1st Official, 100% Pure, CCA Approved

MxN Working Group Meeting!

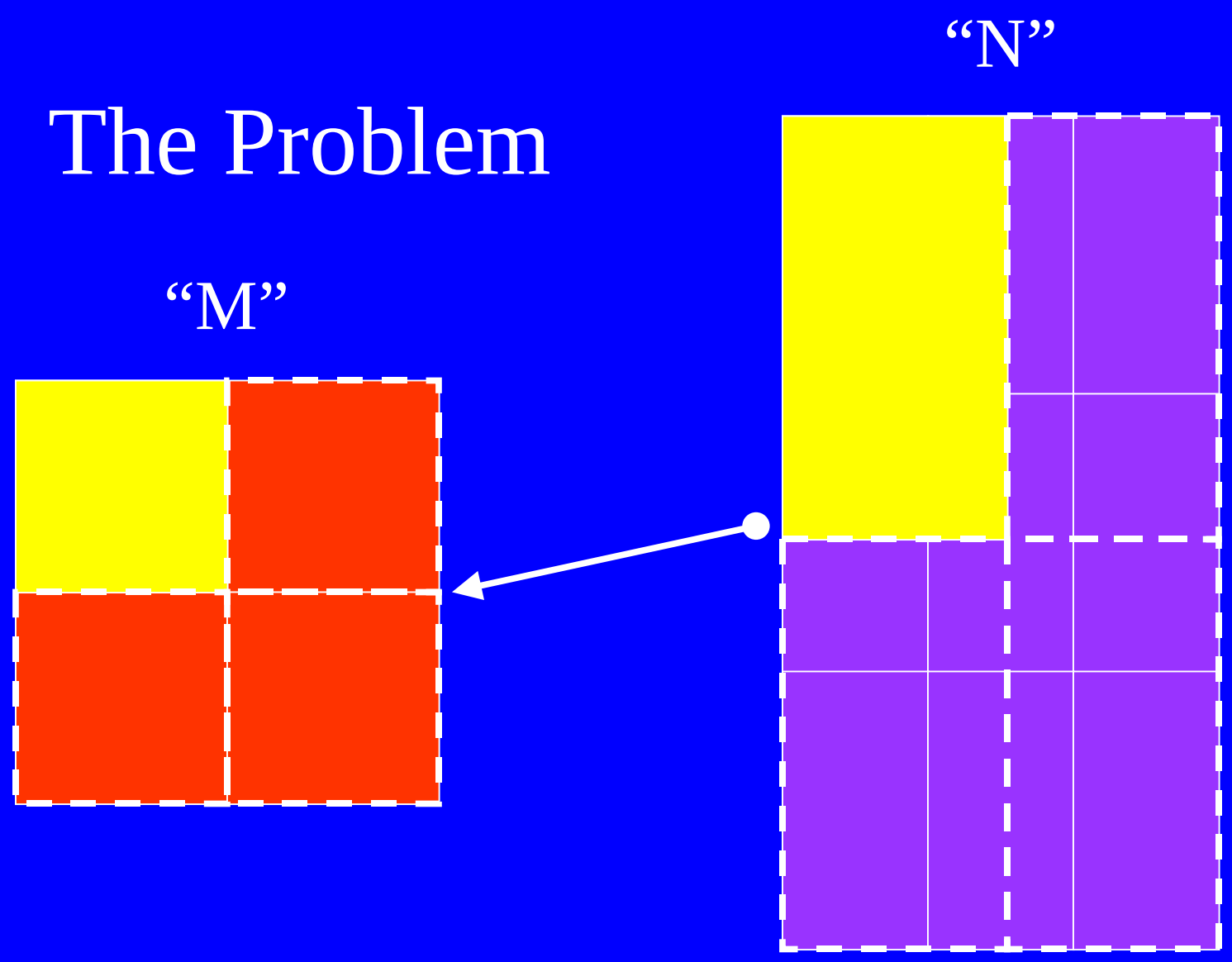
Bishop's Lodge, Santa Fe, NM
January 9, 2002

James "Jeeembo" Kohl, Primary Target
Oak Ridge National Laboratory
(with the full assortment of other MxN-ers!)

“MxN” Parallel Data Redistribution

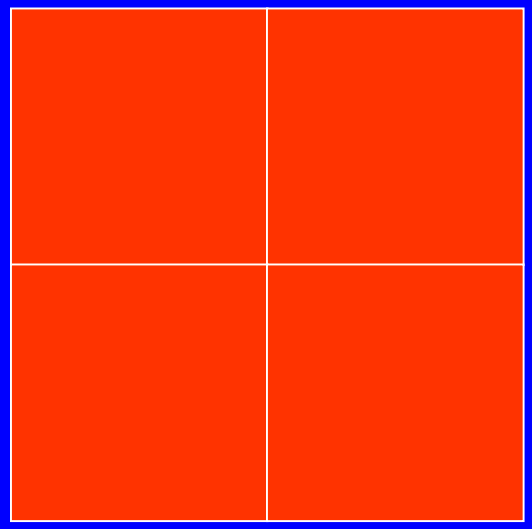
- Share Data Among Coupled Parallel Models
  - ⇒ Ocean & Atmosphere, Solver & Optimizer…
  - ⇒ Visualization (Mx1)
- Encapsulate Existing Technologies:
  - ⇒ CUMULVS (ORNL), PAWS (LANL), Others…
  - ⇒ Data Decompositions, Communication Schedules, Connections & Synchronization Protocols…
- Foundation for “General” Coupling (Later…!)
  - ⇒ Mapping & Interpolation of Parallel Data is **Hard**.
    - Different Grids, Spatial & Temporal, Units

The Problem

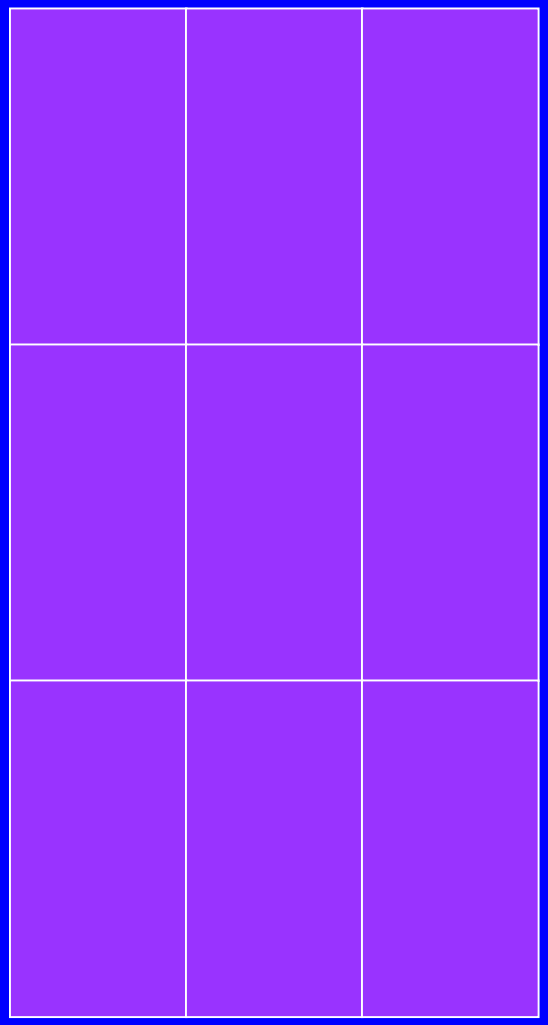


The Problem

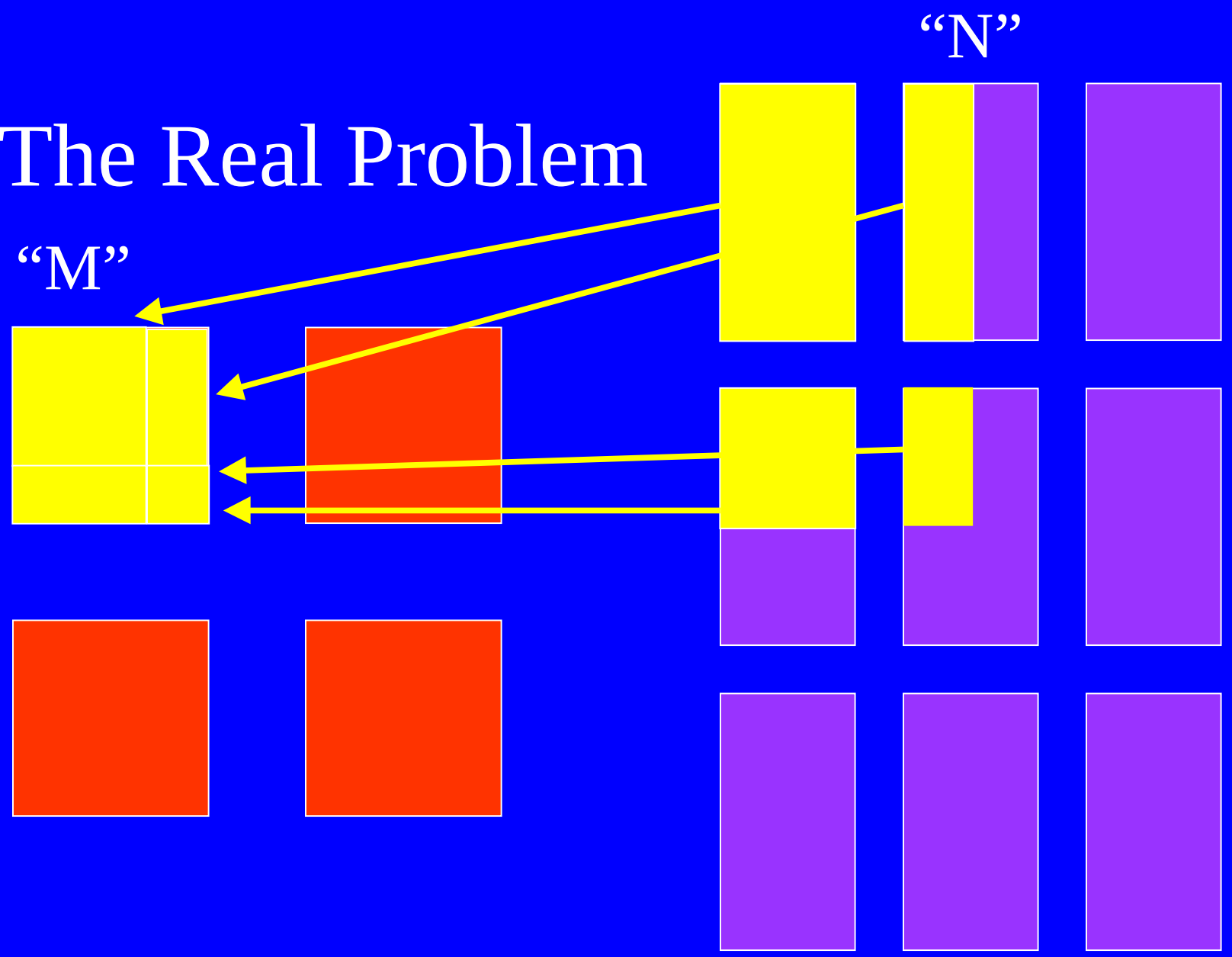
“M”



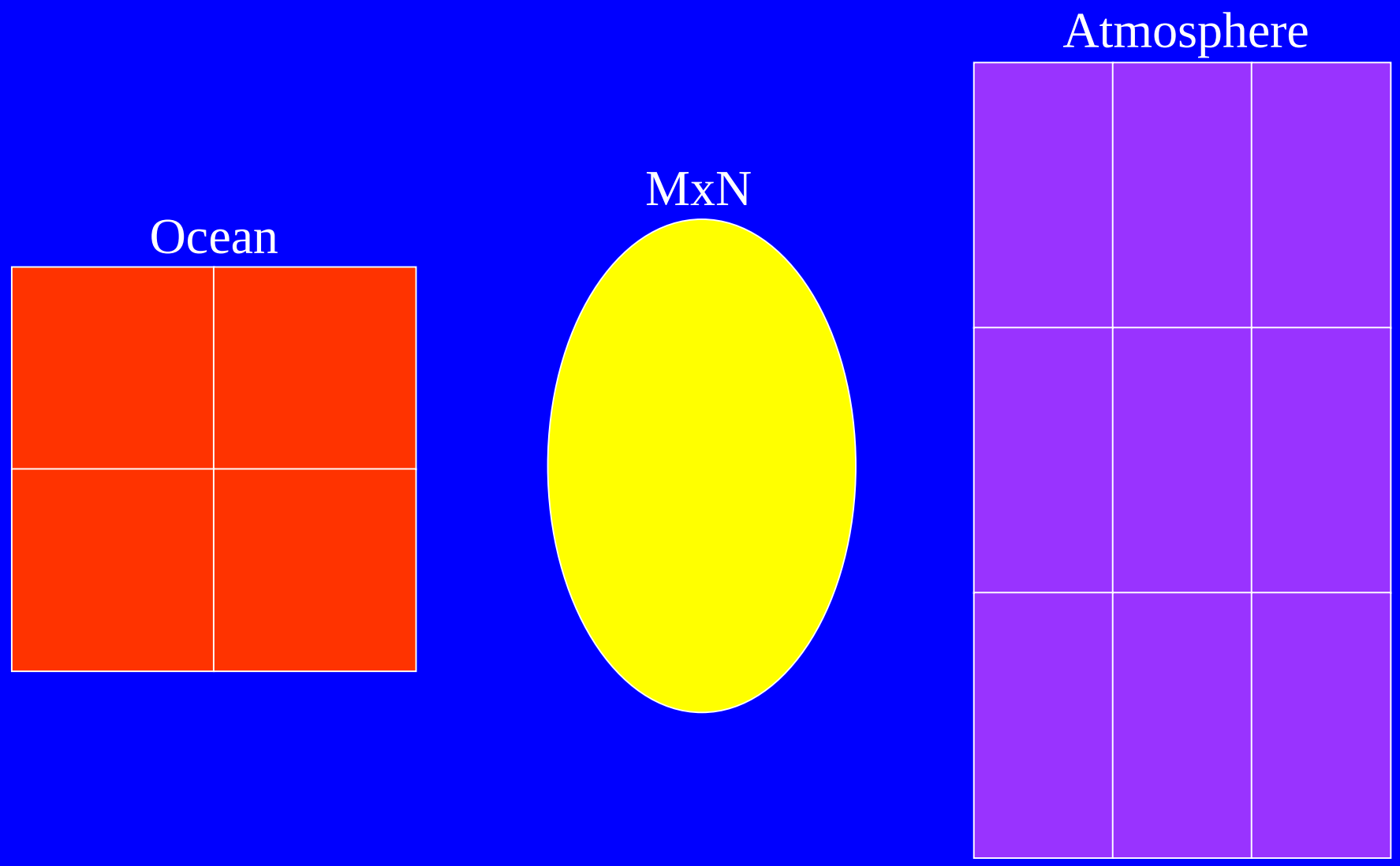
“N”



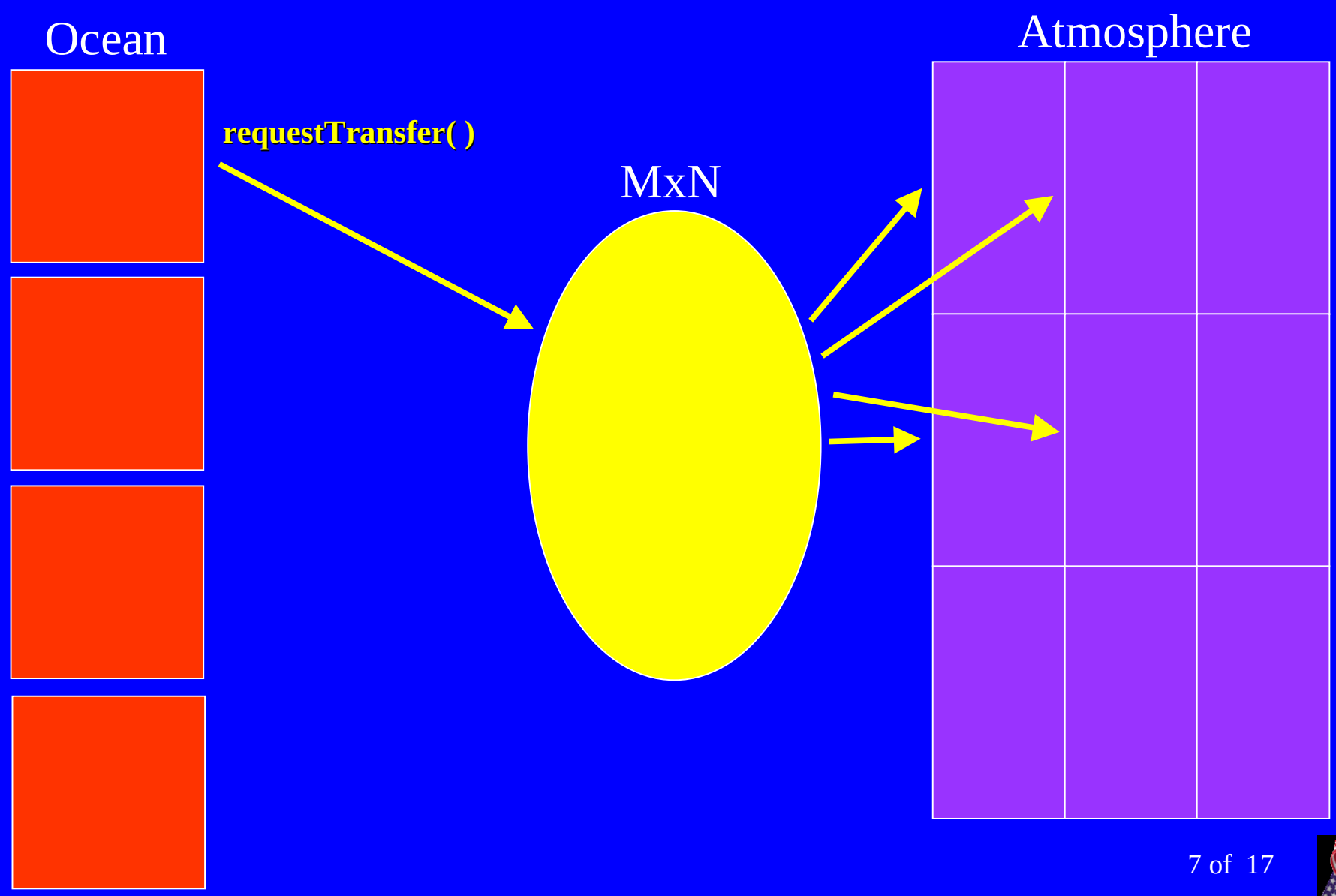
The Real Problem



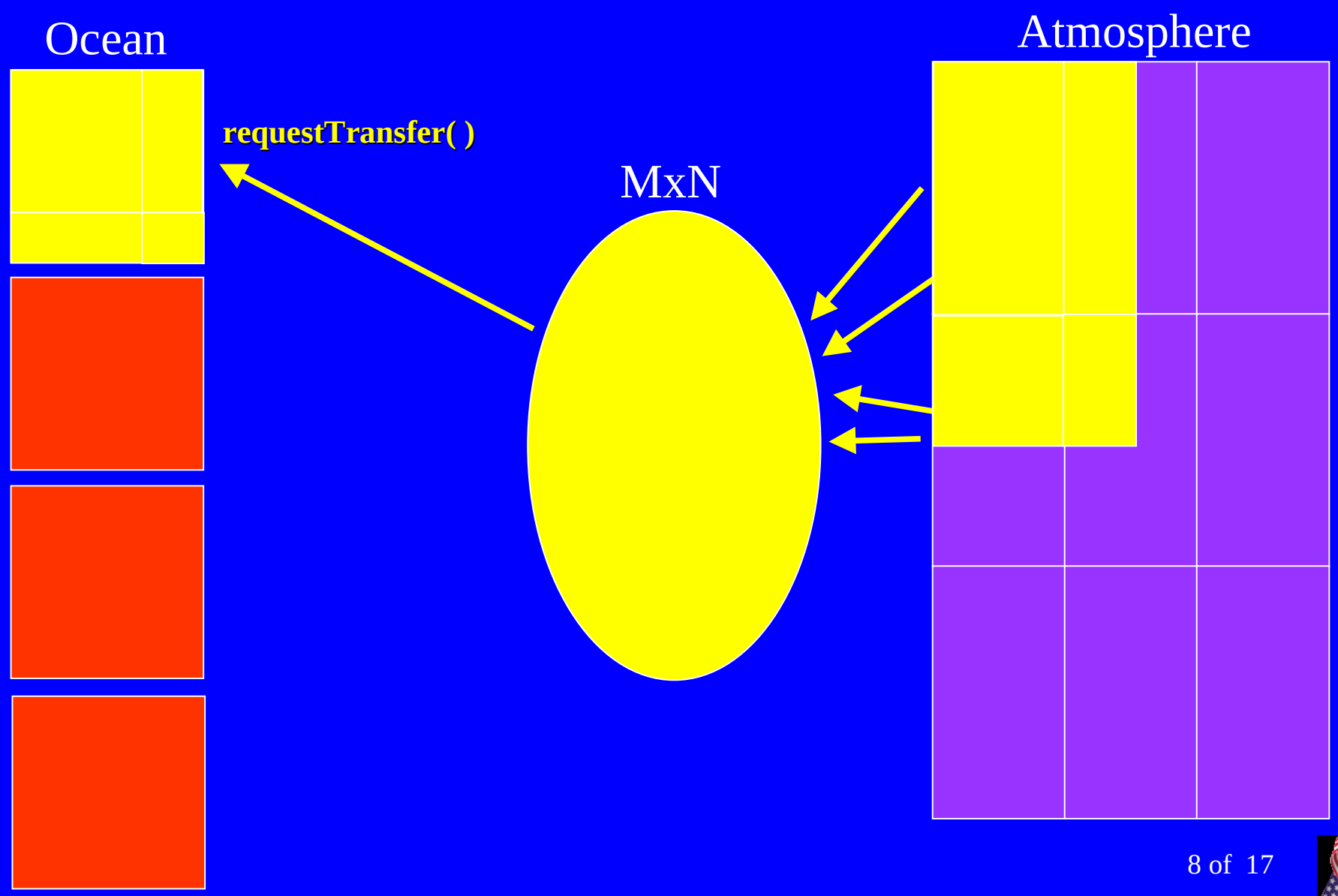
MxN Component Example



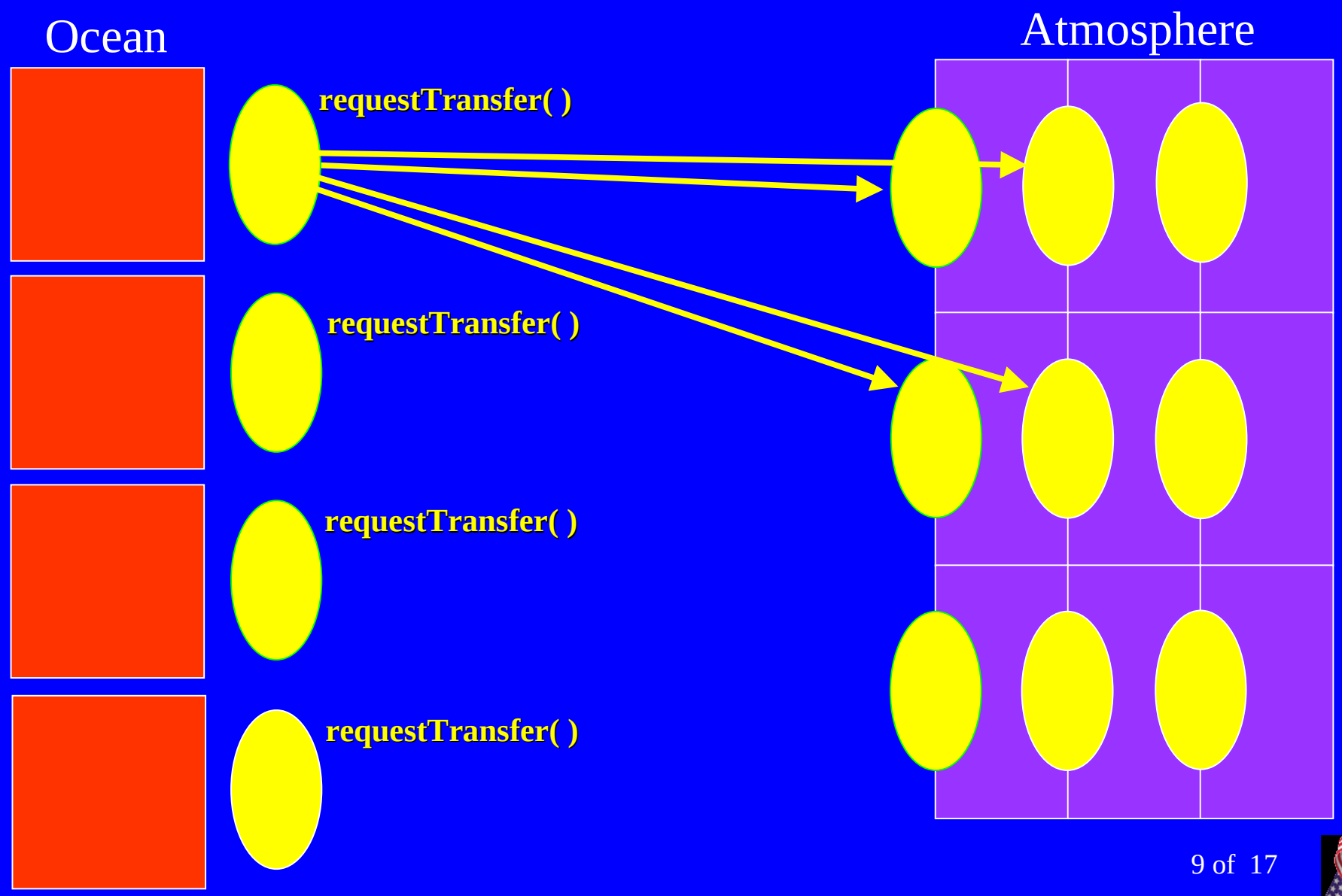
MxN Component Example



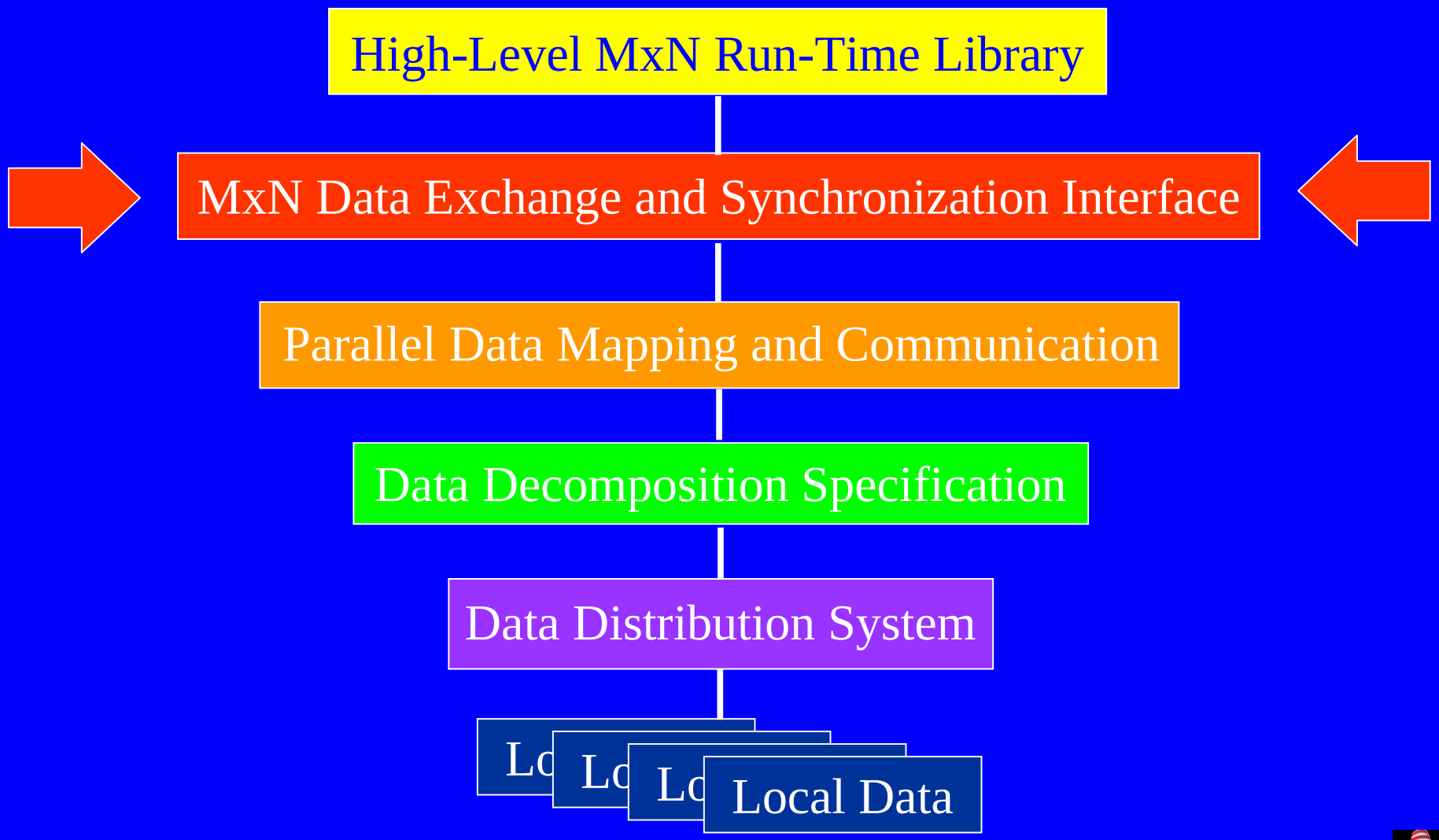
MxN Component Example



Scalable MxN Component Example



MxN Internal Structure



Overview of Current MxN Spec

- General & Flexible
  - ⇒ Legacy / Independent Components
  - ⇒ Direct or 3rd Party Control
- Several Functional Portions
  - ⇒ Data "Registration"
  - ⇒ Communication Schedules
  - ⇒ MxN Connections
  - ⇒ Data Transfers
    - "dataReady( )"…

MxN Data "Registration"

- Tell the "MxN-er" About the Data
  - ⇒ Local Information ~ Ptr, Type, Size
  - ⇒ Distributed Data Decomposition

```
Each A:
  Data Adata;
  registerData( Adata, True, ReadOnly, Adhandle );
                       (Synch)          (Out)

Each B:
  Data Bdata;
  registerData( Bdata, False, WriteOnly, Bdhandle );
                       (Synch)           (Out)
```

MxN Communication Schedules

- Map Together Two Parallel Entities
  - ⇒ Fundamentally, Element by Element…
  - ⇒ Based on Data Decomposition Info

```
All A:
  createCommSched( Bdata, Adata, csHandle );
                                   (Out)
```

(Note: Need Sub-Interface for Manipulating
& Optimizing Comm Schedules…)

MxN Connections

- Connect Up Two Parallel Entities

  ⇒ Commence Synchronization

```
All A:
  makeConnection( Bdhandle, Adhandle, csHandle,
    NoSynch, Default, 4, 5, cHandle );
```

(Src Synch) (Dst Synch) (Src & Dst Freqs) (Out)

MxN Parallel Data Transfers

- Request Actual Data Transfer
  - ⇒ Either Side, or 3rd Party Component Control

```
All A:
  requestTransfer( cHandle, tHandle );
  do {
    . . .
    waitTransfer( tHandle );
  } while ( );
```

Data Ready

- Each Parallel Entity Activates Local Transfer
  - ⇒ "dataReady( )" Releases Local Data for Use
  - ⇒ Indicates Parallel "Consistency" of Data
  - ⇒ Data Elements Transferred As Available…

```
All A:
  dataReady( Adhandle, True, ReadWrite );

All B:
  dataReadyBegin( Bdhandle, True, ReadWrite );
  . . .
  dataReadyEnd( Bdhandle );
```

Current MxN Status

- Draft MxN Interface Specification
  - ⇒ Unifies Several Existing Approaches
    - One-Shot & Periodic Synchronization Models
- Successful SC01 MxN Component Demos
  - ⇒ CUMULVS & PAWS
  - ⇒ Component Re-Use
  - ⇒ Big Tar Ball Coming Soon… ☺
- Next Phase, "Interoperability"…?

Latest Spec: http://www.csm.ornl.gov/cca/mxn/MxNPort.h