

# Statically Checking MPI Type Safety

Dmitri Gribenko <[gribozavr@gmail.com](mailto:gribozavr@gmail.com)>  
High Performance Computing Center  
at NTUU “KPI”

# MPI

```
int data[10];
```

```
MPI_Send(data, 10, MPI_INT, /* ... */);
```

# MPI

```
int data[10];
```

```
MPI_Send(data, 10, MPI_INT, /* ... */);
```

# MPI

```
double data[10];
```

```
MPI_Send(data, 10, MPI_INT, /* ... */);
```

# MPI and C Type System

```
int MPI_Send(void *buf,  
            int count,  
            MPI_Datatype datatype,  
            /* ... */);
```

# Solution: annotations

- Changes in:
  - mpi.h
  - compiler
- Advantages:
  - No changes in users' MPI programs
  - Header-only change
  - All checks are done in compile-time

# Annotations

```
int MPI_Send(void *buf, int count, MPI_Datatype datatype, ...)
    __attribute__(( pointer_with_type_tag(mpi,1,3) ));
```

```
extern struct ompi_predefined_datatype_t ompi_mpi_int
    __attribute__(( type_tag_for_datatype(mpi,int) ));
```

```
#define MPI_INT    (&mpi_mpi_int)
```

```
double *double_buf;
```

```
MPI_Send(double_buf, 1, MPI_INT, /* ... */);
```

# Diagnostics

wrong.c:151:12: **warning**: argument type 'double \*'  
doesn't match specified 'mpi' type tag  
that requires 'int \*' [-Wtype-safety]

```
MPI_Send(double_buf, 1, MPI_INT, /*...*/);
```

^~~~~~  
~~~~~



# Implementation Status

- Implemented in mainline Clang
- Annotated mpi.h for MPICH2, available in version 1.5rc1
- Patch for OpenMPI is under review