The Australian National University

IMPROVEMENT OF UNUM COMPUTING SOFTWARE

STUDENT: Sidong Feng

SUPERVISOR: Josh Milthorpe

MOTIVATION

```
double a = 2.0
for i = 0 to 100
    a = Math.sqrt(a)
end for
---> a = 1.0
```

IEEE STANDARD FOR FLOATING –POINT ARITHMETIC

Rounding Rule: numbers are rounded to the nearest floating-point number

Example:

$\sqrt{2} = 1.0$ if we round to 1 decimal

SLEIPNER OIL PLATFORM DISASTER

Collapsed to ocean floor; Float error in structural analysis

UNUM vs FLOAT

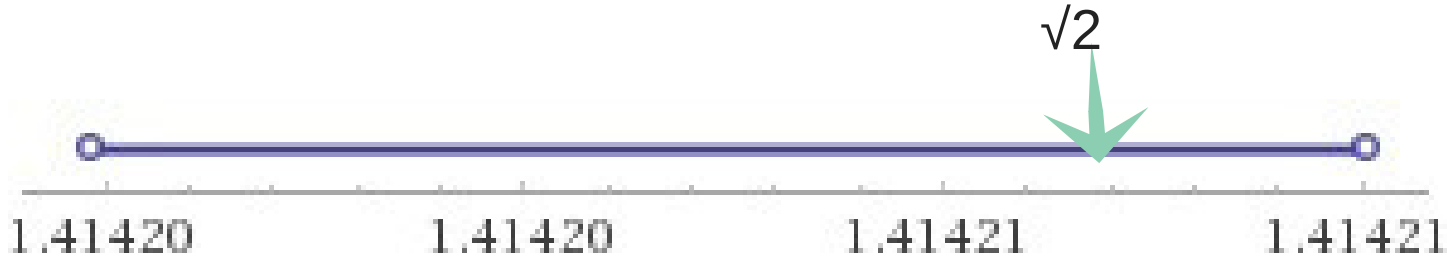
- More Accurate

For example: Represent √2

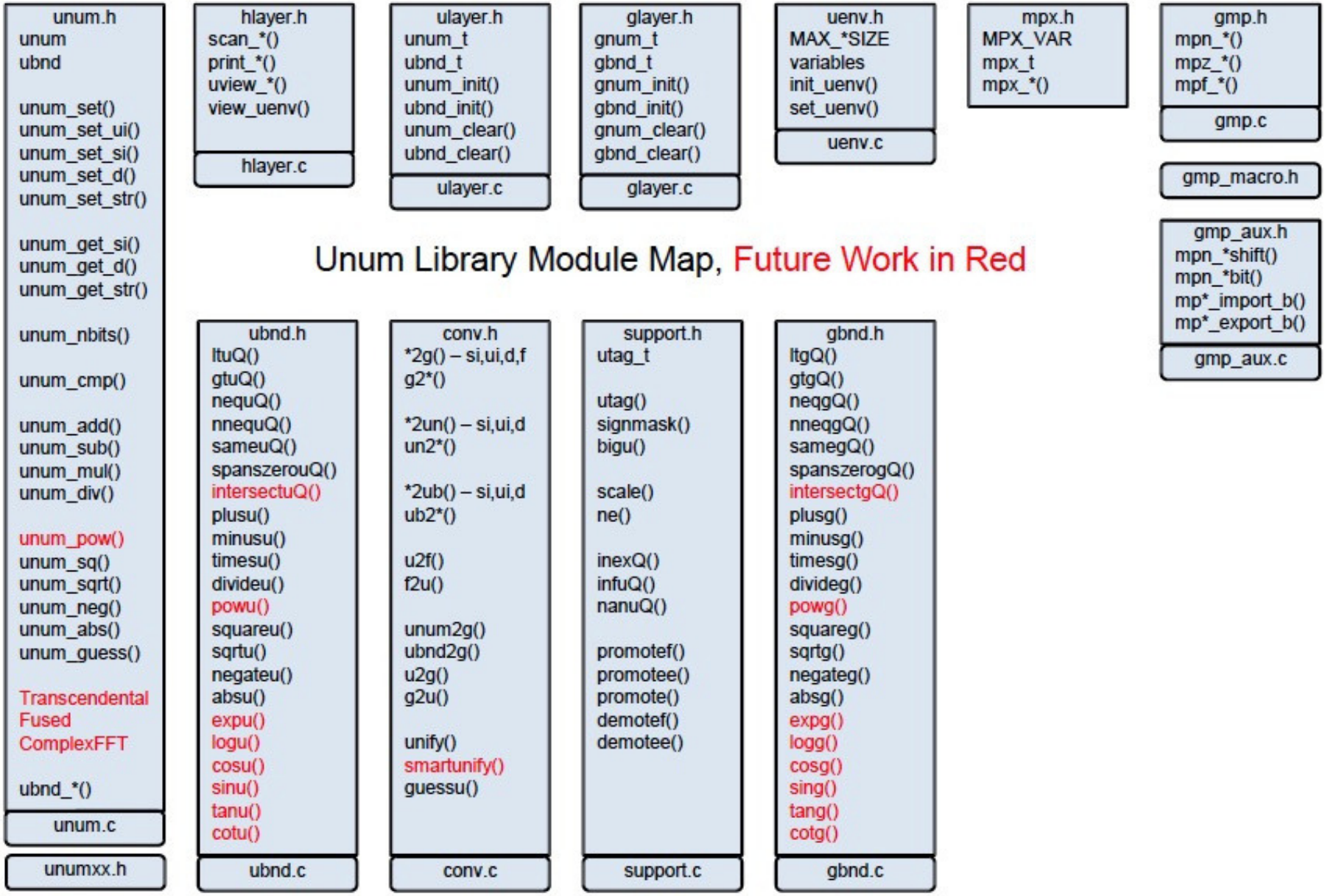
Float : 1.4142135623730951

Unum: (1.4141998291015625,1.41421508789062 5)

Unum format: 0 01 0110101000001001 1 001 1111

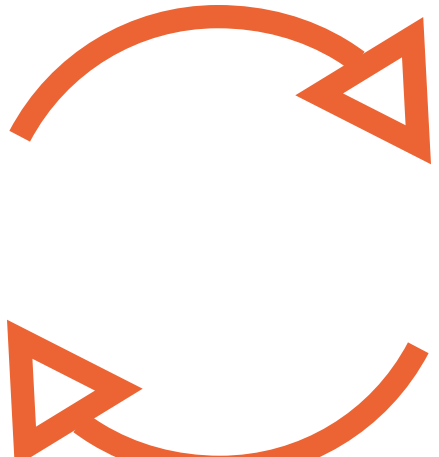


Unum Library by LLNL (Lawrence Livermore National Laboratory)



PROJECT GOAL

- Implement more arithmetic operations
- Improve performance



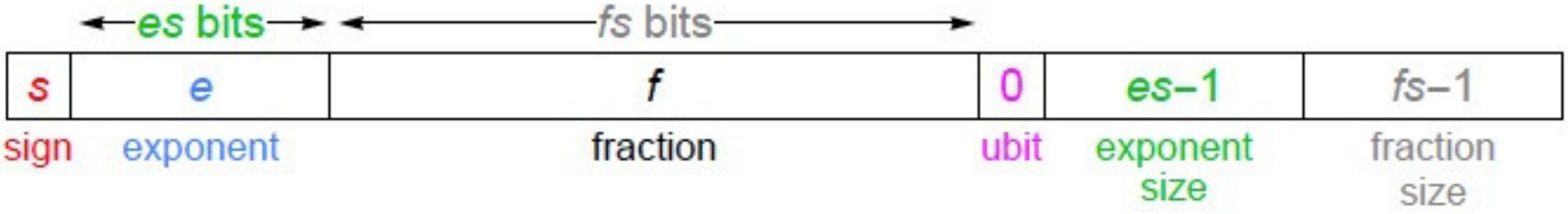
PLAN

- Understanding Unum
- Understanding Library
- Implementing Operations
- Debug and Analysis

- Implementing efficient operations and calculating with no error !!!

BONUS

Unum format:



$$x = (-1)^s \times \begin{cases} 2^{2-2^{es-1}} \times \left(\frac{f}{2^{fs}}\right) & \text{if } e = \text{all 0 bits,} \\ \infty & \text{if } e,\ f,\ es,\ \text{and } fs \text{ have all their bits set to 1,} \\ 2^{1+e-2^{es-1}} \times \left(1 + \frac{f}{2^{fs}}\right) & \text{otherwise.} \end{cases}$$

Unum format: 0 1 0 0 0 0

= 2

BONUS

UNUM vs FLOAT

- Easier to use
- Less demanding on memory and bandwidth

For example: Represent 2

Float(32-bit) : 0 10000000 00000000000000000000000

Unum format: 0 1 0 0 0 0