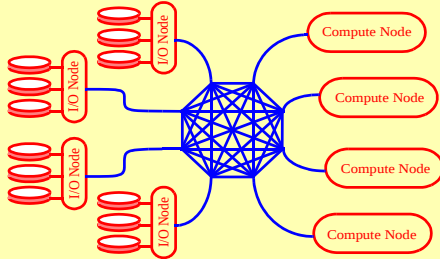


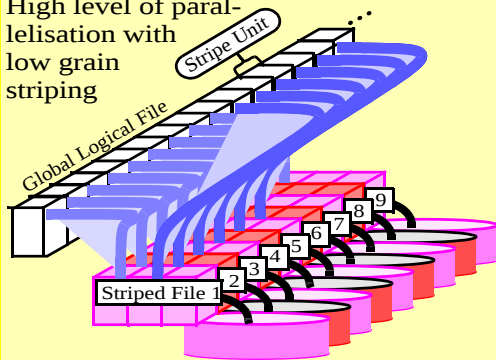
SFIO: MPI-based Parallel I/O for Swiss-Tx

Goal

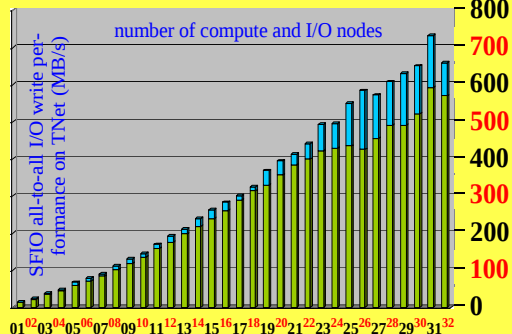
Scalable Throughput for parallel applications



High level of parallelisation with low grain striping

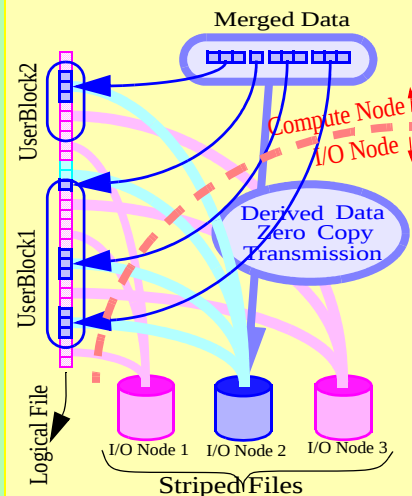


SFIO PERFORMANCE ON SWISS-T1

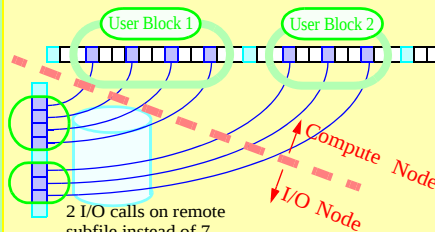


Approach

Network Optimisation



Disk Access Optimization

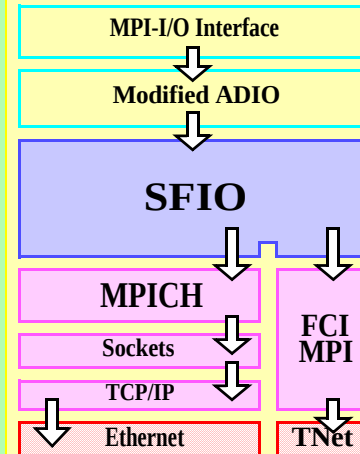


Contacts

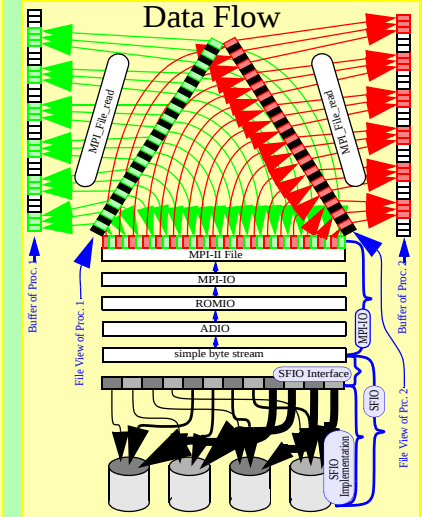
Emin Gabrielyan
e-mail: Emin.Gabrielyan@epfl.ch
Peripheral System Laboratory (LSP)
EPFL, Lausanne, Switzerland

SFIO Integration with MPI

MPI-I/O on SFIO



Data Flow



SFIO Interface

```
MFFILE *f=mopen
```

```
(
    "toneb0/tmp/a0;"
    "toneb1/tmp/a1;"
    "toneb2/tmp/a2;",5
    ,5
);
mwrite(f,0,"Hello World",11);
mclose(f);
```

```
MFFILE *f=mopen(
```

```
"toneb0/tmp/a0;"
"toneb1/tmp/a1;"
"toneb2/tmp/a2;",5);
if(rank()==0)
    mwrite(f,0,"Hello ",6);
else if(rank()==1)
    mwrite(f,6,"World",5);
mclose(f);
```

```
MFFILE* mopen(char *name,int chunk)
```

```
void mwrite(MFFILE* f, long off, char* bu, unsigned sz);
void mread(MFFILE* f, long off, char* bu, unsigned sz);
```

```
void mwriteb(MFFILE* f, unsigned count, long off[], char* bu[], unsigned sz[]);
void mreadb(MFFILE* f, unsigned count, long off[], char* bu[], unsigned sz[]);
```

```
void mclose(MFFILE *f)
```