Sage comme calculatrice

```
>>>  1+1

     2

>>>  factorial(50)

     30414093201713378043612608166064768844377641568960512000000000000

>>>  s = add(1/n for n in (1..30))

>>>  s

     9304682830147/2329089562800

>>>  s.n()

     3.99498713092039

>>>
```

Python

```
>>>  a = -2/3

>>>  type(a)

     <type 'sage.rings.rational.Rational'>

>>>  dir(a)[140:160]

     ['_singular_',
     '_singular_init_',
     '_sub_',
     '_sympy_',
     '_test_category',
     '_test_eq',
     '_test_nonzero_equal',
     '_test_not_implemented_methods',
     '_test_pickling',
     '_tester',
     'abs',
     'absolute_norm',
     'additive_order',
     'base_extend',
     'base_ring',
     'cartesian_product',
     'category',
     'ceil',
     'charpoly',
     'conjugate']

>>>  a.abs(), abs(a)

     (2/3, 2/3)

>>>  def myfact(n):
         res = 1
         for k in srange(1, n+1):
             res = res*k
         return res

>>>  myfact(50)

     30414093201713378043612608166064768844377641568960512000000000000

>>>
```

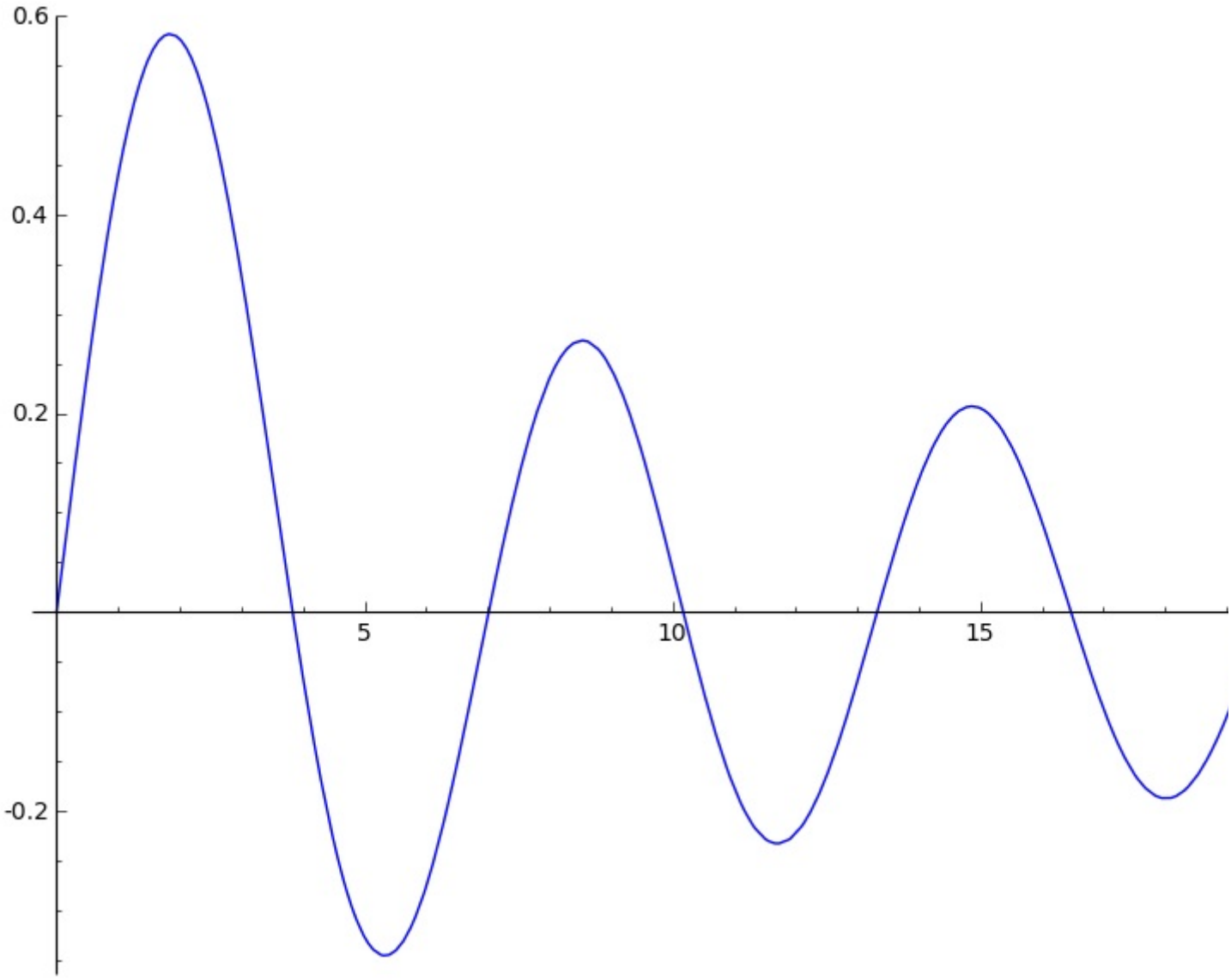
Expressions symboliques

```
>>>  x, y = var('x', 'y')
     u = cos(x)*sin(y)
     u

     cos(x)*sin(y)
```

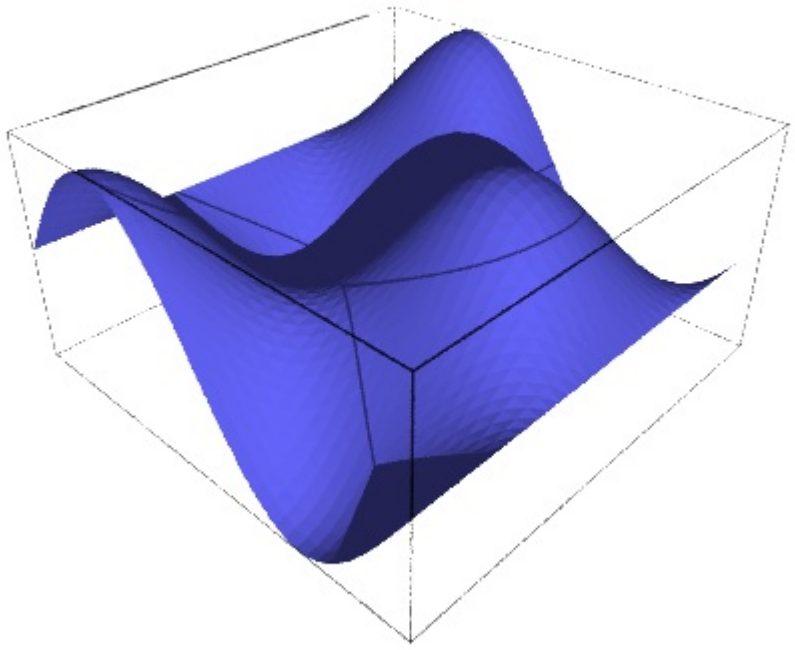
```
>>>   diff(u, x)

      -sin(x)*sin(y)

>>>   u.derivative(y)

      cos(x)*cos(y)

>>>   u.series(x, order=5)

      (sin(y)) + (-1/2*sin(y))*x^2 + (1/24*sin(y))*x^4 + Order(x^5)

>>>   u(x=1)

      cos(1)*sin(y)

>>>   u(x=sqrt(2), y=1).n(prec=1000)

      0.13122209440880167354179466394943557258151479997881813049404739139275005288731829814184371149465604288553757327376174379326766605682621211798986525637145053119092297681960434525400692467056441133781847596004378690230103677152626720456712901449325329568055173470137249434273963974796978559495221133134 5
>>>
```

Graphiques

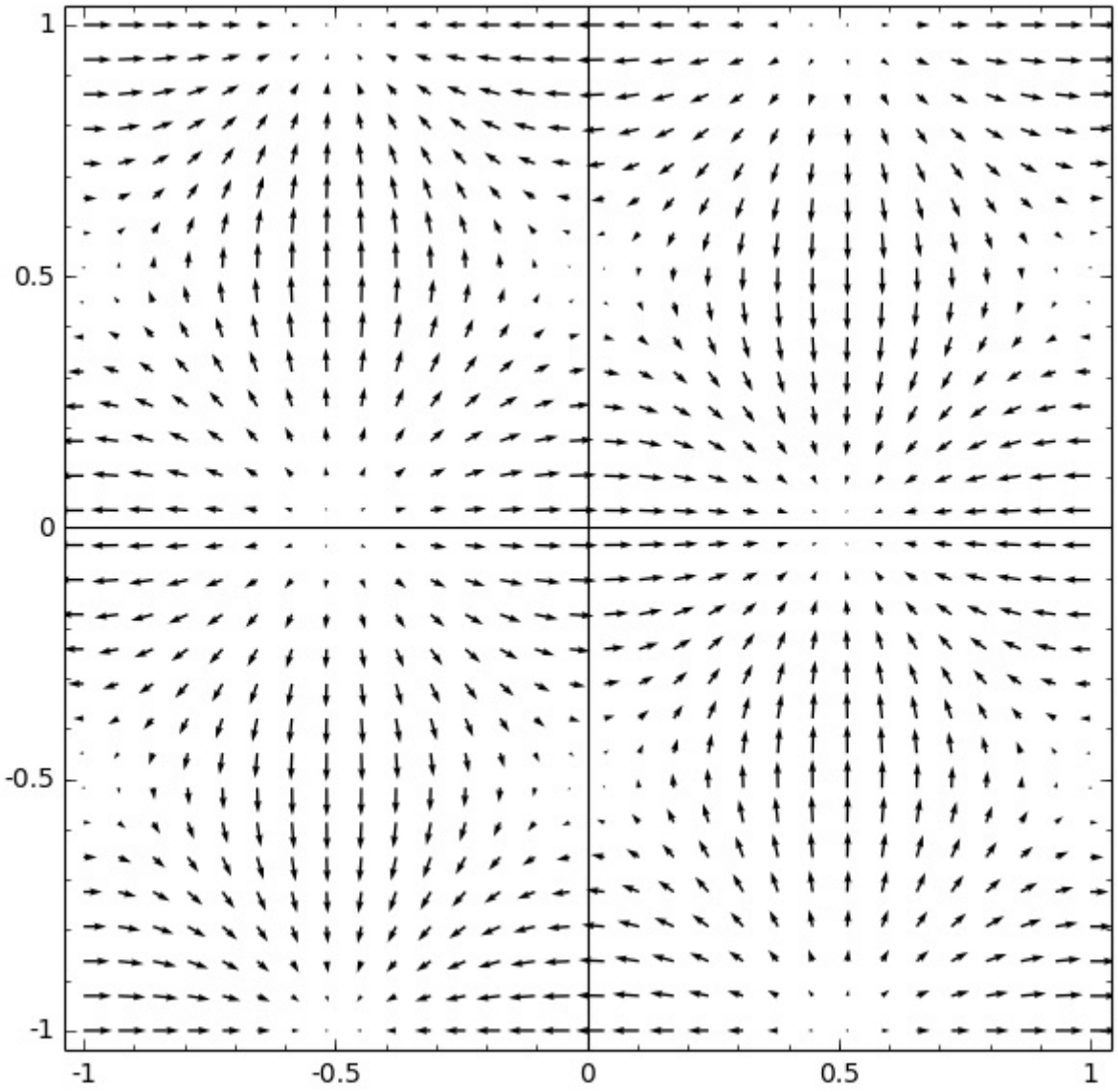
```
>>>   plot(Bessel(1,'J'), 0, 20)
```



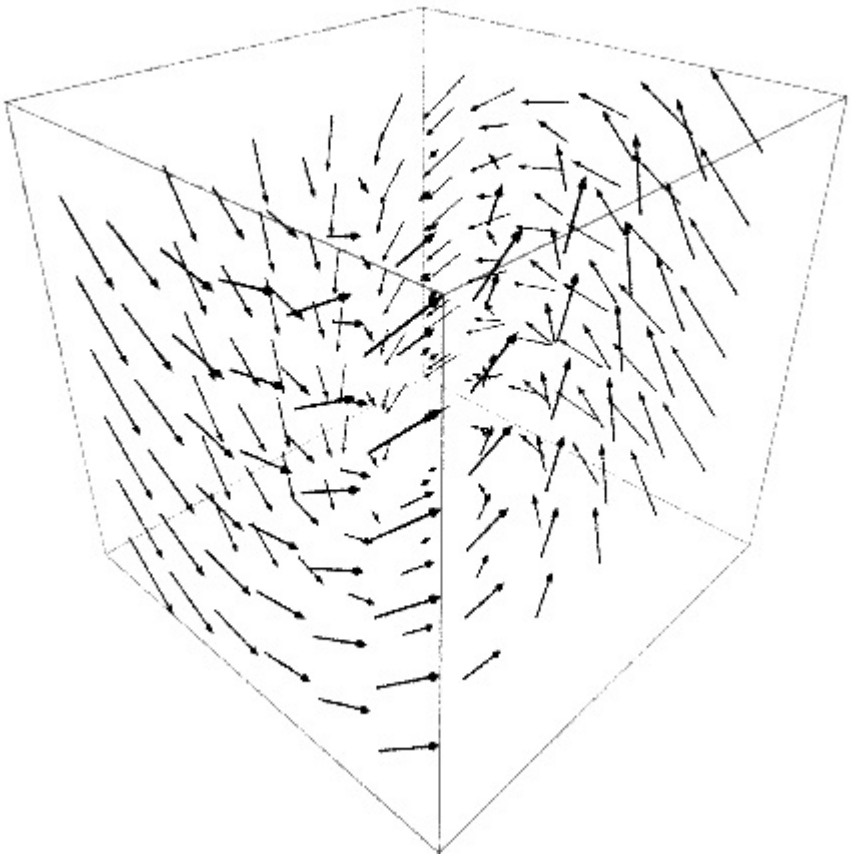
```
>>>   u, v = var('u', 'v')
      f = sin(pi*u) * cos(pi*v)
      plot3d(f, (u, -1, 1), (v, -1, 1))
```



```
>>>   plot_vector_field(f.gradient(), (u,-1,1), (v,-1,1), aspect_ratio=1,
      plot_points=30)
```



```
>>>   x,y,z=var('x y z')
      myplot = plot_vector_field3d(
      (-y, 0, x), (x,-2,2), (y,-2,2), (z,-2,2),
      colors='black', plot_points=6)
      show(myplot, viewer="tachyon")
```



```
>>>
```

Éléments et parents

```
>>>   type(1)
      <type 'sage.rings.integer.Integer'>
>>>   parent(1)
      Integer Ring
>>>   ZZ
      Integer Ring
>>>   parent(1) is ZZ
      True
>>>   parent(1/1)
      Rational Field
>>>   type(ZZ)
      <type 'sage.rings.integer_ring.IntegerRing_class'>
>>>   type(QQ)
      <class 'sage.rings.rational_field.RationalField_with_category'>
>>>   ZZ.cardinality()
      +Infinity
```

```
>>>   MyParent = QQ.cartesian_product(ZZ); MyParent
      The cartesian product of (Rational Field, Integer Ring)
>>>   MyParent.is_ring()
      True
>>>   MyParent.an_element()
      (1/2, 1)
>>>   QQ.category()
      Category of quotient fields
>>>
```

Quelques parents

```
>>>   Integers()
      Integer Ring
>>>   Rationals()
      Rational Field
>>>   R = IntegerModRing(10); R
      Ring of integers modulo 10
>>>   R(8)^2
      4
>>>   Reals()
      Real Field with 53 bits of precision
>>>   Complexes()
      Complex Field with 53 bits of precision
>>>   MatrixSpace(RDF, 2, 3)
      Full MatrixSpace of 2 by 3 dense matrices over Real Double Field
>>>   PolynomialRing(QQ, 'x')
      Univariate Polynomial Ring in x over Rational Field
>>>   PolynomialRing(QQ, 'x, y, z')
      Multivariate Polynomial Ring in x, y, z over Rational Field
>>>   MatrixSpace(PolynomialRing(ZZ, 'x'), 2)
      Full MatrixSpace of 2 by 2 dense matrices over Univariate Polynomial Ring in x over Integer Ring
>>>   _.random_element()
      [x^2 + 12*x - 1 -x^2]
      [ x^2 + x x + 84]
>>>
>>>
```

Conversions

```
>>>   RDF
      Real Double Field
>>>   RDF(42)
      42.0
```

```
>>>   RDF(42).parent()

      Real Double Field

>>>   ZZ(1.0)

      1

>>>   ZZ(1.5)

      ---------------------------------------------------------------------------
      TypeError Traceback (most recent call last)
      <ipython-input-69-562488d24cf3> in <module>()
      ----> 1 ZZ(RealNumber('1.5'))

      /home/marc/co/sage/local/lib/python2.7/site-packages/sage/structure/parent.so in
      sage.structure.parent.Parent.__call__ (build/cythonized/sage/structure/parent.c:9603)()

      /home/marc/co/sage/local/lib/python2.7/site-packages/sage/structure/coerce_maps.so in
      sage.structure.coerce_maps.NamedConvertMap._call_
      (build/cythonized/sage/structure/coerce_maps.c:5577)()

      /home/marc/co/sage/local/lib/python2.7/site-packages/sage/rings/real_mpfr.so in
      sage.rings.real_mpfr.RealNumber._integer_ (build/cythonized/sage/rings/real_mpfr.c:15923)()

      TypeError: Attempt to coerce non-integral RealNumber to Integer
>>>
```

Coercitions

(= conversions canoniques automatiques)

```
>>>   a = 42
      a, a.parent()

      (42, Integer Ring)

>>>   b = a + 1/2
      b, b.parent()

      (85/2, Rational Field)

>>>   c = b + 1/2
      (c, c.parent())

      (43, Rational Field)

>>>   d = ZZ(c)
      (d, d.parent())

      (43, Integer Ring)

>>>   M = MatrixSpace(ZZ, 3); M

      Full MatrixSpace of 3 by 3 dense matrices over Integer Ring

>>>   obj = M.identity_matrix() + 1/2

>>>   obj

      [3/2 0 0]
      [ 0 3/2 0]
      [ 0 0 3/2]

>>>   obj.parent()

      Full MatrixSpace of 3 by 3 dense matrices over Rational Field

>>>
```