The new t_REAL format

B. Allombert

IMB
CNRS/Université Bordeaux 1

27/01/2012

Motivation

- The current t_REAL format
- The new t_REAL format

Compatibility issue

Motivation for a new format

.

In this talk we will always assume a 64bit platform.
We want to address two limitations of the traditional format :

- The precision of a t_REAL is multiple of 64bit, instead of a number of bits.
- The real significand word order is different for the integer word order (with the GMP kernel).

Lignes directrices

The current t_REAL format

```
[ code_1 ] [ code_2 ] [ man_1 ] ... [ man_k ]
```

- code_1 : type and total length l
- code_2 : sign s and exposant e
- man_1...man_k : significand, normalized, $64k$ bits, most significant word first.

The accuracy in bit is $p = (l - 2) \times 64$. The number represented is $x = sM \times 2^{e-1-64k}$.

Lignes directrices

The new t_REAL format

```
[ code_1 ] [ code_2 ] [ man_1 ] ... [ man_k ]
```

- code_1 : type and total length
- code_2 : sign s, "rough exposant" E and "lost precision" P.
- man_1...man_k : significand, not normalized, same word order as for integers.

If p is the accuracy, in bit, then the total length is $l = \frac{p+126}{64}$, rounded down.

The significand

The bits man_1...man_k are interpreted as the unnormalized mantissa of an integer M. We require the lowest P bit of N to be zero.
The accuracy is the exponent of M, minus P.
The number represented is $x = sM2^{64\times(E+1-k)}$.

Compatibility issue

The main compatibility issue is that now, the `prec` parameter to functions will be in bits instead of length. For example

```
GEN gexp(GEN x, long prec)
```

Now `prec` is in bit. However as long as `gexp` only use prec as an argument to other functions, then `gexp` does not need to be changed !
It is no more possible to use `setlg` to reduce the precision of a `t_REAL` . Indeed with GMP, this would remove the most significant bits.
`cgetr` takes a precision instead of a length.

The intermediary stage

Currently PARI is in an intermediary stage where it needs to work with both formats. So a lot of macros has been introduced that will be removed once the conversion is finished, for example `prec2nbits`, `bit_prec`, etc.
In the old system, the precision of a `t_REAL` *x* was obtained by `prec=lg(x)`. Please now use `prec=realprec(x)`.