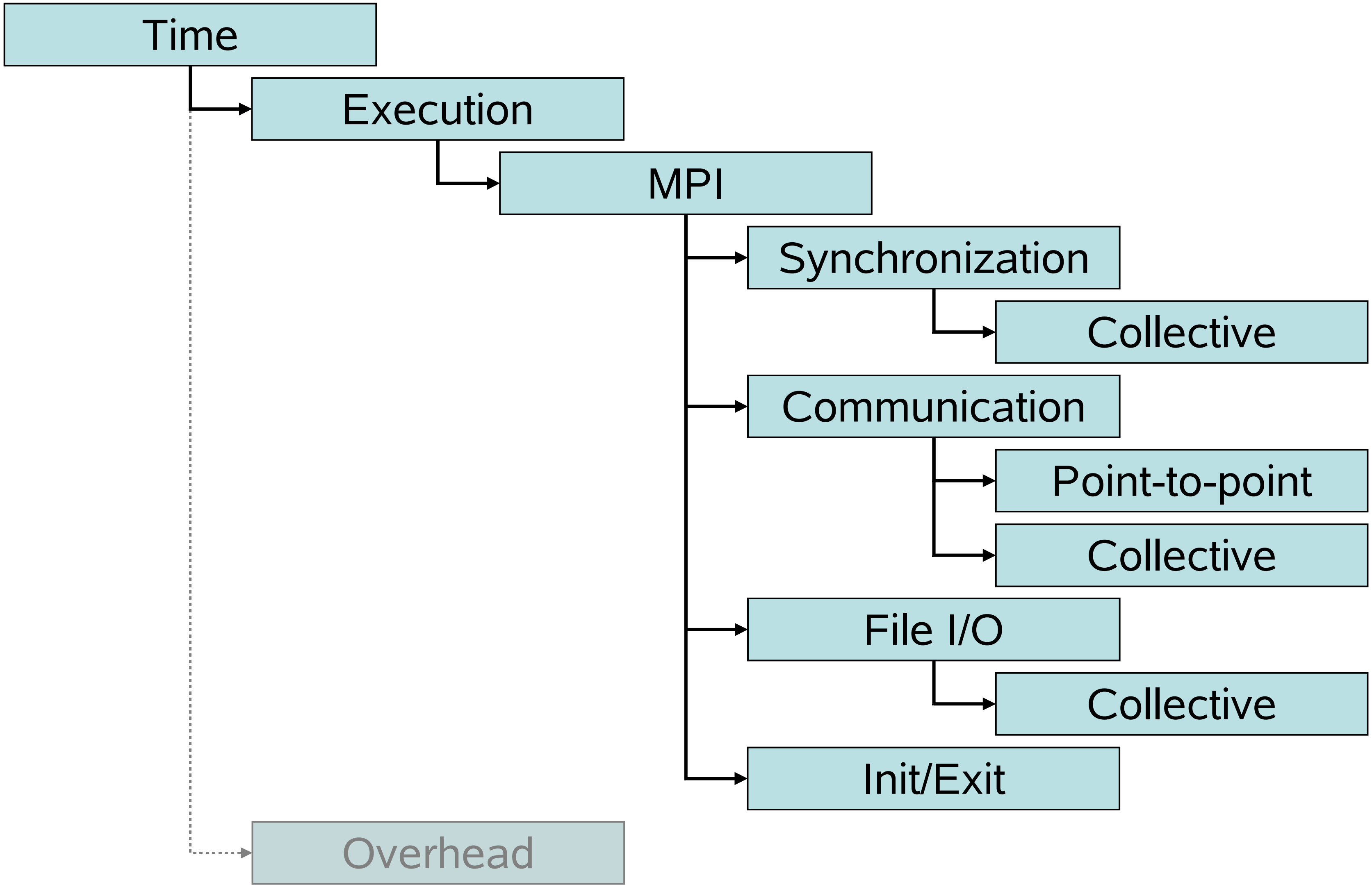
Simple load imbalance heuristic

Computational imbalance

- Simple load imbalance heuristic
- Focuses only on computational parts
- Easy to calculate
  - Absolute difference to average exclusive execution time
- Captures global imbalances
  - Based on entire measurement
  - Does not compare individual instances of function calls
- High value = Imbalance in the sub-calltree underneath
  - Expand the subtree to find the real location of the imbalance

MPI-related metrics

MPI Time hierarchy



MPI Time hierarchy details

MPI Time spent in pre-instrumented MPI functions

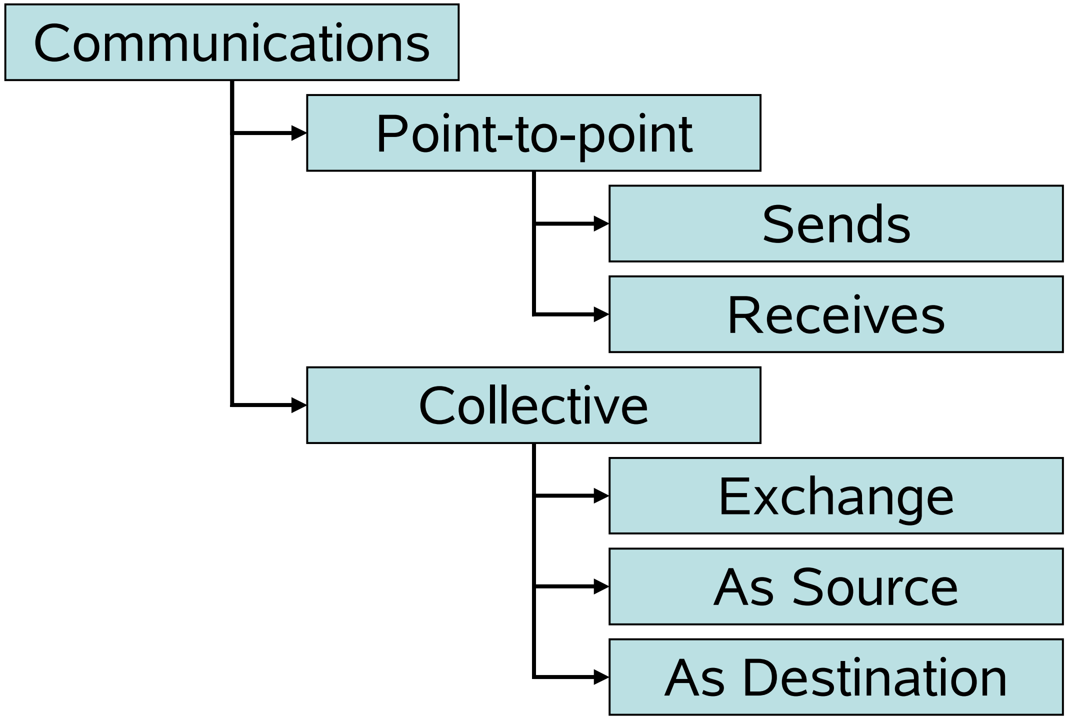
Synchronization Time spent in calls to `MPI_Barrier`

Communication Time spent in MPI communication calls, subdivided into collective and point-to-point

File I/O Time spent in MPI file I/O functions, with specialization for collective I/O calls

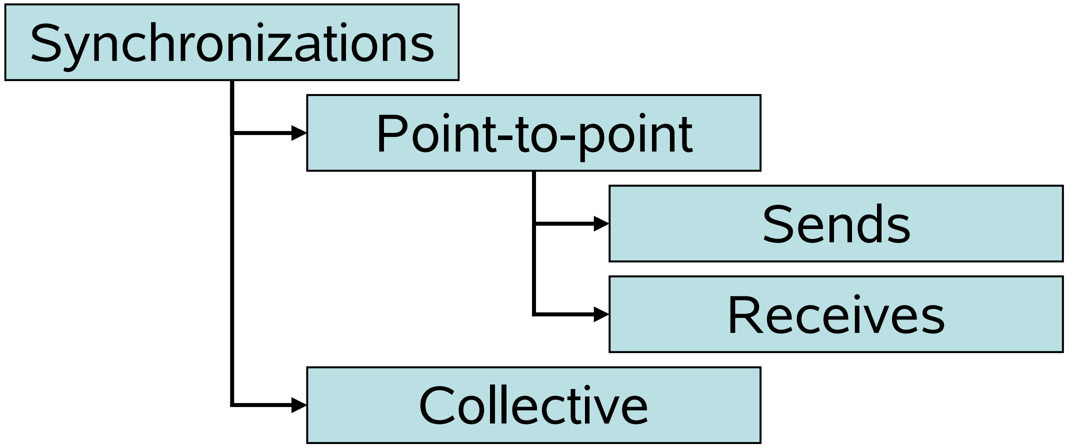
Init/Exit Time spent in `MPI_Init` and `MPI_Finalize`

MPI Communications hierarchy



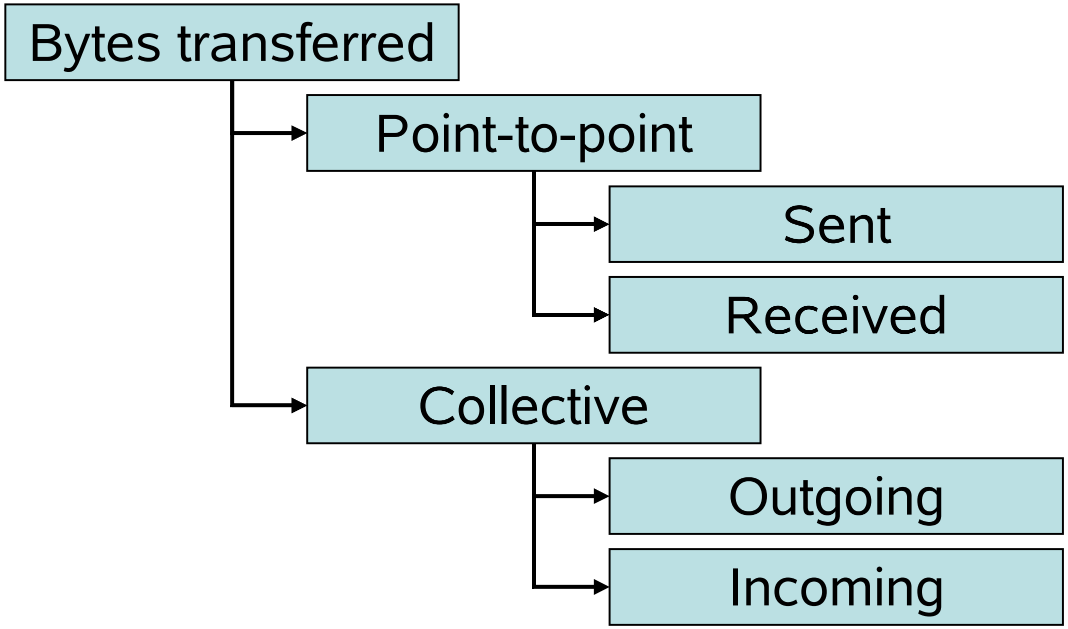
- Provides the number of calls to an MPI communication function of the corresponding class
- Zero-sized message transfers are considered synchronization!

MPI Synchronizations hierarchy



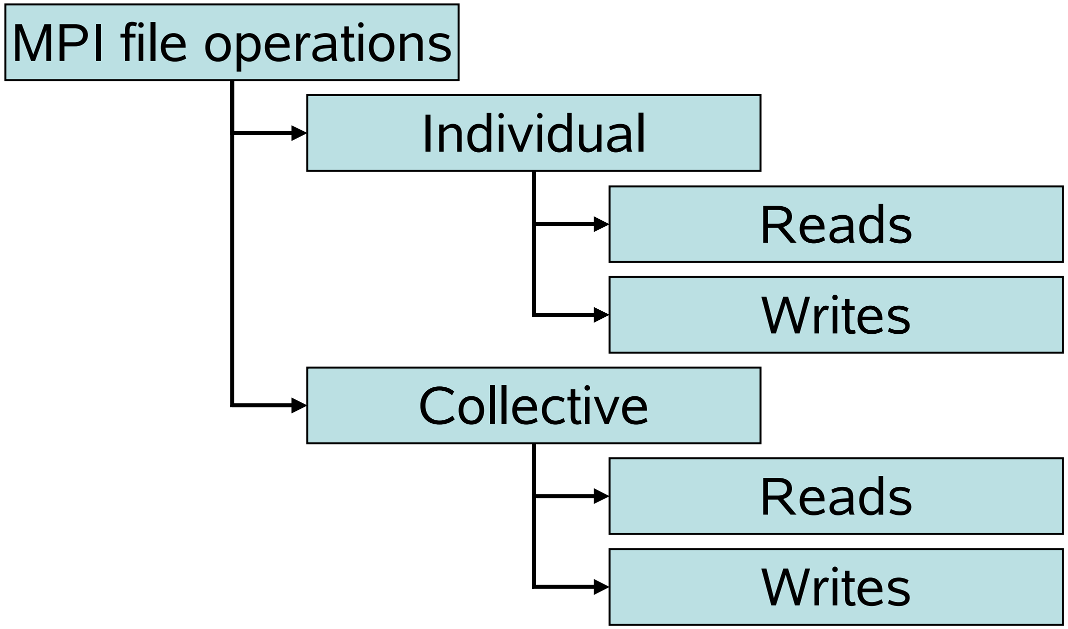
- Provides the number of calls to an MPI synchronization function of the corresponding class
- Synchronizations include zero-sized message transfers!

MPI Transfer hierarchy



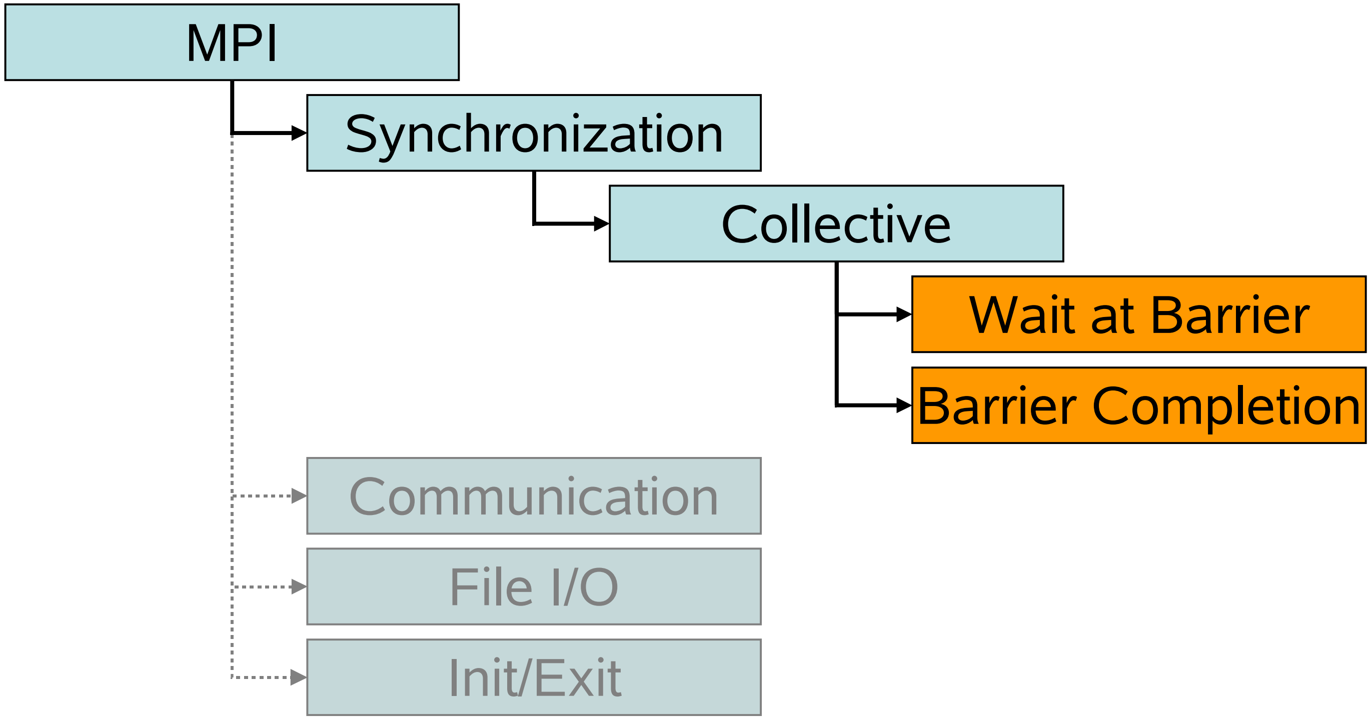
- Provides the number of bytes transferred by an MPI communication function of the corresponding class

MPI File operations hierarchy

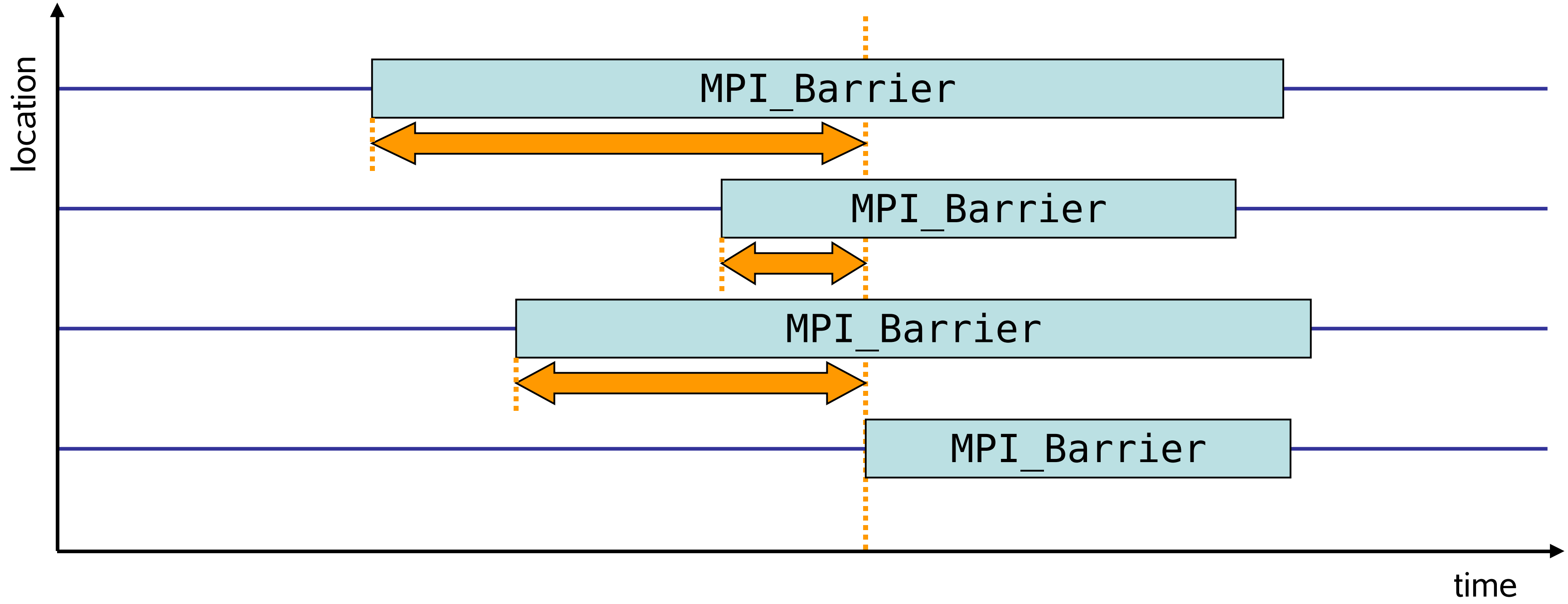


- Provides the number of calls to MPI file I/O functions of the corresponding class

MPI collective synchronization time

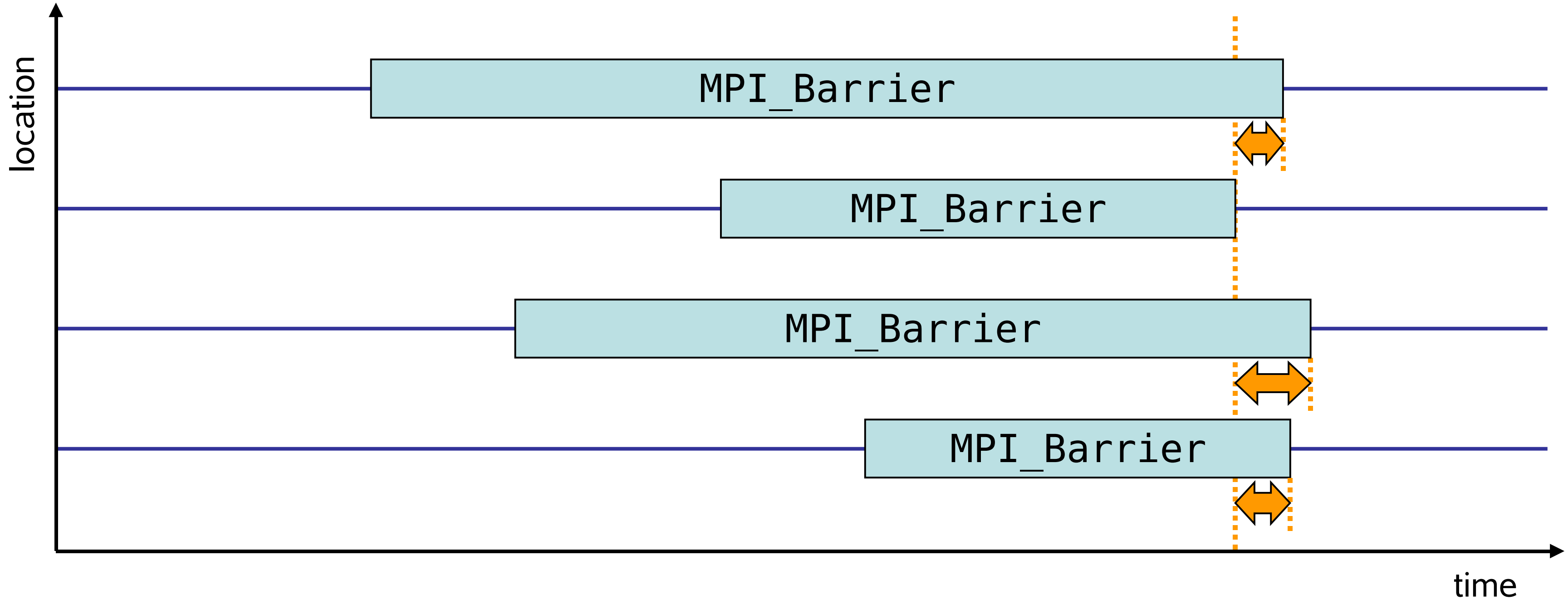


Wait at Barrier



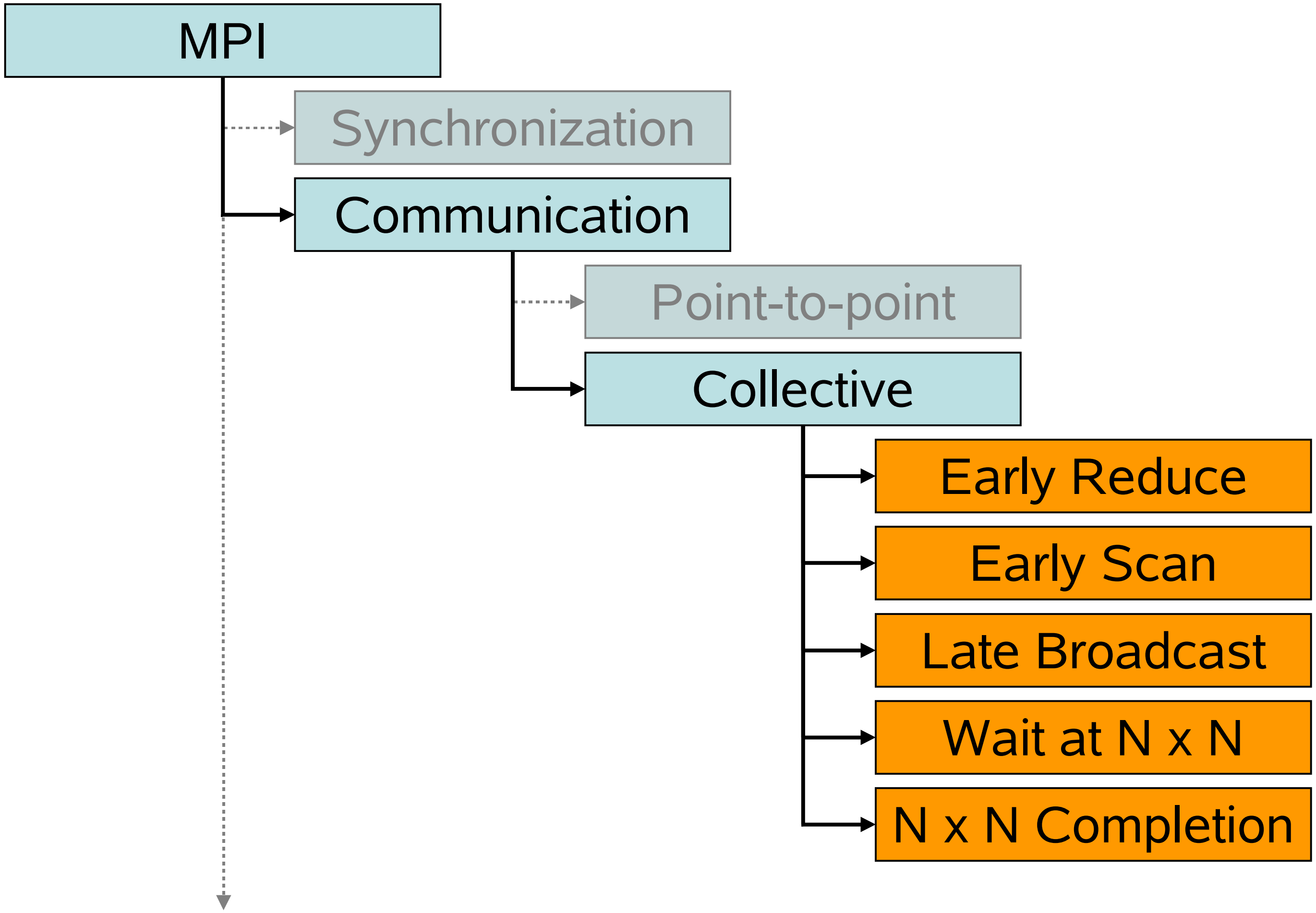
- Time spent waiting in front of a barrier call until the last process reaches the barrier operation
- Applies to: `MPI_Barrier`

Barrier Completion

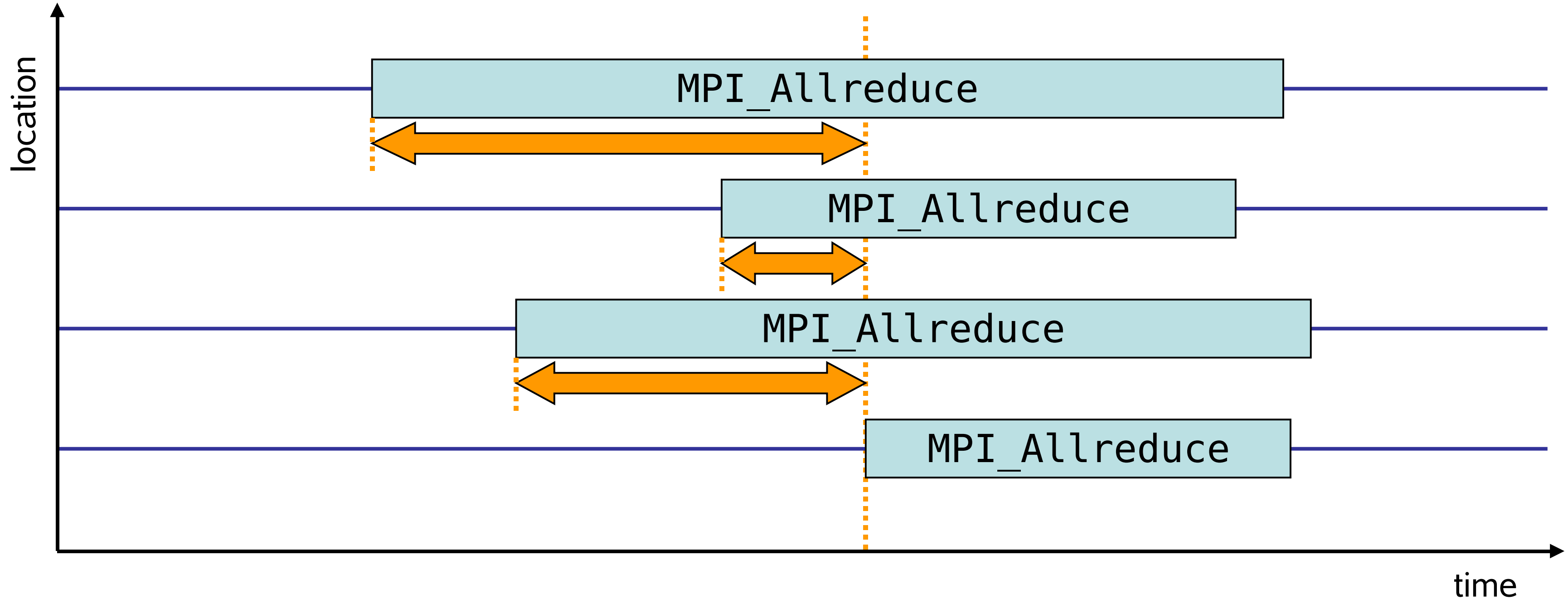


- Time spent in barrier after the first process has left the operation
- Applies to: `MPI_Barrier`

MPI collective communication time

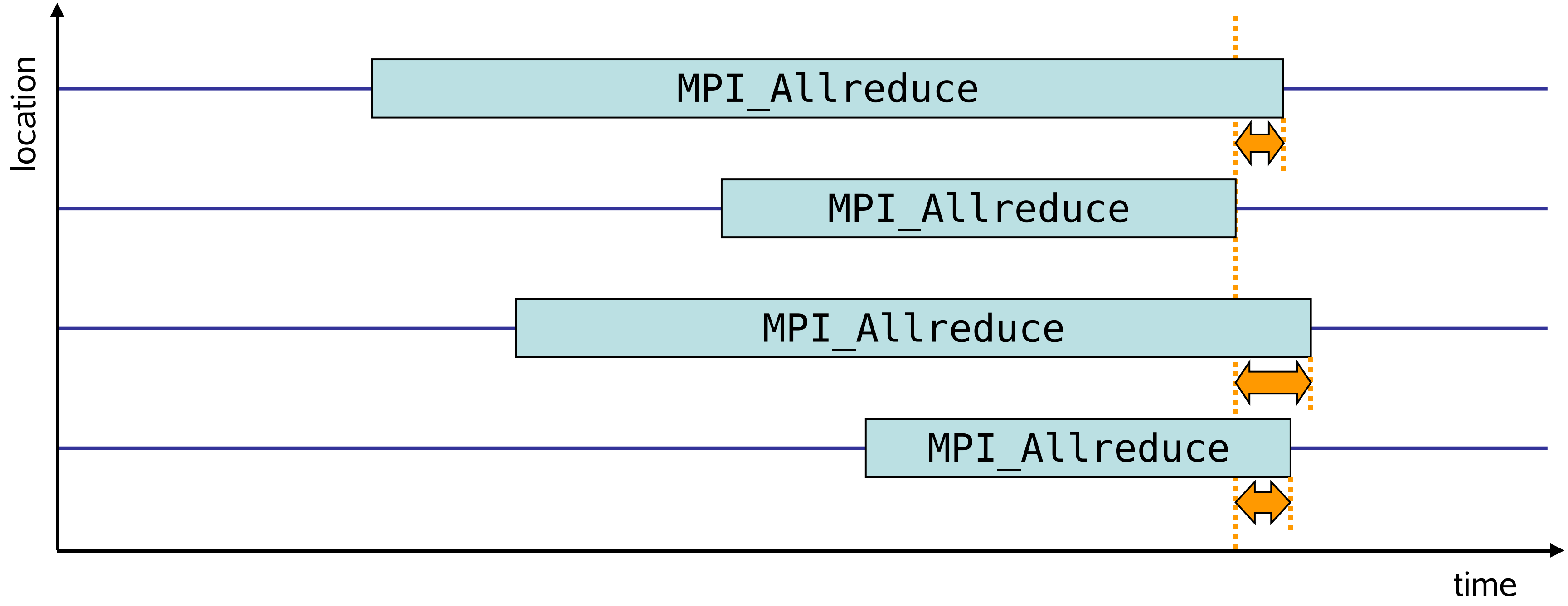


Wait at N x N



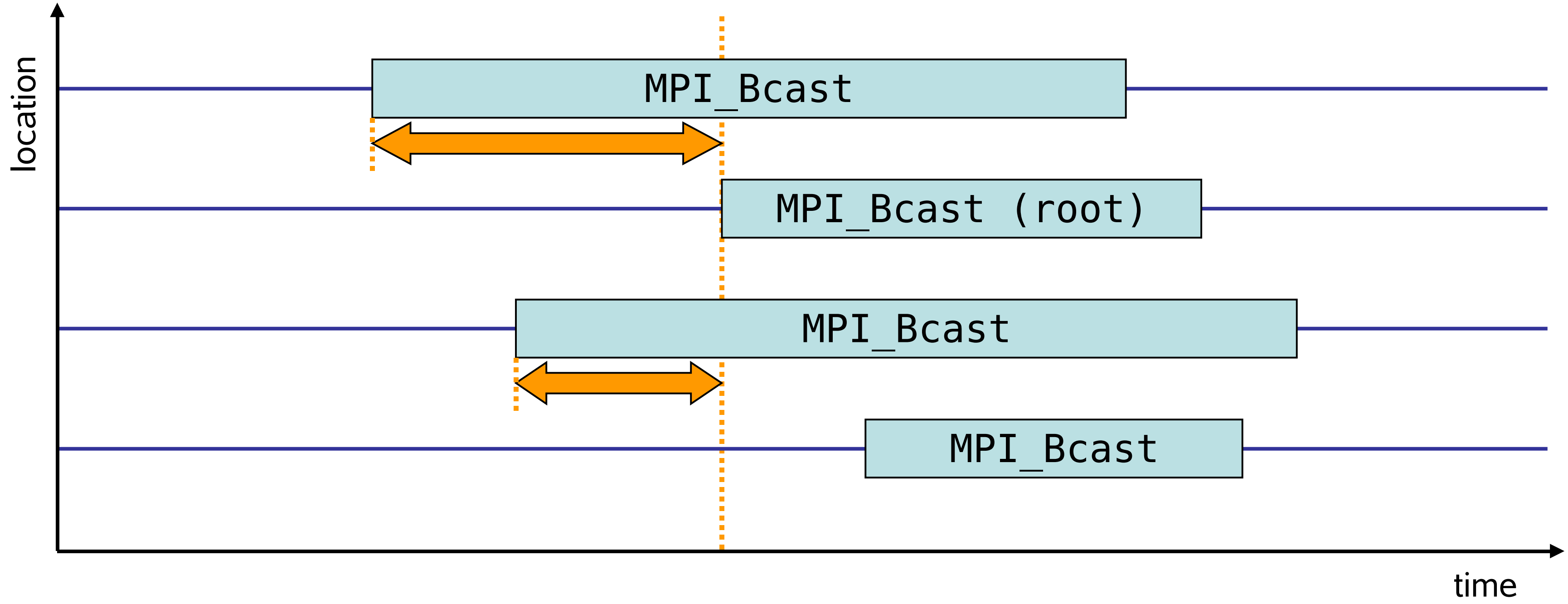
- Time spent waiting in front of a synchronizing collective operation call until the last process reaches the operation
- Applies to: MPI_Allreduce, MPI_Alltoall, MPI_Alltoallv, MPI_Allgather, MPI_Allgatherv, MPI_Reduce_scatter

N x N Completion



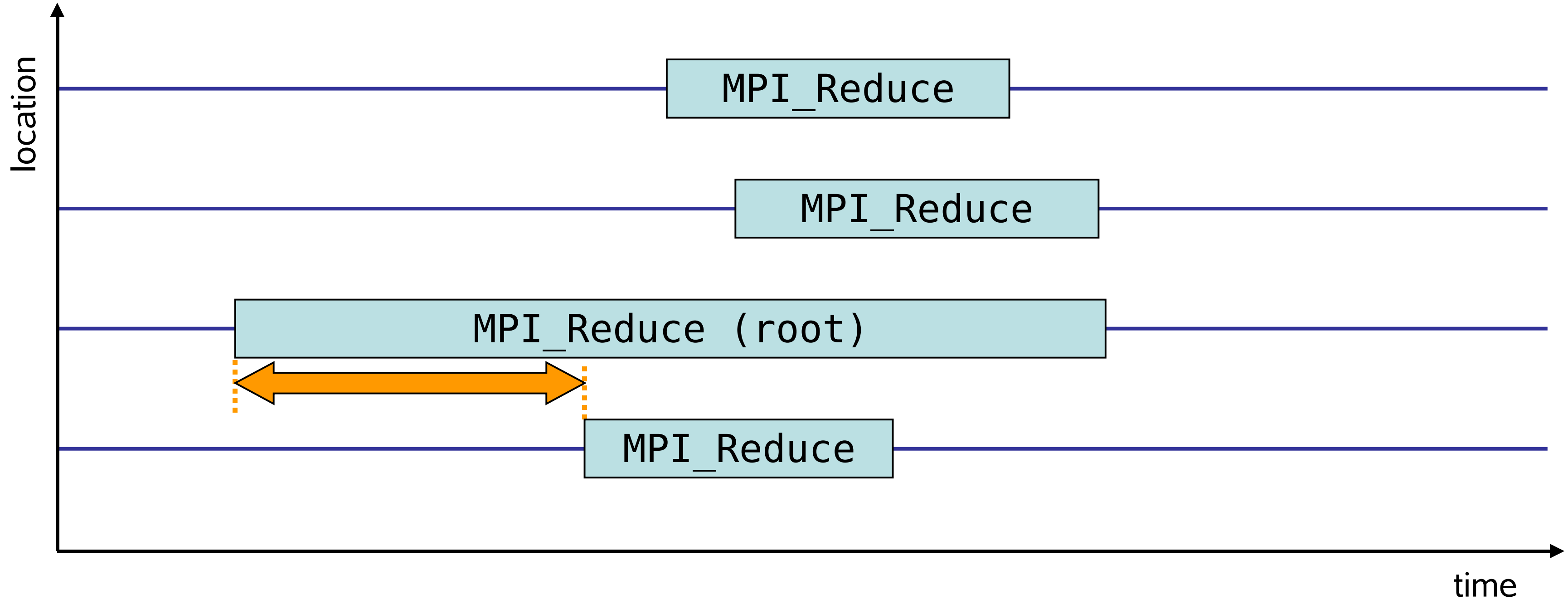
- Time spent in synchronizing collective operations after the first process has left the operation
- Applies to: MPI_Allreduce, MPI_Alltoall, MPI_Alltoallv, MPI_Allgather, MPI_Allgatherv, MPI_Reduce_scatter

Late Broadcast



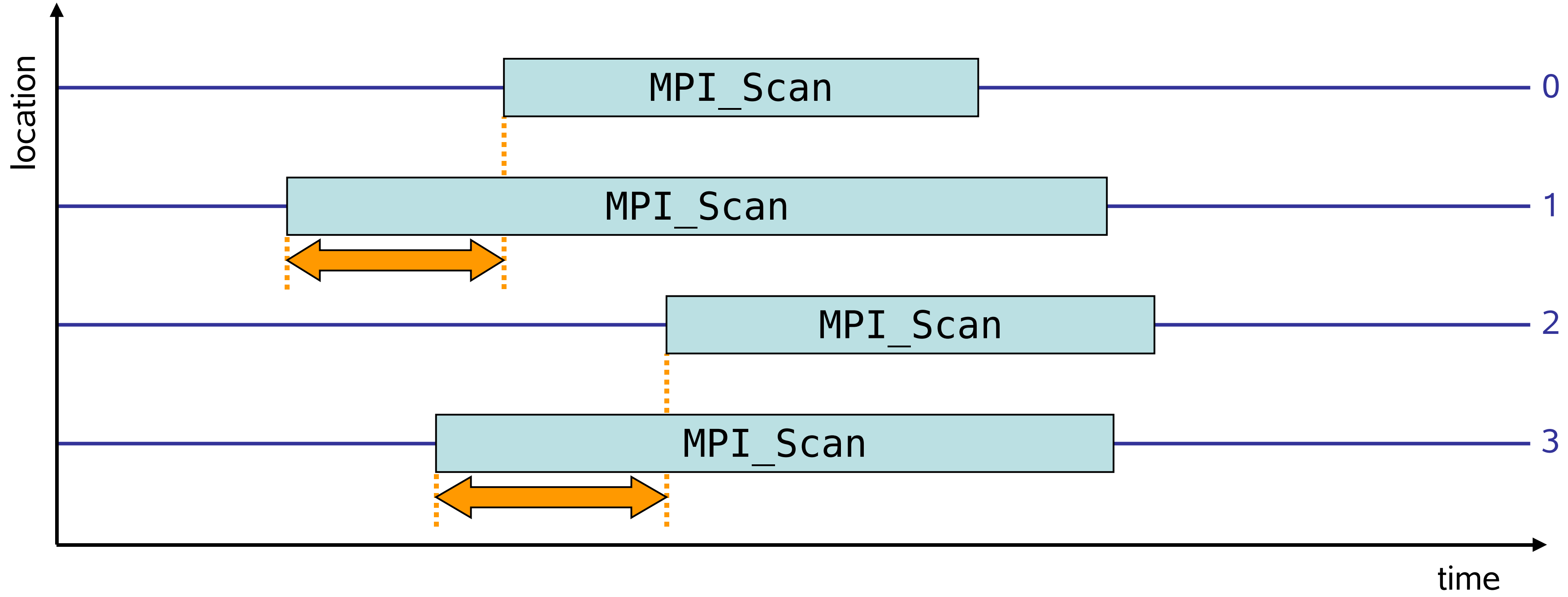
- Waiting times if the destination processes of a collective 1-to-N communication operation enter the operation earlier than the source process (root)
- Applies to: `MPI_Bcast`, `MPI_Scatter`, `MPI_Scatterv`

Early Reduce



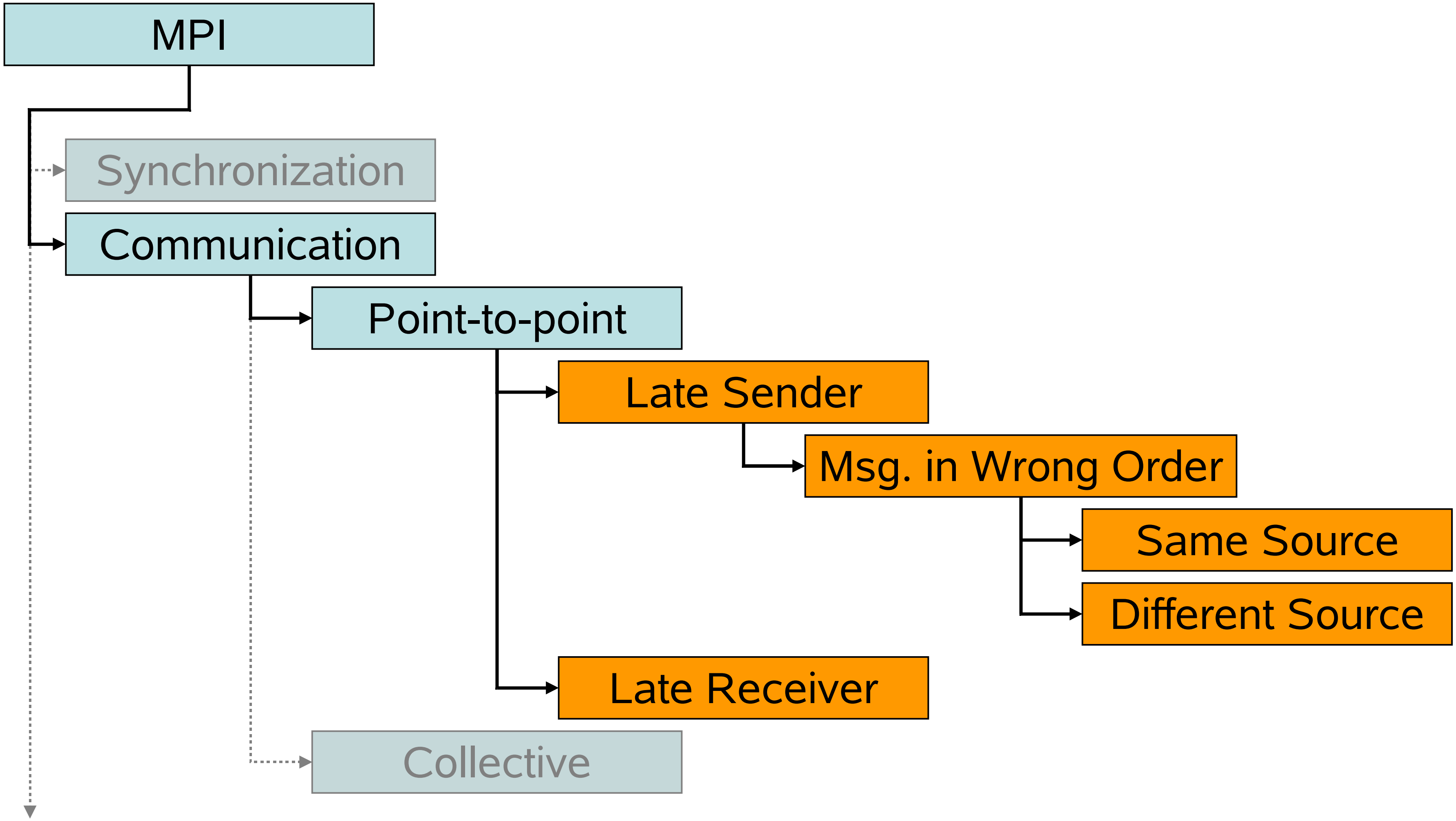
- Waiting time if the destination process (root) of a collective N-to-1 communication operation enters the operation earlier than its sending counterparts
- Applies to: MPI_Reduce, MPI_Gather, MPI_Gatherv

Early Scan

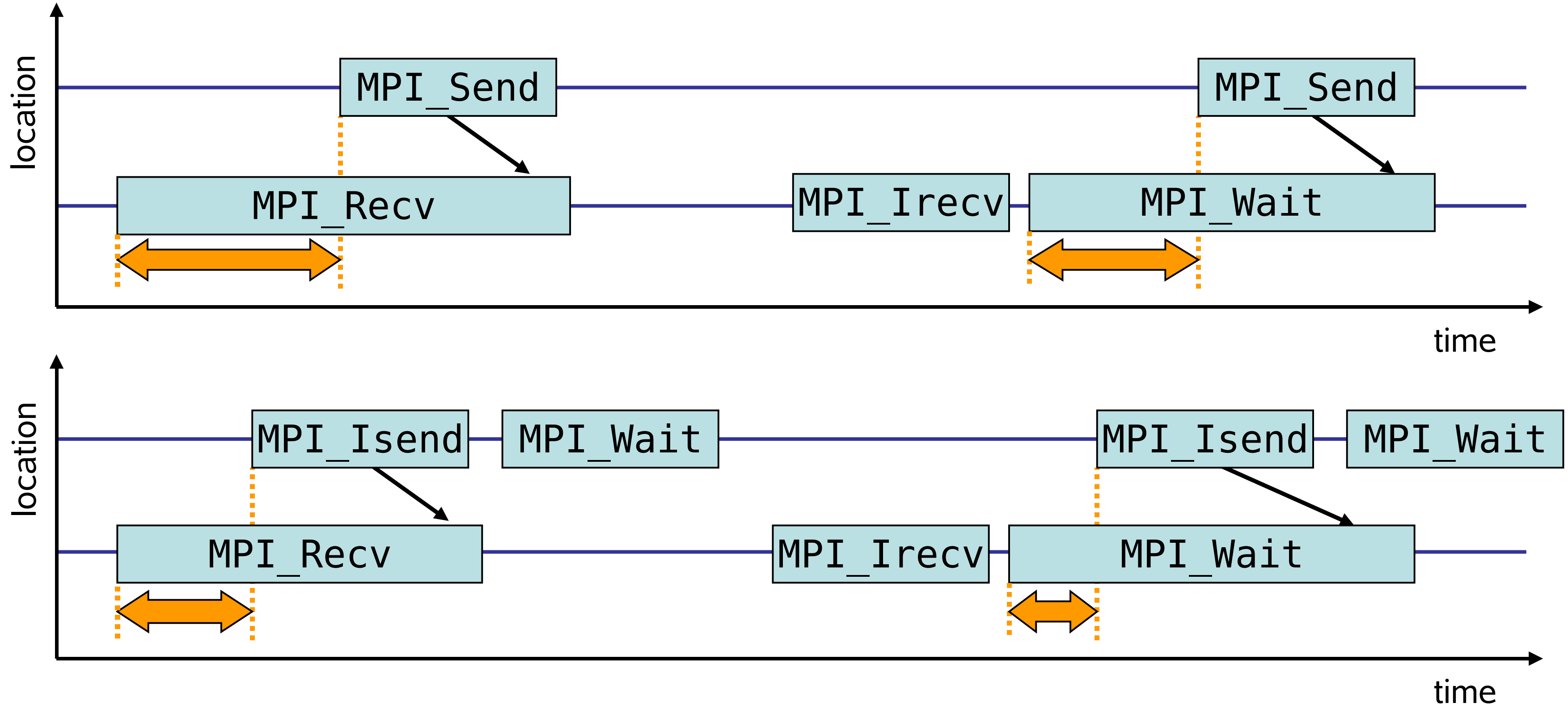


- Waiting time if process *n* enters a prefix reduction operation earlier than its sending counterparts (i.e., ranks 0..*n*-1)
- Applies to: `MPI_Scan`

MPI point-to-point communication time

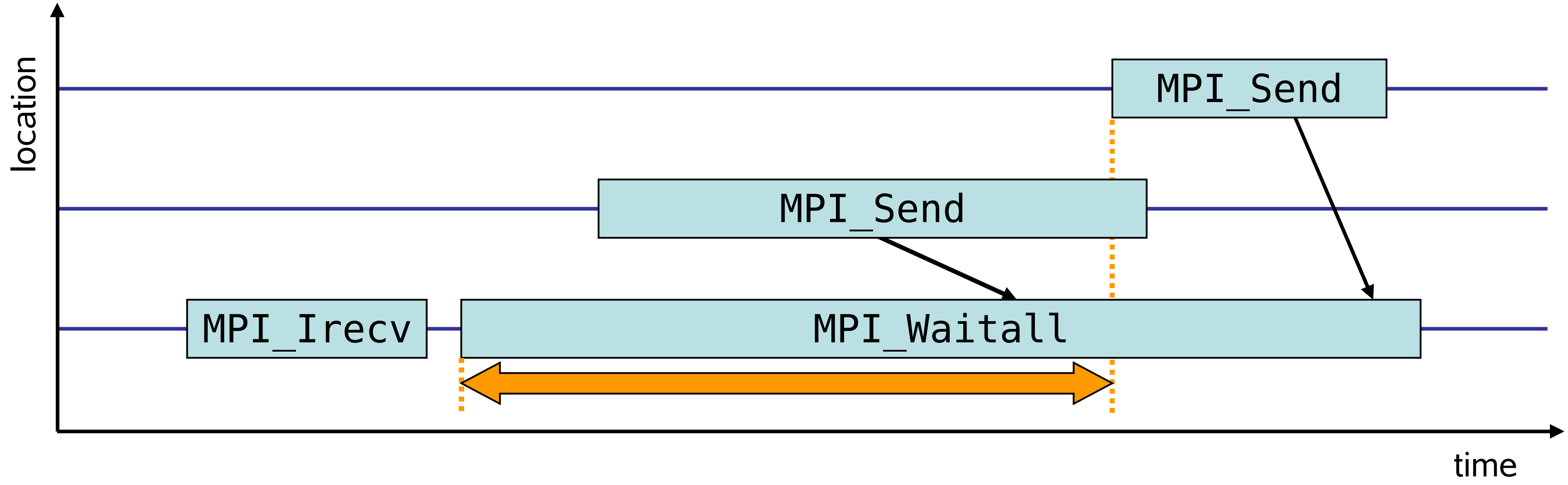


Late Sender



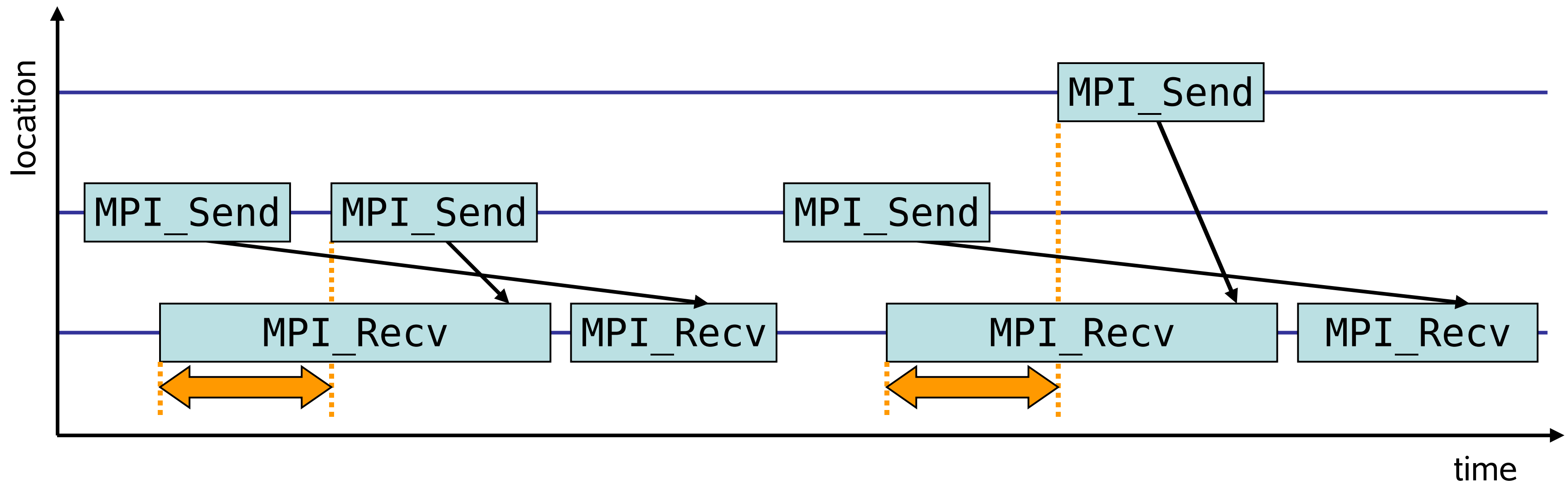
- Waiting time caused by a blocking receive operation posted earlier than the corresponding send operation
- Applies to blocking as well as non-blocking communication

Late Sender (II)



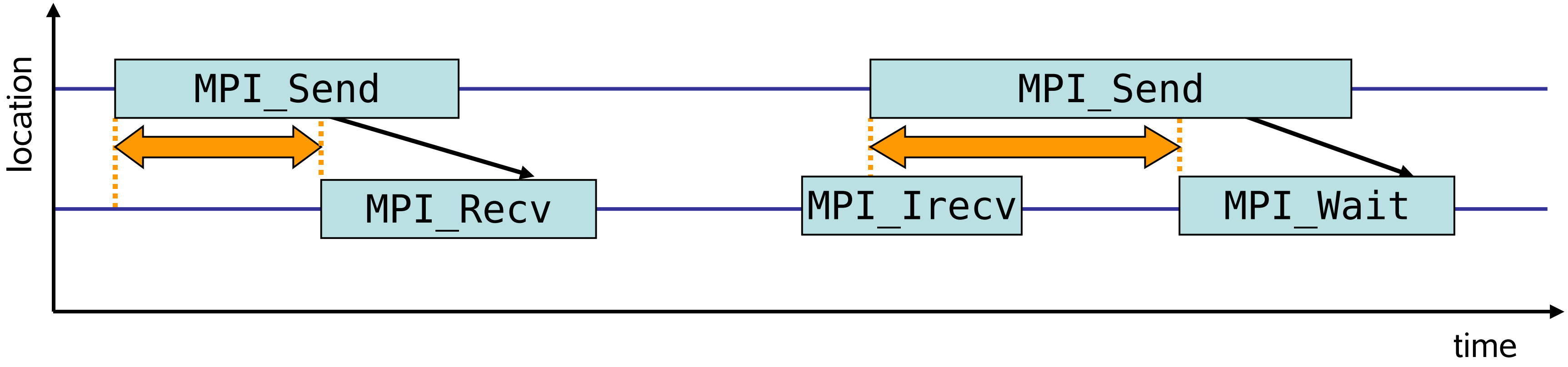
- While waiting for several messages, the maximum waiting time is accounted
- Applies to: `MPI_Waitall`, `MPI_Waitsome`

Late Sender, Messages in Wrong Order



- Refers to Late Sender situations which are caused by messages received in wrong order
- Comes in two flavours:
  - Messages sent from same source location
  - Messages sent from different source locations

Late Receiver



- Waiting time caused by a blocking send operation posted earlier than the corresponding receive operation
- Calculated by receiver but waiting time attributed to sender
- Does currently not apply to non-blocking sends

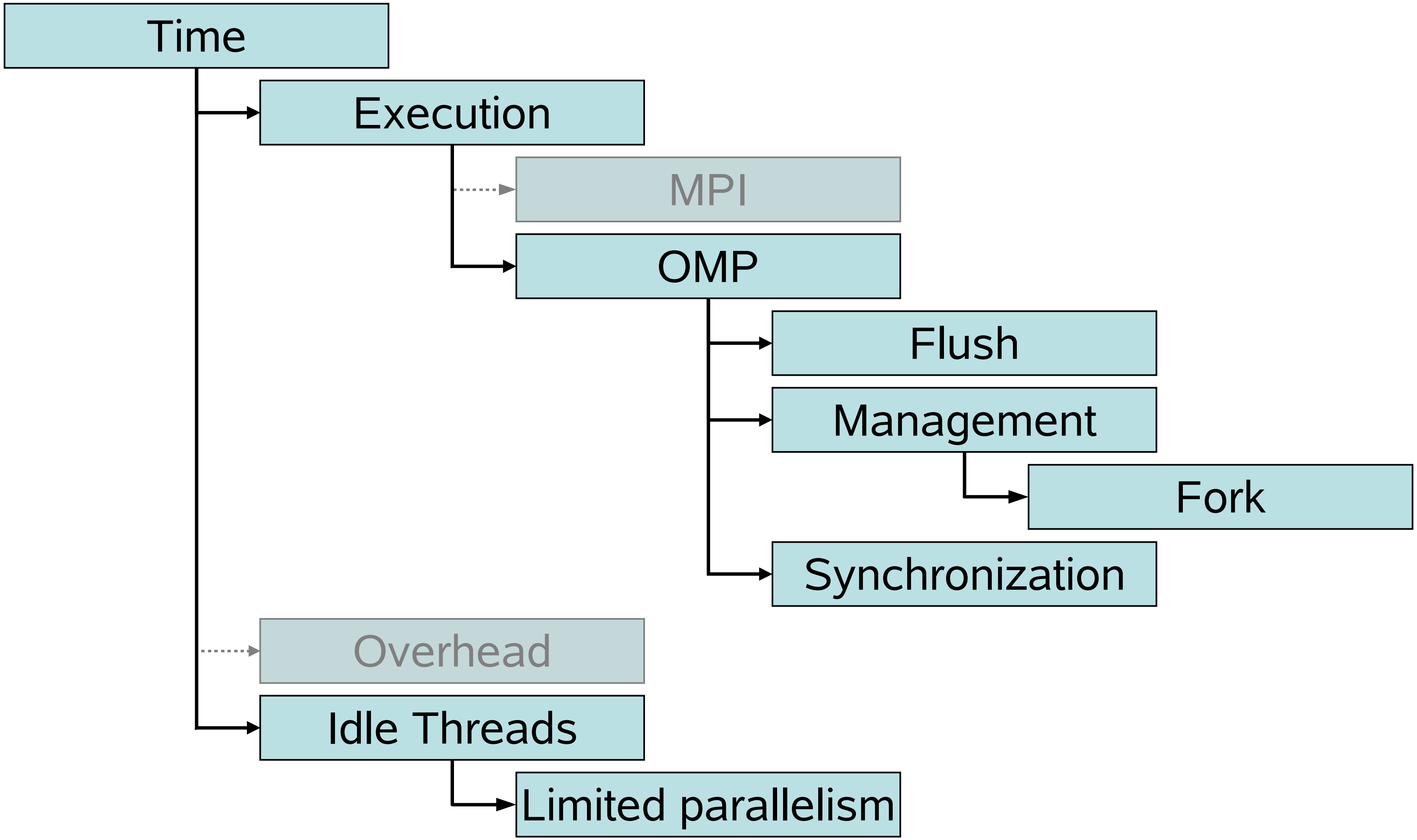
Late Sender/Receiver Counts

- The number of Late Sender / Late Receiver instances are also available
- They are divided into communications & synchronizations and shown in the corresponding hierarchies

OpenMP-related metrics

(as produced by Scalasca 1.2 runtime summarization
and trace analysis for hybrid MPI/OpenMP apps)

OpenMP Time hierarchy



Time

Execution

MPI

OMP

Flush

Management

Fork

Synchronization

Overhead

Idle Threads

Limited parallelism

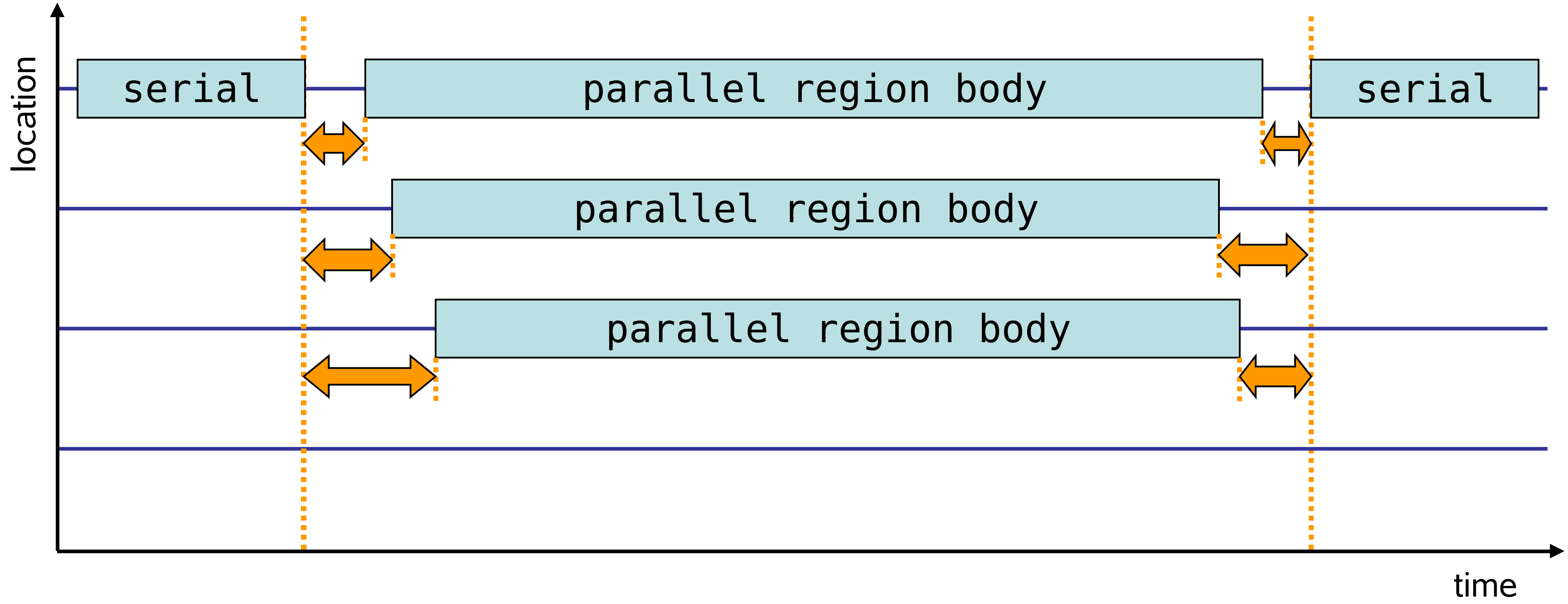
OpenMP Time hierarchy details

OMP — Time spent for OpenMP-related tasks

Flush — Time spent in OpenMP flush directives

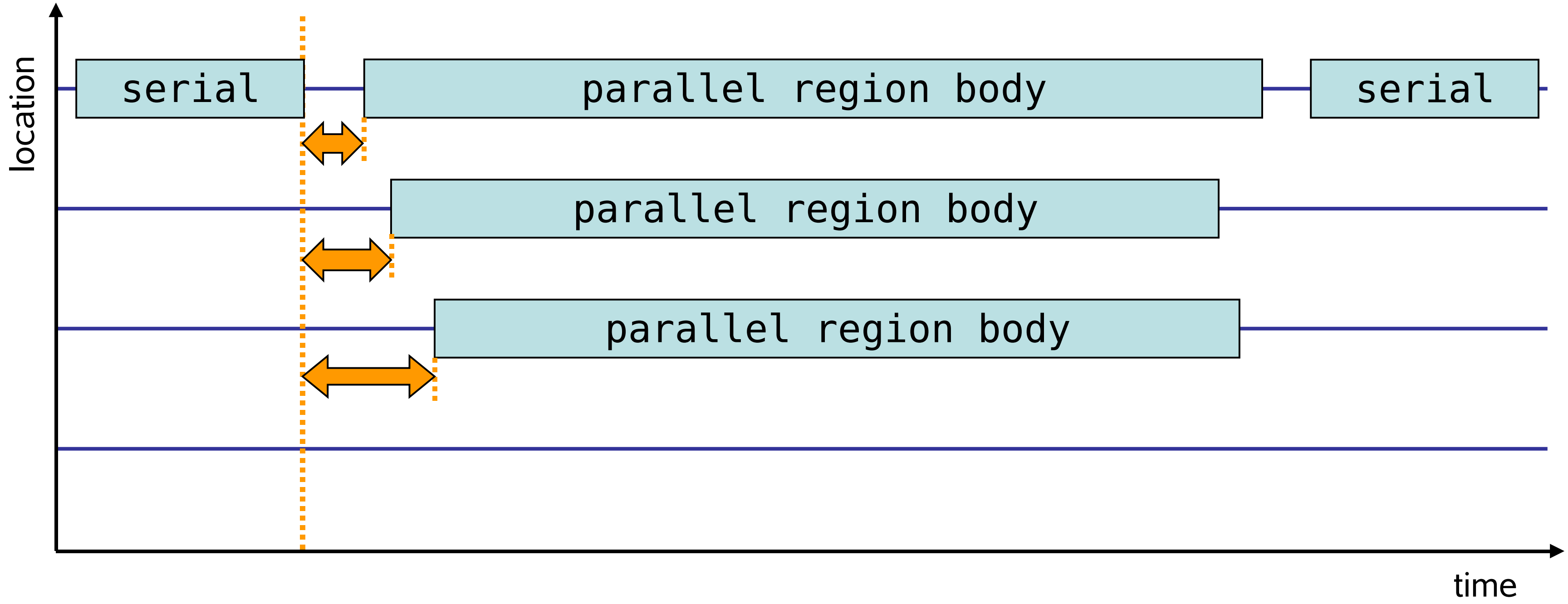
Synchronization — Time spent to synchronize OpenMP threads

OpenMP Management Time



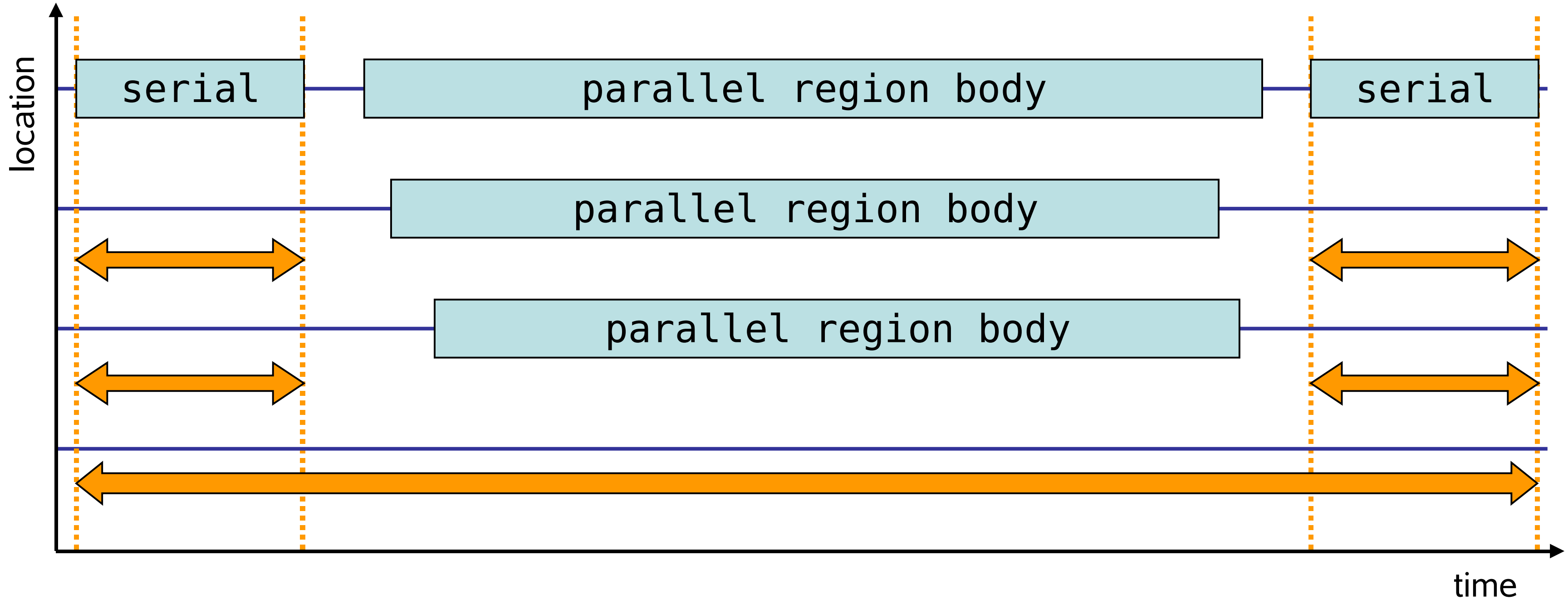
- Time spent on master thread for creating/destroying OpenMP thread teams

OpenMP Fork Time



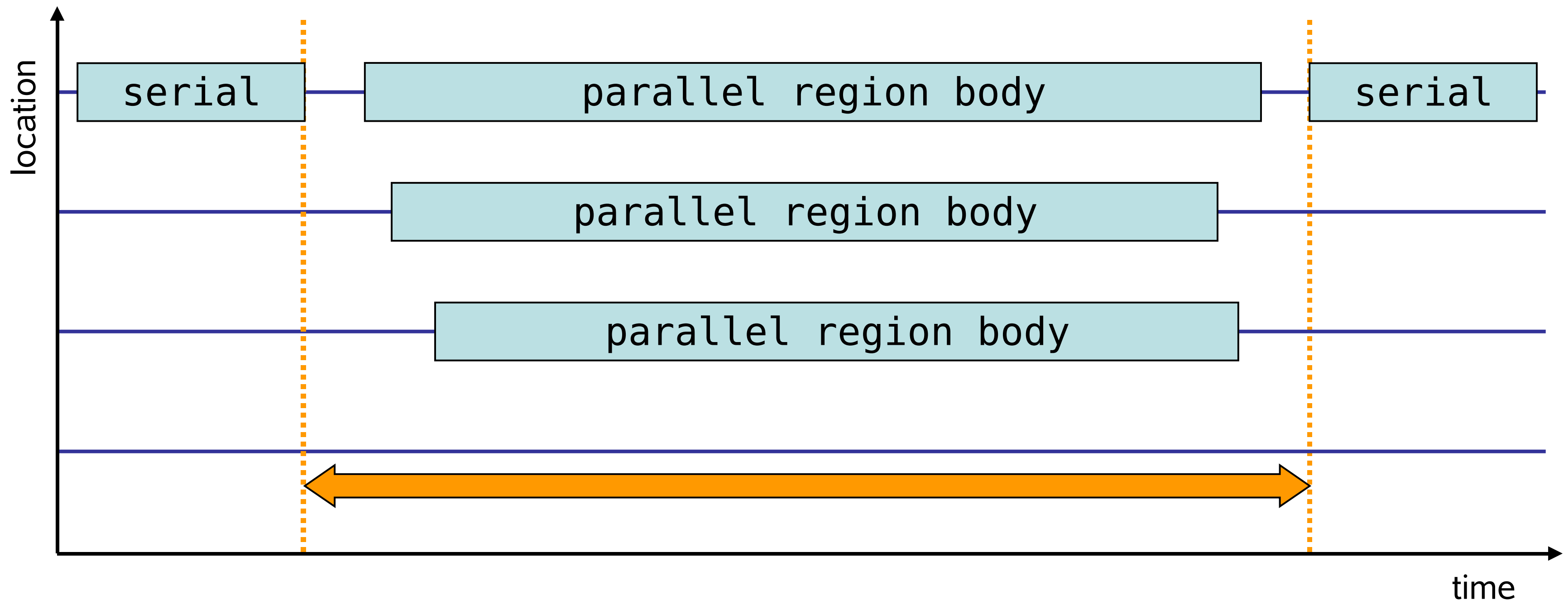
- Time spent on master threads for creating OpenMP thread teams

OpenMP Idle Threads



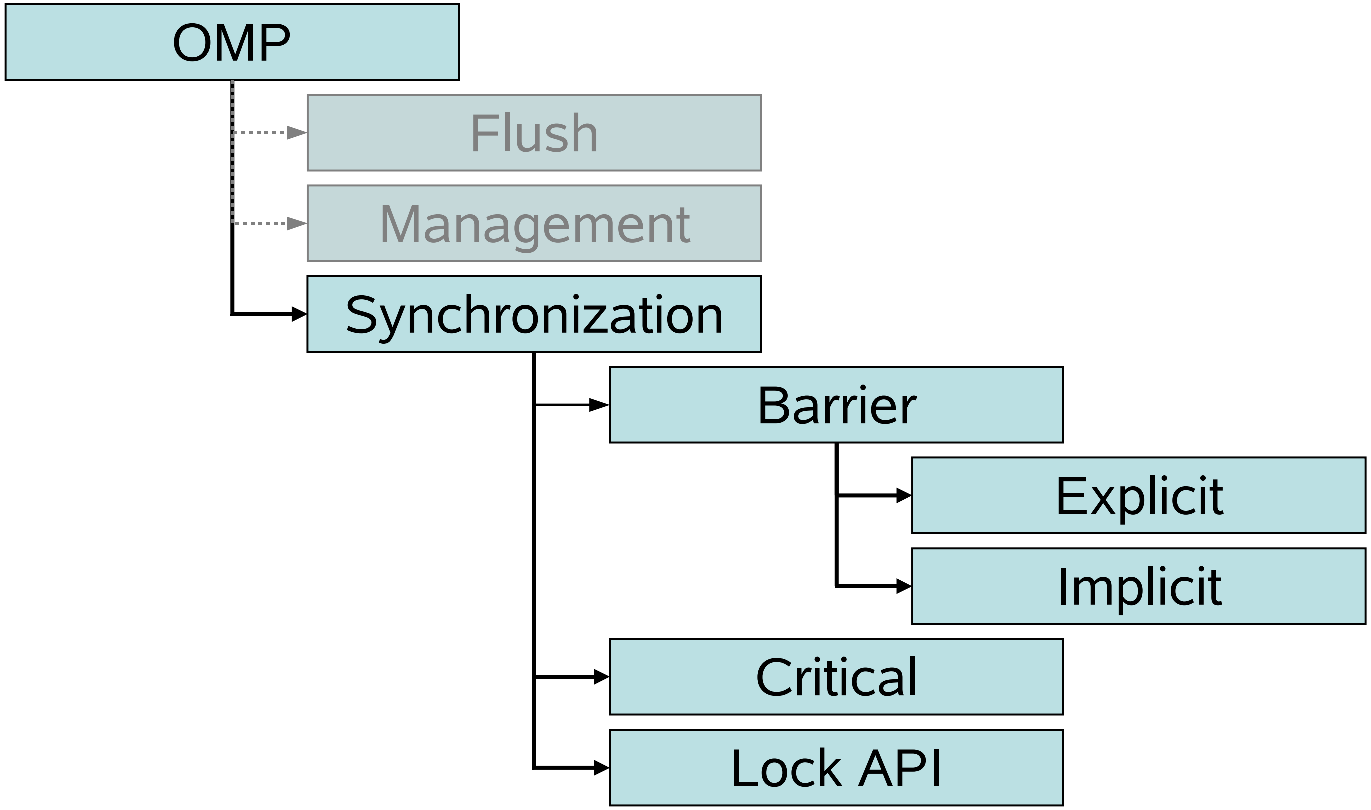
- Time spent idle on CPUs reserved for worker threads

OpenMP Limited Parallelism



- Time spent idle on worker threads within parallel regions

OpenMP Synchronization Time hierarchy

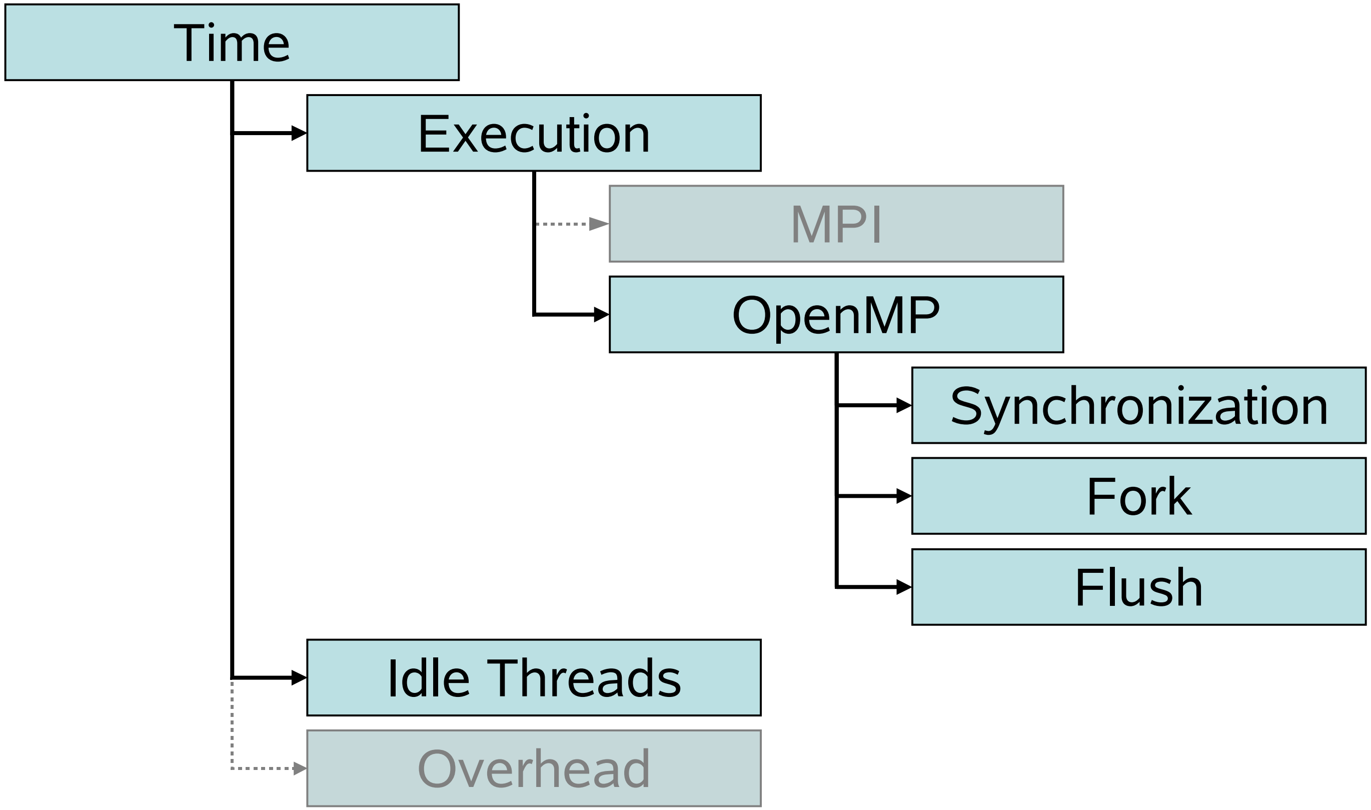


- Time spent in OpenMP atomic constructs is attributed to the “Critical” metric

OpenMP-related metrics

(as produced by Scalasca 1.2 trace analysis
for pure OpenMP apps)

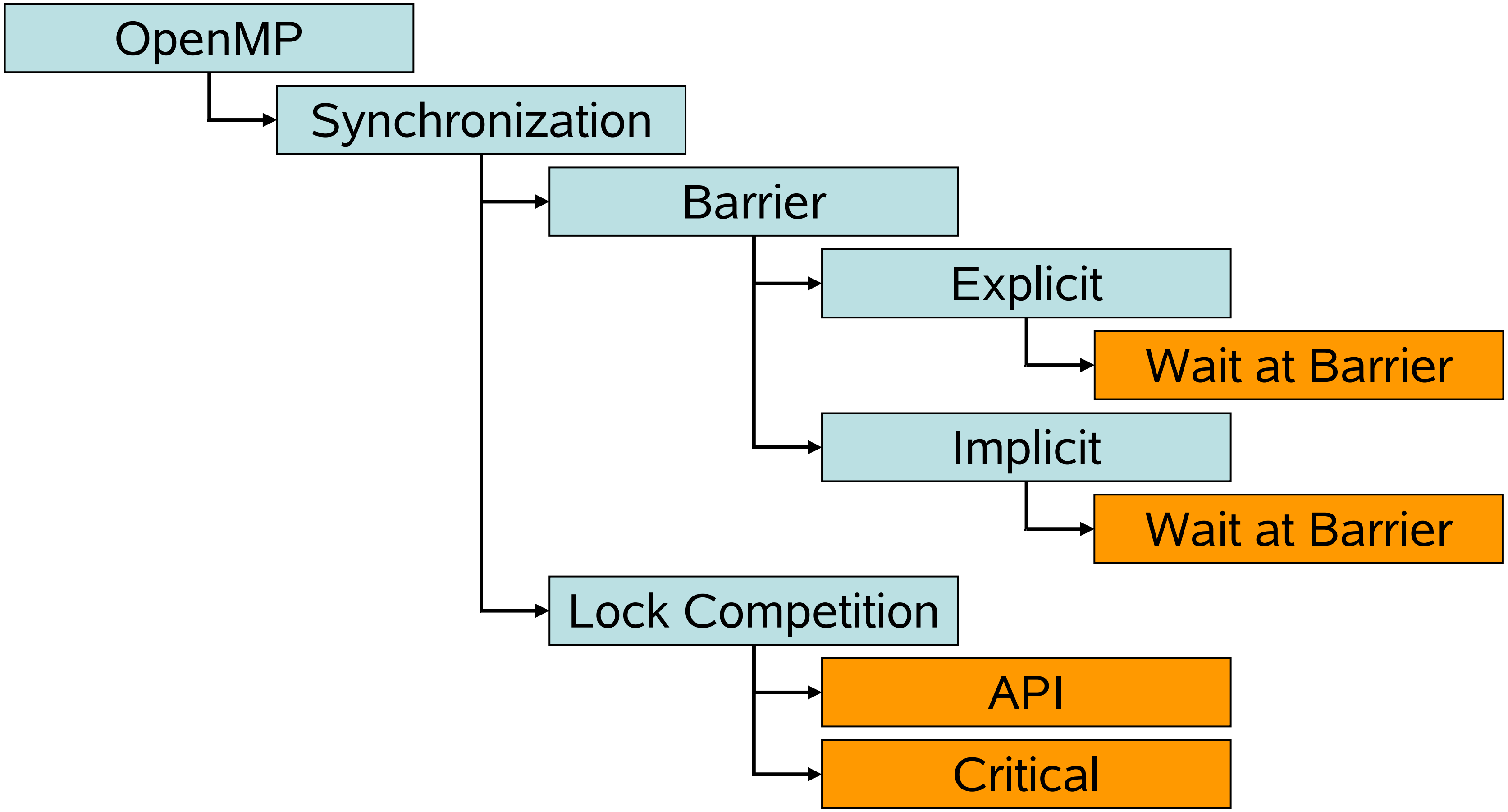
OpenMP Time hierarchy


Time
Execution
MPI
OpenMP
Synchronization
Fork
Flush
Idle Threads
Overhead

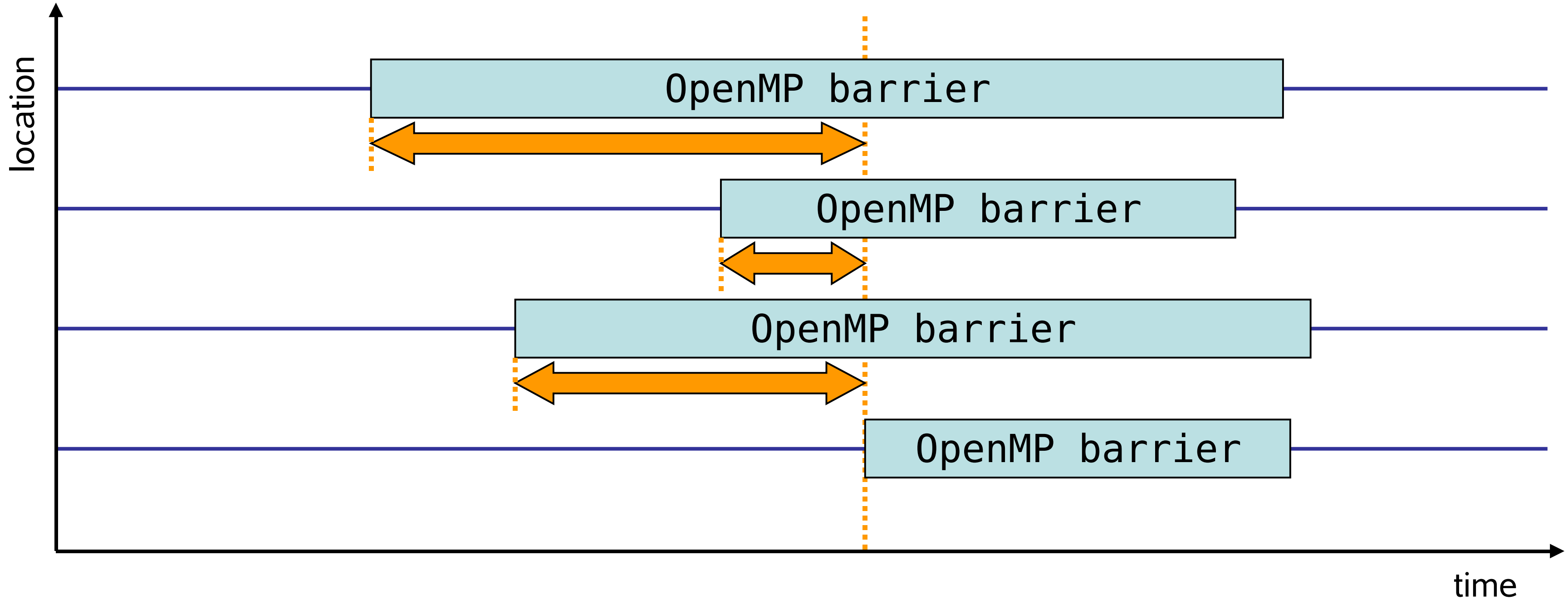
OpenMP Time hierarchy details

OpenMP	Time spent for OpenMP-related tasks
Synchronization	Time spent for synchronizing OpenMP threads
Fork	Time spent by master thread to create thread teams
Flush	Time spent in OpenMP flush directives
Idle Threads	Time spent idle on CPUs reserved for worker threads

OpenMP synchronization time

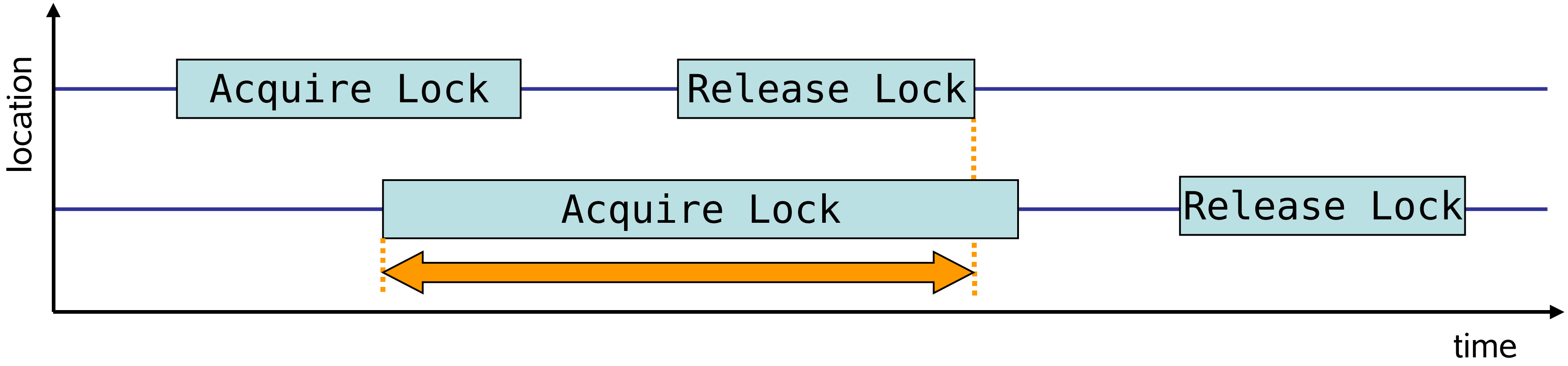


Wait at Barrier



- Time spent waiting in front of a barrier call until the last process reaches the barrier operation
- Applies to: Implicit/explicit barriers

Lock Competition



- Time spent waiting for a lock that has been previously acquired by another thread
- Applies to: critical sections, OpenMP lock API