Performance Improvements in 2022

NumberTheory

- The ChineseRemainder command in the NumberTheory package now handles large systems of congruences much more quickly than in previous releases, and can take better advantage of multi-core systems.

- Timings can vary widely between machines but, on one particular 32-core machine, the following call to ChineseRemainder took about half an hour (real time) in Maple 2021, but only about half a minute in 2022. On another 4-core machine, the call completed in about 17 seconds (real time) in 2022, while running for about 21 minutes in Maple 2021. (Both machines were running the same version of the Linux operating system.)

```maple
> R := [seq]( 1 .. 10^5 ):
> M := [seq]( ithprime( i ), i = 1 .. 10^5 ):
> res := CodeTools:-Usage( NumberTheory:-ChineseRemainder( R, M ) ):
```

Bell Numbers

- Computing Bell numbers, using the bell command in the combinat package, has been sped up considerably. Before Maple 2022, the following computation took 7 seconds and more than a GB of memory on a machine whereas it now takes 35ms and less than 5MB:

```maple
> combinat:-bell(175);
```

Units[Simple] Package Operations

- Operations using the Units[Simple] package have been improved to be faster and use less memory in Maple 2022. See the units update page for more details.

ArrayTools[Reverse]

- The command ArrayTools[Reverse] has been sped up in Maple 2022. The speed at which the command works now depends on the datatype of the argument; the following cases now take about 0.3s and 0.1s, respectively, on a machine where either used to take around 15-20s. The memory use has been reduced by about a factor 10.

```maple
> with_datatype_anything := Vector(10^7, i -> i):
```
CodeTools:-Usage(ArrayTools:-Reverse(with_datatype_anything)):
memory used=76.31MiB, alloc change=76.30MiB, cpu time=204.00ms, real
time=113.00ms, gc time=141.22ms

with_datatype_float := Vector(10^7, i -> i, datatype=float):

CodeTools:-Usage(ArrayTools:-Reverse(with_datatype_float)):
memory used=76.30MiB, alloc change=76.30MiB, cpu time=57.00ms, real
time=57.00ms, gc time=0ns

IntersectionMultiplicity Command

- The IntersectionMultiplicity command in the RegularChains:-AlgebraicGeometryTools
  package has a newly designed algorithm that makes the command much faster. Details
  are provided in the Intersection Multiplicities section of the Advanced Math page.

DEtools

- The FindODE command in the DEtools package is more efficient on many examples.
  For more information see the Differential Equations section of the Advanced Math page.