Master in High Performance Computing

More on // Programming

Ivan Girotto – igirotto@ictp.it
Information & Communication Technology Section (ICTS)
International Centre for Theoretical Physics (ICTP)

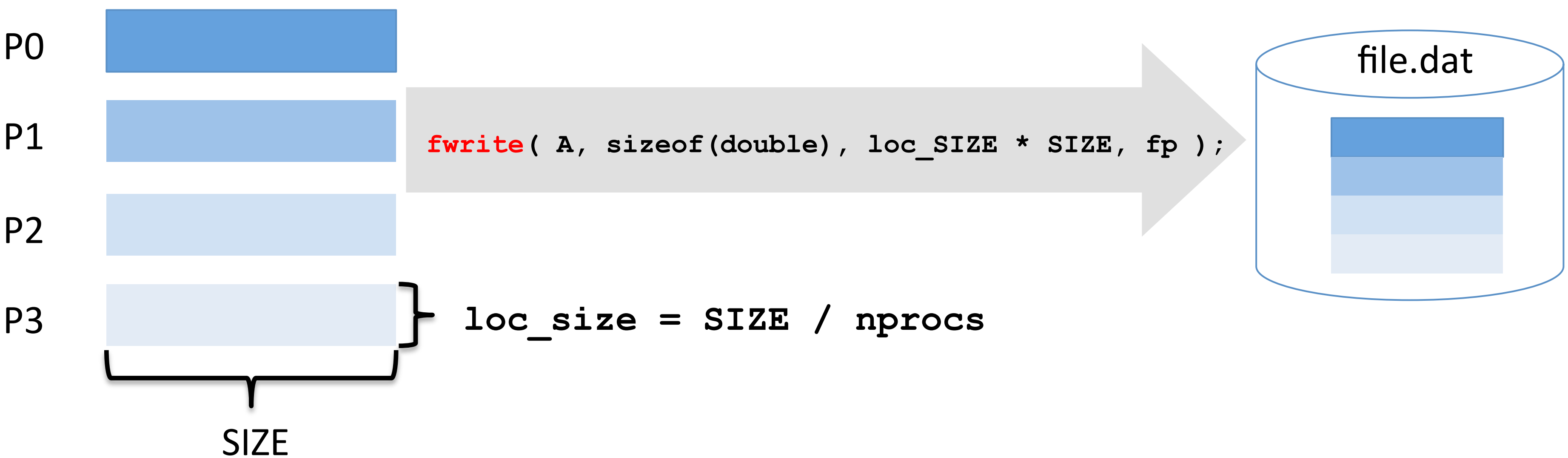
Scuola Internazionale Superiore di Studi Avanzati

How to write distributed data on a FILE?

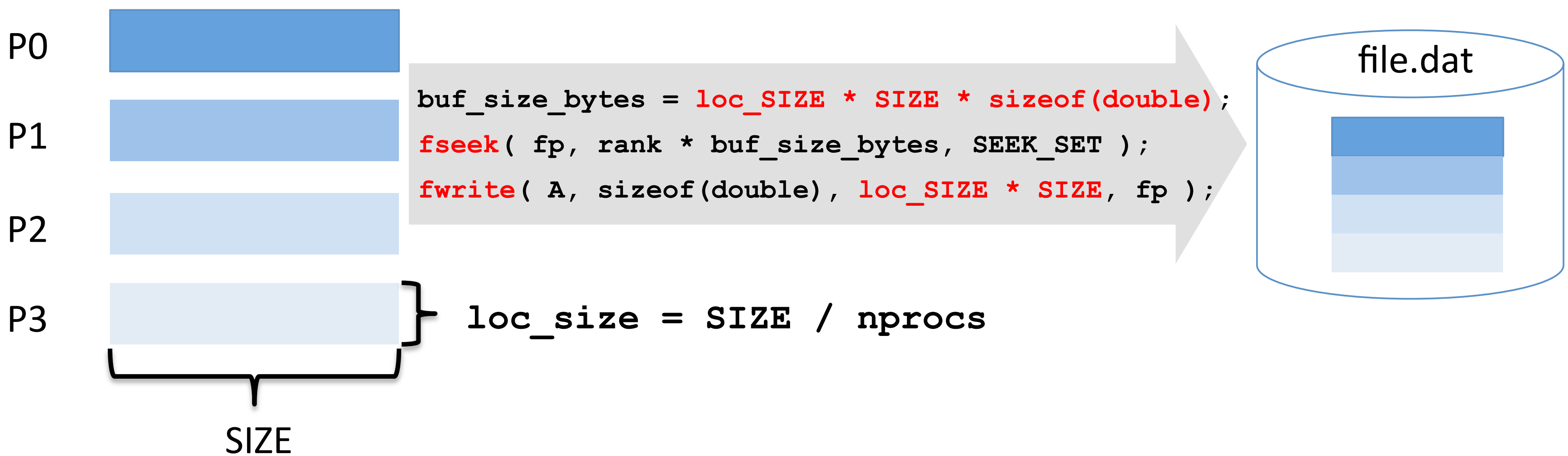
- It requires a serialized access from a single process in case of a text file (sequential access)
- It can be performed in parallel in case of binary files (random access)
- Common sequence for creating and writing a binary file:

```
int main( int argc, char * argv[] ){

  double * A;
  int i = 0;
  FILE * fp;

  A = (double *) malloc( SIZE * SIZE * sizeof(double) );

  for( i = 0; i < SIZE * SIZE; i++ ){

    A[i] = (double) ( rand() % 1000 + 1 );
  }

  fp = fopen( "matrix.dat", ”w" );
  fwrite( A, sizeof(double), SIZE * SIZE, fp );
  fclose( fp );

  free( A );

  return 0;
}
```

How to write distributed data on a FILE?



How to write distributed data on a FILE?



The fseek() function sets the file position indicator for the stream pointed to by stream. The new position, measured in bytes, is obtained by adding offset bytes to the position specified by whence. If whence is set to SEEK_SET, SEEK_CUR, or SEEK_END, the offset is relative to the start of the file, the current position indicator, or end-of-file, respectively.

// Matrix-Matrix Multiplication

```
int MPI_Allgather(const void *sendbuf, int sendcount, MPI_Datatype sendtype,
void *recvbuf, int recvcount, MPI_Datatype recvtype, MPI_Comm comm)
```

sendbuf starting address of send buffer (choice)

sendcount number of elements in send buffer (integer)

sendtype data type of send buffer elements (handle)

recvcount number of elements received from any process (integer)

recvtype data type of receive buffer elements (handle)

comm communicator (handle)

1. Distribute the Matrix
2. Initialize the Distributed Matrix
3. At every time step use MPI_Allgather to send at all processes a block of column of B
4. Repeat point 3 for all blocks of column of B
5. Parallel Print the Matrix C
6. Analyse the performance scaling of your Parallel GEMM on multiple nodes
7. Replace you mat-mul with the BLAS “cblas_dgemm” interface

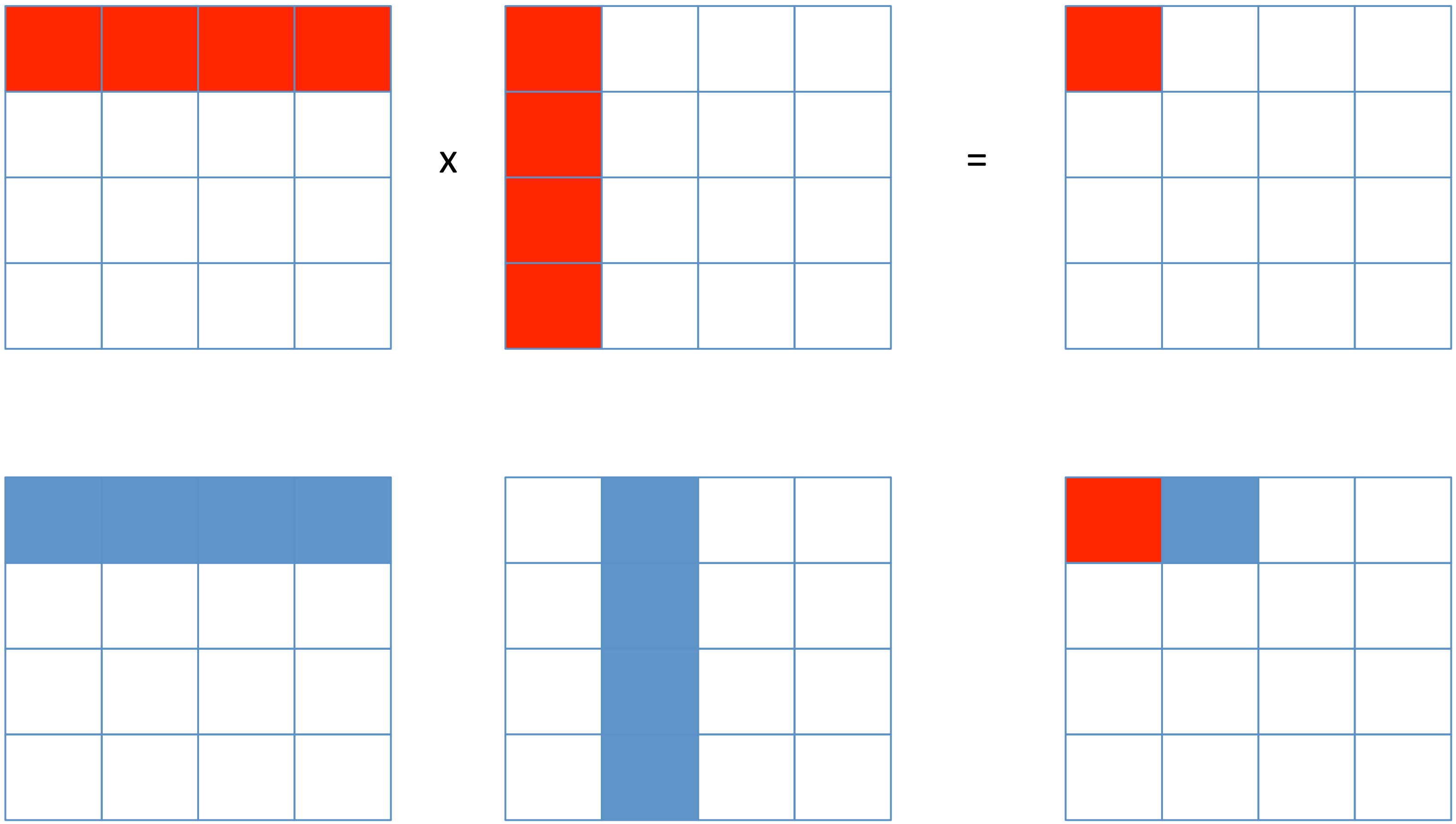
```
cblas_dgemm(CblasRowMajor, CblasNoTrans, CblasNoTrans,
            m, n, k, alpha, A, k, B, n, beta, C, n);
```

* from Intel MKL documentation

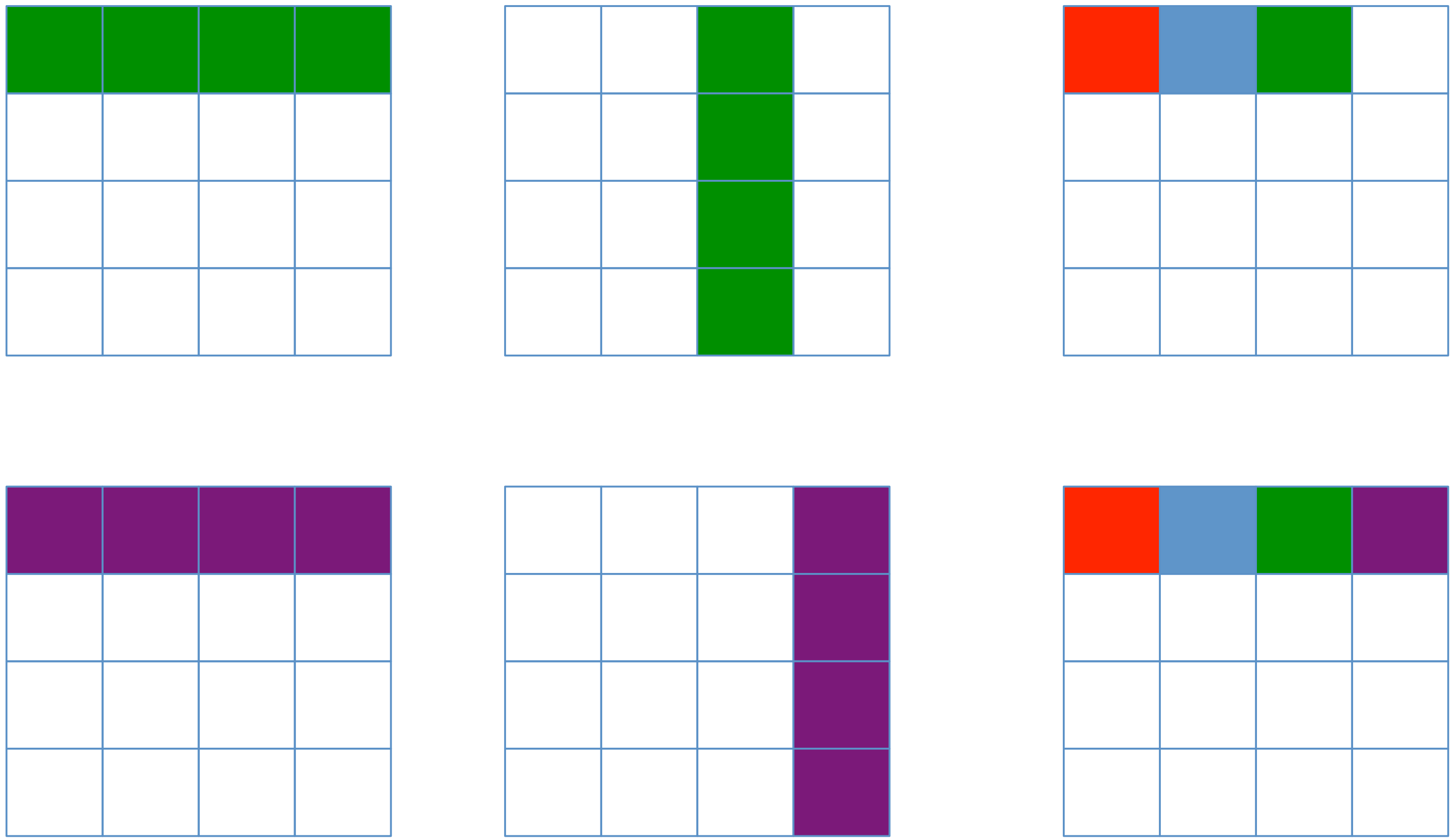
The arguments provide options for how Intel MKL performs the operation. In this case:

`CblasRowMajor`	Indicates that the matrices are stored in row major order, with the elements of each row of the matrix stored contiguously as shown in the figure above.
`CblasNoTrans`	Enumeration type indicating that the matrices *A* and *B* should not be transposed or conjugate transposed before multiplication.
`m, n, k`	Integers indicating the size of the matrices: • *A*: `m` rows by `k` columns • *B*: `k` rows by `n` columns • *C*: `m` rows by `n` columns
`alpha`	Real value used to scale the product of matrices *A* and *B*.
`A`	Array used to store matrix *A*.
`k`	Leading dimension of array `A`, or the number of elements between successive rows (for row major storage) in memory. In the case of this exercise the leading dimension is the same as the number of columns.
`B`	Array used to store matrix *B*.
`n`	Leading dimension of array `B`, or the number of elements between successive rows (for row major storage) in memory. In the case of this exercise the leading dimension is the same as the number of columns.
`beta`	Real value used to scale matrix *C*.
`C`	Array used to store matrix *C*.
`n`	Leading dimension of array `C`, or the number of elements between successive rows (for row major storage) in memory. In the case of this exercise the leading dimension is the same as the number of columns.

See the following link for complete documentation:
https://software.intel.com/en-us/mkl-developer-reference-c-cblas-gemm

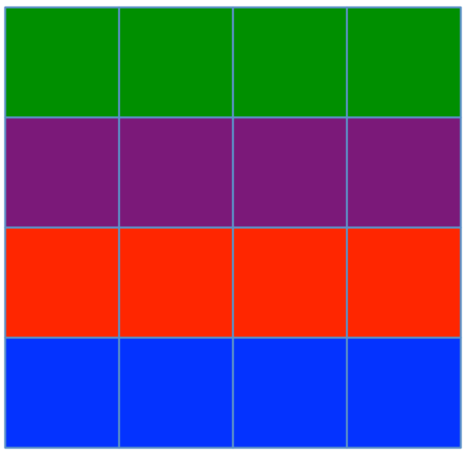


MHPC
Master in High Performance Computing

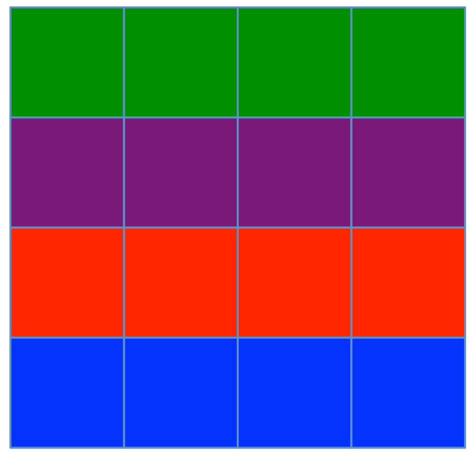


SISSA
Scuola Internazionale Superiore di Studi Avanzati

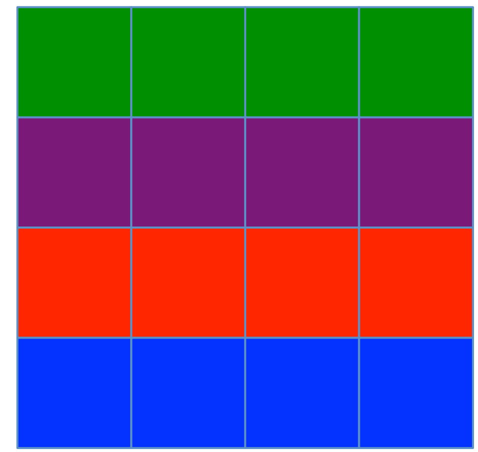
ICTP
The Abdus Salam International Centre for Theoretical Physics



A

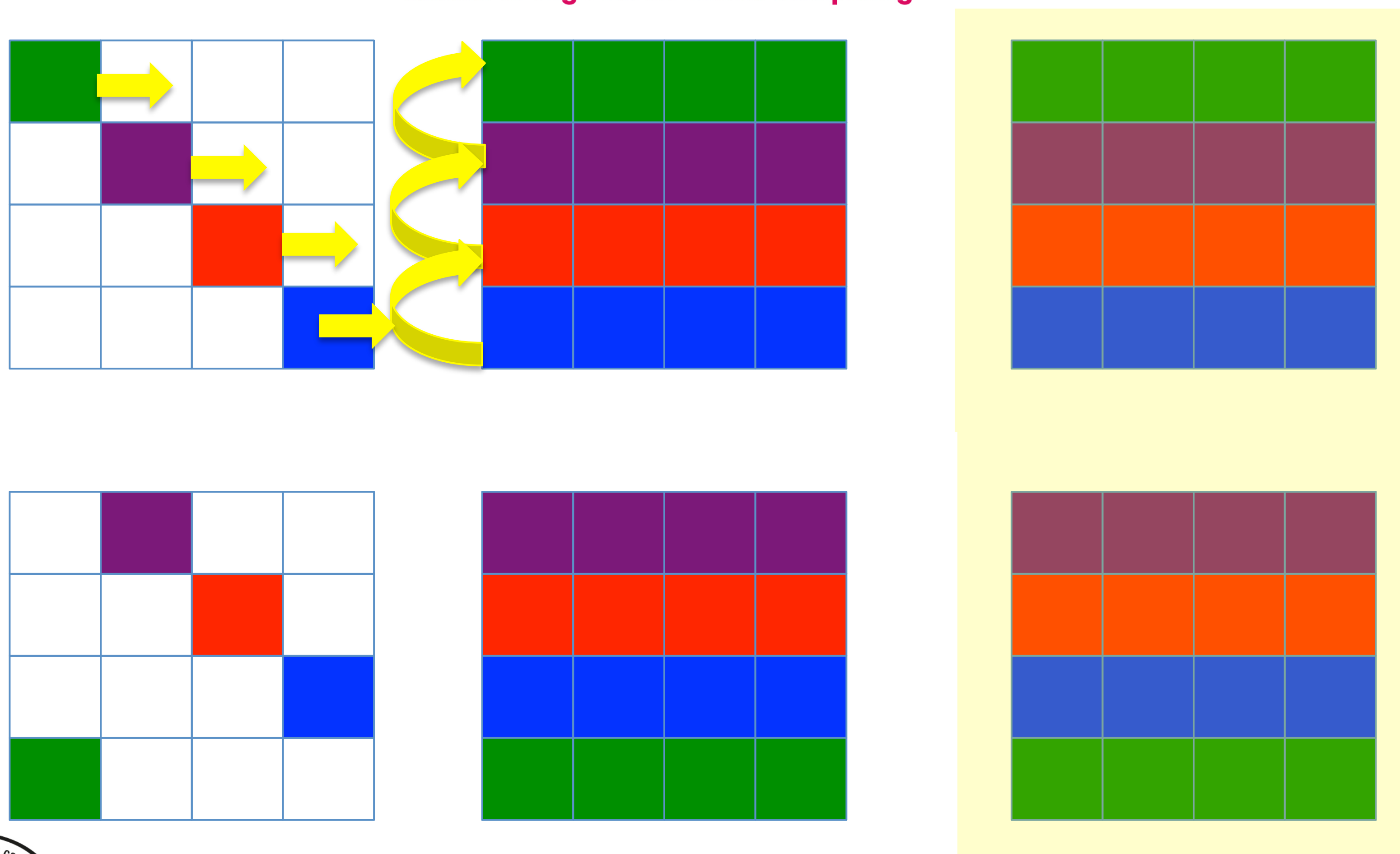


B



C

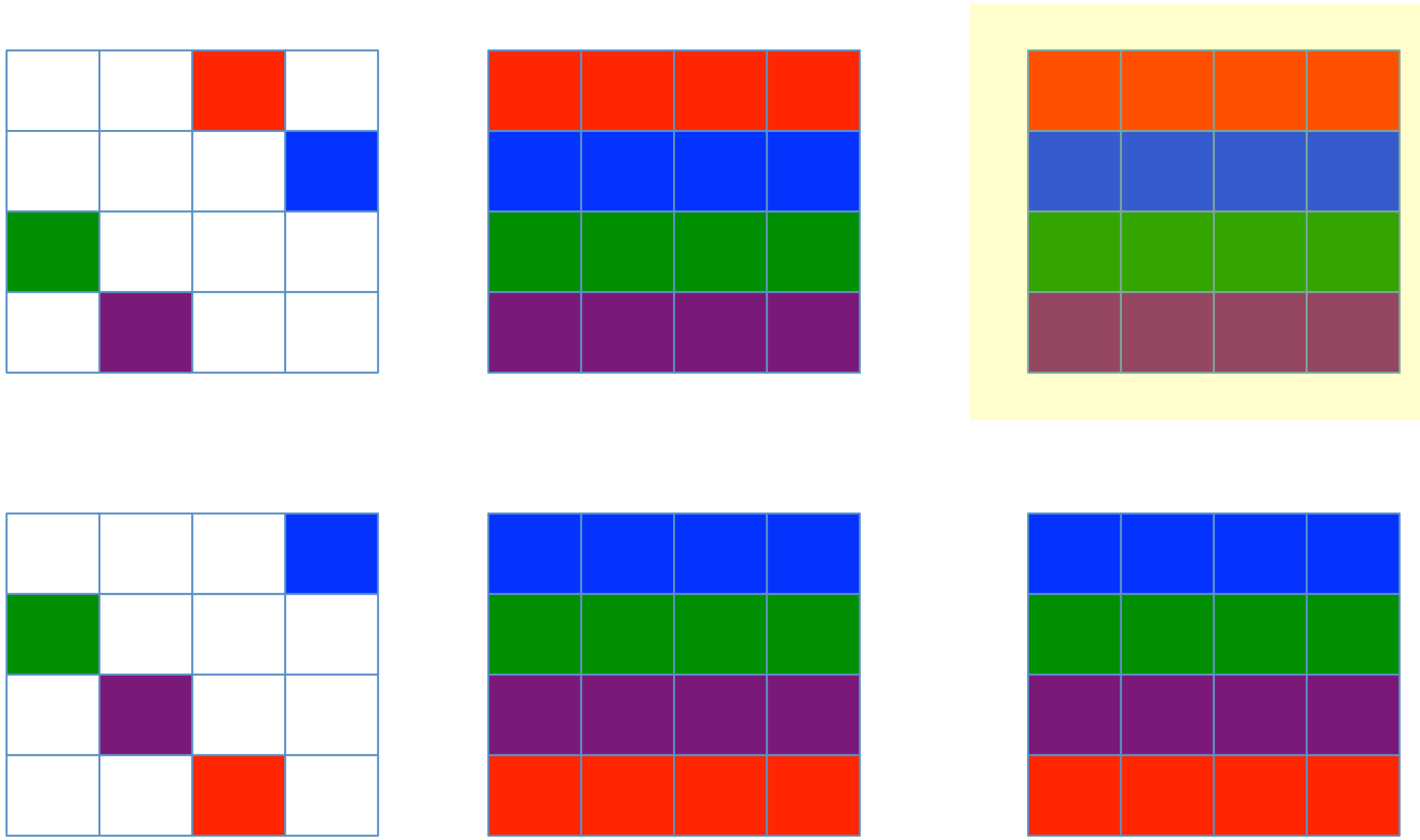
MHPC
Master in High Performance Computing



ma per seguir virtute e canoscenza
SISSA
Scuola Internazionale Superiore
di Studi Avanzati

The Abdus Salam
ICTP
International Centre
for Theoretical Physics

MHPC
Master in High Performance Computing



SISSA
ma per seguir virtute e canoscenza
Scuola Internazionale Superiore di Studi Avanzati

ICTP
The Abdus Salam International Centre for Theoretical Physics