

# Parallel Programming

Prof. **Paolo Bientinesi**

`pauldj@aices.rwth-aachen.de`

WS 17/18



```
MPI_Comm_split(  
    MPI_Comm comm,           <- EVERYBODY involved!  
    int color,               <- where do I belong  
    int key,                 <- for the new rank  
    MPI_Comm* newcomm)
```

```
MPI_Comm_split(  
    MPI_Comm comm,           <- EVERYBODY involved!  
    int color,               <- where do I belong  
    int key,                 <- for the new rank  
    MPI_Comm* newcomm)
```

```
MPI_Comm_dup  
MPI_Comm_create  
MPI_Comm_create_group  
MPI_Group_incl  
...  
MPI_Group_union, MPI_Group_intersection
```

## Cartesian topology

```
MPI_Cart_create(  
    old_communicator,  
    ndims,           <- number of dimensions of the grid  
    dims[],         <- size of each dimension  
    periods[],     <- periodic boundaries?  
    reorder,  
    *cart_communicator <- new communicator  
)
```

# Topologies: Who are my neighbors?

---

## Cartesian topology

```
MPI_Cart_create(  
    old_communicator,  
    ndims,                <- number of dimensions of the grid  
    dims[],               <- size of each dimension  
    periods[],           <- periodic boundaries?  
    reorder,  
    *cart_communicator   <- new communicator  
)
```

## Who am I?

```
MPI_Comm_rank( cart_COMM, &rank );  
MPI_Cart_coords( cart_COMM, rank, ndims, coords[] );
```

# Topologies: Who are my neighbors?

---

## Cartesian topology

```
MPI_Cart_create(  
    old_communicator,  
    ndims,           <- number of dimensions of the grid  
    dims[],         <- size of each dimension  
    periods[],      <- periodic boundaries?  
    reorder,  
    *cart_communicator <- new communicator  
)
```

## Who am I?

```
MPI_Comm_rank( cart_COMM, &rank );  
MPI_Cart_coords( cart_COMM, rank, ndims, coords[] );
```

## My neighbors?      No neighbor: `MPI_PROC_NULL`

```
MPI_Cart_shift( cart_COMM, direction, displacement, *rank_source, *rank_dest )
```

# Topologies: Who are my neighbors?

---

## Cartesian topology

```
MPI_Cart_create(  
    old_communicator,  
    ndims,                <- number of dimensions of the grid  
    dims[],               <- size of each dimension  
    periods[],           <- periodic boundaries?  
    reorder,  
    *cart_communicator  <- new communicator  
)
```

## Who am I?

```
MPI_Comm_rank( cart_COMM, &rank );  
MPI_Cart_coords( cart_COMM, rank, ndims, coords[] );
```

## My neighbors?          No neighbor: `MPI_PROC_NULL`

```
MPI_Cart_shift( cart_COMM, direction, displacement, *rank_source, *rank_dest )
```

## Other topologies: `MPI_Graph_create`

# 1-sided communication

---

Communication happens without the agreement of both sides!

- `MPI_Put`  
write into target's memory
- `MPI_Get`  
read from target's memory
- `MPI_Win_create`, `MPI_Win_start`, `MPI_Win_complete`, ...  
define & manage memory space accessible from other processes