Massive On-shell QED One-loop Box

```
(* Time of evaluation and the version of the Mathematica kernel used *)
```

```
DateString[]
```

```
Wed 23 Sep 2015 13:59:13
```

```
$Version
```

```
9.0 for Linux x86 (64-bit) (November 20, 2012)
```

Derivation of MB representation

```
<< MB.m
```

```
MB 1.2

by Michal Czakon

improvements by Alexander Smirnov

more info in hep-ph/0511200

last modified 2 Jan 09
```

```
<< AMBREv1.2.m
```

```
by K.Kajda   ver: 1.2
last modified 9 Apr 2008
last executed on 23.09.2015 at 13:59
```

```
(* The Exp[eps EulerGamma] is added later *)
```

```
boxB4l2m = Fullintegral[{1},
   {PR[k1, m, n1] PR[k1 + p1, 0, n2] PR[k1 + p1 + p2, m, n3] PR[k1 - p3, 0, n4]}, {k1}];
```

$$\text{invariants} = \left\{p1^2 \to m^2,\ p2^2 \to m^2,\ p3^2 \to m^2,\ p1\,p2 \to -m^2 + \frac{s}{2},\ p1\,p3 \to -m^2 + \frac{t}{2},\ p2\,p3 \to -\left(-m^2 + \frac{s}{2} + \frac{t}{2}\right)\right\};$$

```
IntPart[1]
```

```
numerator=1
integral=PR[k1, m, n1] PR[k1 + p1, 0, n2] PR[k1 + p1 + p2, m, n3] PR[k1 - p3, 0, n4]
momentum=k1
```

Fauto::mode :
U and F polynomials will be calculated in AUTO mode. In order to use MANUAL mode execute Fauto[0].

```
repr = SubLoop[integral];
```

```
Iteration nr1: >>Integrating over k1<<

Computing U & F polynomial in AUTO mode >>Fauto[1]<<

U polynomial...
X[1] + X[2] + X[3] + X[4]
F polynomial...
m^2 FX[X[1] + X[3]]^2 - s X[1] X[3] - t X[2] X[4]

Final representation:

((-1)^(n1+n2+n3+n4) (m^2)^z1 (-s)^z2 (-t)^(2-eps-n1-n2-n3-n4-z1-z2) Gamma[-z1]
   Gamma[2 - eps - n1 - n2 - n3 - z1 - z2] Gamma[2 - eps - n1 - n3 - n4 - z1 - z2]
   Gamma[-z2] Gamma[-2 + eps + n1 + n2 + n3 + n4 + z1 + z2]
   Gamma[n3 + 2 z1 + z2 - z3] Gamma[-z3] Gamma[-2 z1 + z3] Gamma[n1 + z2 + z3]) /
 (Gamma[n1] Gamma[n2] Gamma[n3] Gamma[4 - 2 eps - n1 - n2 - n3 - n4] Gamma[n4] Gamma[-2 z1])
```

fin = repr /. {n1 → 1, n2 → 1, n3 → 1, n4 → 1}

```
((m^2)^z1 (-s)^z2 (-t)^(-2-eps-z1-z2) Gamma[-z1] Gamma[-1 - eps - z1 - z2]^2
   Gamma[-z2] Gamma[2 + eps + z1 + z2] Gamma[1 + 2 z1 + z2 - z3] Gamma[-z3]
   Gamma[-2 z1 + z3] Gamma[1 + z2 + z3]) / (Gamma[-2 eps] Gamma[-2 z1])
```

(* Appling Barnes-Lemmas *)

fin = BarnesLemma[fin, 1]

```
>> Barnes 1st Lemma will be checked for: {z3} <<
   Starting with dim=3 representation...

1. Checking z3...Barnes Lemma was applied.
>> Representation after 1st Barnes Lemma:  <<
   1st Barnes Lemma was applied for: {z3}
   Obtained representation has: dim=2

((m^2)^z1 (-s)^z2 (-t)^(-2-eps-z1-z2) Gamma[-z1] Gamma[-1 - eps - z1 - z2]^2
   Gamma[-z2] Gamma[1 + z2]^2 Gamma[2 + eps + z1 + z2] Gamma[2 + 2 z1 + 2 z2]) /
 (Gamma[-2 eps] Gamma[2 + 2 z2])
```

(* x=m^2/t, y=s/t *)

fin = fin /. (m^2)^z1 (-s)^z2 (-t)^(-2-eps-z1-z2) -> (-x)^z1 (y)^z2 (-t)^(-2-eps)

```
((-t)^(-2-eps) (-x)^z1 y^z2 Gamma[-z1] Gamma[-1 - eps - z1 - z2]^2 Gamma[-z2] Gamma[1 + z2]^2
   Gamma[2 + eps + z1 + z2] Gamma[2 + 2 z1 + 2 z2]) / (Gamma[-2 eps] Gamma[2 + 2 z2])
```

(* Finding rules and applying MBcontinue *)

Rules = MBoptimizedRules[fin, eps → 0, {}, {eps}]

MBrules::norules : no rules could be found to regulate this integral

$$\left\{\left\{eps \to -\frac{3}{4}\right\},\ \left\{z1 \to -\frac{1}{2},\ z2 \to -\frac{1}{4}\right\}\right\}$$

integrals = MBcontinue[fin, eps → 0, Rules] // MBmerge // Simplify

```
Level 1
Taking -residue in z2 = -1 - eps - z1
Level 2
Integral {1}
2 integral(s) found
```

$$\{MBint[((-t)^{-3-eps}\, t\, (-x)^{z1}\, y^{-1-eps-z1}\, Gamma[-eps-z1]^2\, Gamma[-z1]\, Gamma[1+eps+z1]\, (EulerGamma + Log[y] + 2\, PolyGamma[0, -2\, eps] + 2\, PolyGamma[0, -eps-z1] - 2\, PolyGamma[0, -2\, (eps+z1)] - PolyGamma[0, 1+eps+z1])) / Gamma[-2\, (eps+z1)], \{\{eps \to 0\}, \{z1 \to -\frac{1}{2}\}\}],$$
$$MBint[((-t)^{-2-eps}\, (-x)^{z1}\, y^{z2}\, Gamma[-z1]\, Gamma[-1-eps-z1-z2]^2\, Gamma[-z2]\, Gamma[1+z2]^2\, Gamma[2\, (1+z1+z2)]\, Gamma[2+eps+z1+z2]) / (Gamma[-2\, eps]\, Gamma[2+2\, z2]), \{\{eps \to 0\}, \{z1 \to -\frac{1}{2}, z2 \to -\frac{1}{4}\}\}]\}$$

(* next line, B4l2mMBrepres,
is the general MB-representation at arbitrary, small eps for B4l2m *)

B4l2mMBrepres = integrals

$$\{MBint[((-t)^{-3-eps}\, t\, (-x)^{z1}\, y^{-1-eps-z1}\, Gamma[-eps-z1]^2\, Gamma[-z1]\, Gamma[1+eps+z1]\, (EulerGamma + Log[y] + 2\, PolyGamma[0, -2\, eps] + 2\, PolyGamma[0, -eps-z1] - 2\, PolyGamma[0, -2\, (eps+z1)] - PolyGamma[0, 1+eps+z1])) / Gamma[-2\, (eps+z1)], \{\{eps \to 0\}, \{z1 \to -\frac{1}{2}\}\}],$$
$$MBint[((-t)^{-2-eps}\, (-x)^{z1}\, y^{z2}\, Gamma[-z1]\, Gamma[-1-eps-z1-z2]^2\, Gamma[-z2]\, Gamma[1+z2]^2\, Gamma[2\, (1+z1+z2)]\, Gamma[2+eps+z1+z2]) / (Gamma[-2\, eps]\, Gamma[2+2\, z2]), \{\{eps \to 0\}, \{z1 \to -\frac{1}{2}, z2 \to -\frac{1}{4}\}\}]\}$$

? MBexpand

MBexpand[integrals, norm, {x, x0, n}] expands integrals in
the specified order n of expansion in x around x0. A normalization factor norm
is applied to every integrand.

```
B4l2mMBexpand =
 MBexpand[B4l2mMBrepres, Exp[eps EulerGamma], {eps, 0, 1}] // MBmerge
```

$$\left\{\text{MBint}\left[\left((-x)^{z1}\, y^{-1-z1}\, \text{Gamma}[-z1]^3\, \text{Gamma}[1+z1]\left(-6+3\,\text{eps}^2\,\text{EulerGamma}^2+4\,\text{eps}^2\,\pi^2+6\,\text{eps}\,\text{Log}[-t]-3\,\text{eps}^2\,\text{Log}[-t]^2-6\,\text{eps}^2\,\text{EulerGamma}\,\text{Log}[y]+3\,\text{eps}^2\,\text{Log}[y]^2+12\,\text{eps}^2\,\text{PolyGamma}[0,-2\,z1]^2+12\,\text{eps}^2\,\text{PolyGamma}[0,-z1]^2+6\,\text{eps}^2\,\text{EulerGamma}\,\text{PolyGamma}[0,1+z1]-6\,\text{eps}^2\,\text{Log}[y]\,\text{PolyGamma}[0,1+z1]+3\,\text{eps}^2\,\text{PolyGamma}[0,1+z1]^2-12\,\text{eps}^2\,\text{PolyGamma}[0,-z1]\,(\text{EulerGamma}-\text{Log}[y]+\text{PolyGamma}[0,1+z1])+12\,\text{eps}^2\,\text{PolyGamma}[0,-2\,z1]\,(\text{EulerGamma}-\text{Log}[y]-2\,\text{PolyGamma}[0,-z1]+\text{PolyGamma}[0,1+z1])-12\,\text{eps}^2\,\text{PolyGamma}[1,-2\,z1]+6\,\text{eps}^2\,\text{PolyGamma}[1,-z1]+3\,\text{eps}^2\,\text{PolyGamma}[1,1+z1]\right)\right)\Big/\left(6\,\text{eps}\,t^2\,\text{Gamma}[-2\,z1]\right),\left\{\{\text{eps}\to 0\},\left\{z1\to -\frac{1}{2}\right\}\right\}\right],$$

$$\text{MBint}\left[-\frac{1}{t^2\,\text{Gamma}[2+2\,z2]}\,2\,\text{eps}\,(-x)^{z1}\,y^{z2}\,\text{Gamma}[-z1]\,\text{Gamma}[-1-z1-z2]^2\,\text{Gamma}[-z2]\,\text{Gamma}[1+z2]^2\,\text{Gamma}[2\,(1+z1+z2)]\,\text{Gamma}[2+z1+z2],\left\{\{\text{eps}\to 0\},\left\{z1\to -\frac{1}{2},\,z2\to -\frac{1}{4}\right\}\right\}\right]\right\}$$

```
? ToString
```

ToString[*expr*] gives a string corresponding to the printed form of *expr* in OutputForm.
ToString[*expr*, *form*] gives the string corresponding to output in the specified form. ≫

```
ToString[B4l2mMBexpand, InputForm, PageWidth → 60]
```

```
{MBint[((-x)^z1*y^(-1 - z1)*Gamma[-z1]^3*Gamma[1 + z1]*
    (-6 + 3*eps^2*EulerGamma^2 + 4*eps^2*Pi^2 +
     6*eps*Log[-t] - 3*eps^2*Log[-t]^2 -
     6*eps^2*EulerGamma*Log[y] + 3*eps^2*Log[y]^2 +
     12*eps^2*PolyGamma[0, -2*z1]^2 +
     12*eps^2*PolyGamma[0, -z1]^2 + 6*eps^2*EulerGamma*
      PolyGamma[0, 1 + z1] - 6*eps^2*Log[y]*
      PolyGamma[0, 1 + z1] + 3*eps^2*PolyGamma[0, 1 + z1]^
       2 - 12*eps^2*PolyGamma[0, -z1]*(EulerGamma -
       Log[y] + PolyGamma[0, 1 + z1]) +
     12*eps^2*PolyGamma[0, -2*z1]*(EulerGamma - Log[y] -
       2*PolyGamma[0, -z1] + PolyGamma[0, 1 + z1]) -
     12*eps^2*PolyGamma[1, -2*z1] +
     6*eps^2*PolyGamma[1, -z1] +
     3*eps^2*PolyGamma[1, 1 + z1]))/
   (6*eps*t^2*Gamma[-2*z1]), {{eps -> 0}, {z1 -> -1/2}}],
 MBint[(-2*eps*(-x)^z1*y^z2*Gamma[-z1]*
    Gamma[-1 - z1 - z2]^2*Gamma[-z2]*Gamma[1 + z2]^2*
    Gamma[2*(1 + z1 + z2)]*Gamma[2 + z1 + z2])/
   (t^2*Gamma[2 + 2*z2]), {{eps -> 0},
   {z1 -> -1/2, z2 -> -1/4}}]}
```

```
Lk = {x → -1 / 10, y → 1 / 50, t → -10};
```

MBintegrate[B4l2mMBexpand, Lk]

```
Shifting contours...
```

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {−0.5}. ≫

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {−0.5}. ≫

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

General::stop : Further output of Power::infy will be suppressed during this calculation. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {−0.5}. ≫

General::stop : Further output of FindMinimum::nrgnum will be suppressed during this calculation. ≫

ReplaceAll::reps : $\left\{z1, -\frac{1}{2}\right\}$ is neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing. ≫

```
Minimum search failed
```

$$\text{MBint}\Big[\frac{1}{12\,\text{Gamma}[-2\,z1]}\,5^{z1}\,\text{Gamma}[-z1]^3\,\text{Gamma}[1+z1]\,\big(3\,\text{EulerGamma}^2+4\,\pi^2-3\,\text{Log}[10]^2+6\,\text{EulerGamma}\,\text{Log}[50]+3\,\text{Log}[50]^2+12\,\text{PolyGamma}[0,-2\,z1]^2+12\,\text{PolyGamma}[0,-z1]^2+6\,\text{EulerGamma}\,\text{PolyGamma}[0,1+z1]+6\,\text{Log}[50]\,\text{PolyGamma}[0,1+z1]+3\,\text{PolyGamma}[0,1+z1]^2-12\,\text{PolyGamma}[0,-z1]\,(\text{EulerGamma}+\text{Log}[50]+\text{PolyGamma}[0,1+z1])+12\,\text{PolyGamma}[0,-2\,z1]\,(\text{EulerGamma}+\text{Log}[50]-2\,\text{PolyGamma}[0,-z1]+\text{PolyGamma}[0,1+z1])-12\,\text{PolyGamma}[1,-2\,z1]+6\,\text{PolyGamma}[1,-z1]+3\,\text{PolyGamma}[1,1+z1]\big),\,\{\{eps\to 0\},\,\{z1\to -\tfrac{1}{2}\}\}\Big]$$

```
Performing 3 lower-dimensional integrations with NIntegrate...1...2...3
Higher-dimensional integrals
Preparing MBpart1eps1 (dim 2)
Running MBpart1eps1
```

$$\left\{0.2228776387609497-\frac{0.0967945286535068}{eps}+0.617597\,eps,\ \left\{8.79914\times 10^{-6}\,eps,\ 0\right\}\right\}$$

Derivation of sums

<< MBsums.v1.0.m

```
MBsums v1.0 by Michal Ochman
The author would like to thank Tord Riemann
for many fruitful discussions
```

```
dim1int = B4l2mMBexpand[[1]]
```

```
MBint[((-x)^z1 y^(-1-z1) Gamma[-z1]^3 Gamma[1 + z1]
   (-6 + 3 eps^2 EulerGamma^2 + 4 eps^2 π^2 + 6 eps Log[-t] - 3 eps^2 Log[-t]^2 -
     6 eps^2 EulerGamma Log[y] + 3 eps^2 Log[y]^2 + 12 eps^2 PolyGamma[0, -2 z1]^2 +
     12 eps^2 PolyGamma[0, -z1]^2 + 6 eps^2 EulerGamma PolyGamma[0, 1 + z1] -
     6 eps^2 Log[y] PolyGamma[0, 1 + z1] + 3 eps^2 PolyGamma[0, 1 + z1]^2 -
     12 eps^2 PolyGamma[0, -z1] (EulerGamma - Log[y] + PolyGamma[0, 1 + z1]) +
     12 eps^2 PolyGamma[0, -2 z1]
      (EulerGamma - Log[y] - 2 PolyGamma[0, -z1] + PolyGamma[0, 1 + z1]) - 12 eps^2
      PolyGamma[1, -2 z1] + 6 eps^2 PolyGamma[1, -z1] + 3 eps^2 PolyGamma[1, 1 + z1])) /
  (6 eps t^2 Gamma[-2 z1]), {{eps → 0}, {z1 → -1/2}}]
```

```
dim1sum = MBIntToSum[dim1int, Lk, {z1 → L}]
```

```
z1->L ( Re z1 < -1/2 )
```

```
{MBsum[-((-1)^-n1 (-x)^-n1 y^n1 (n1!)^2
     (-4 + 3 eps^2 π^2 + 2 eps^2 HarmonicNumber[n1]^2 - 2 eps^2 HarmonicNumber[n1, 2] +
       4 eps Log[-t] - 2 eps^2 Log[-t]^2 - 4 eps^2 HarmonicNumber[n1] Log[-x] +
       2 eps^2 Log[-x]^2)) / (4 eps t^2 x (1 + 2 n1)!), n1 ≥ 0, {n1}]}
```

```
ToString[dim1sum, InputForm, PageWidth → 60]
```

```
{MBsum[-(y^n1*n1!^2*(-4 + 3*eps^2*Pi^2 +
       2*eps^2*HarmonicNumber[n1]^2 -
       2*eps^2*HarmonicNumber[n1, 2] + 4*eps*Log[-t] -
       2*eps^2*Log[-t]^2 - 4*eps^2*HarmonicNumber[n1]*
        Log[-x] + 2*eps^2*Log[-x]^2))/(4*(-1)^n1*eps*t^2*
     (-x)^n1*x*(1 + 2*n1)!), n1 >= 0, {n1}]}
```

```
num1 = MBintegrate[{dim1int}, Lk] // N
```

```
Shifting contours...
```

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {-0.5}. ≫

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {-0.5}. ≫

Power::infy : Infinite expression $\frac{1}{0.^2}$ encountered. ≫

General::stop : Further output of Power::infy will be suppressed during this calculation. ≫

FindMinimum::nrgnum : The gradient is not a vector of real numbers at {z1} = {-0.5}. ≫

General::stop : Further output of FindMinimum::nrgnum will be suppressed during this calculation. ≫

ReplaceAll::reps : $\{z1, -\frac{1}{2}\}$ is neither a list of replacement rules nor a valid dispatch table, and so cannot be used for replacing. ≫

```
Minimum search failed
```

$$\text{MBint}\Big[\frac{1}{12\,\text{Gamma}[-2\,z1]}\,5^{z1}\,\text{Gamma}[-z1]^3\,\text{Gamma}[1+z1]\,\big(3\,\text{EulerGamma}^2+4\,\pi^2-3\,\text{Log}[10]^2+6\,\text{EulerGamma}\,\text{Log}[50]+3\,\text{Log}[50]^2+12\,\text{PolyGamma}[0,-2\,z1]^2+12\,\text{PolyGamma}[0,-z1]^2+6\,\text{EulerGamma}\,\text{PolyGamma}[0,1+z1]+6\,\text{Log}[50]\,\text{PolyGamma}[0,1+z1]+3\,\text{PolyGamma}[0,1+z1]^2-12\,\text{PolyGamma}[0,-z1]\,(\text{EulerGamma}+\text{Log}[50]+\text{PolyGamma}[0,1+z1])+12\,\text{PolyGamma}[0,-2\,z1]\,(\text{EulerGamma}+\text{Log}[50]-2\,\text{PolyGamma}[0,-z1]+\text{PolyGamma}[0,1+z1])-12\,\text{PolyGamma}[1,-2\,z1]+6\,\text{PolyGamma}[1,-z1]+3\,\text{PolyGamma}[1,1+z1]\big),\,\Big\{\{\text{eps}\to 0\},\,\Big\{z1\to -\frac{1}{2}\Big\}\Big\}\Big]$$

```
Performing 3 lower-dimensional integrations with NIntegrate...1...2...3
Higher-dimensional integrals
```

$$\Big\{0.222878-\frac{0.0967945}{\text{eps}}+0.709316\,\text{eps},\,0.\Big\}$$

num2 = DoAllMBSums[dim1sum, 50, Lk] // N

$$0.222878-\frac{0.0967945}{\text{eps}}+0.709316\,\text{eps}$$

num1[[1]] - num2

$$1.36427\times 10^{-11}-\frac{5.92497\times 10^{-12}}{\text{eps}}+1.11022\times 10^{-16}\,\text{eps}$$

dim2int = B4l2mMBexpand[[2]]

$$\text{MBint}\Big[-\frac{1}{t^2\,\text{Gamma}[2+2\,z2]}\,2\,\text{eps}\,(-x)^{z1}\,y^{z2}\,\text{Gamma}[-z1]\,\text{Gamma}[-1-z1-z2]^2\,\text{Gamma}[-z2]\,\text{Gamma}[1+z2]^2\,\text{Gamma}[2\,(1+z1+z2)]\,\text{Gamma}[2+z1+z2],\,\Big\{\{\text{eps}\to 0\},\,\Big\{z1\to -\frac{1}{2},\,z2\to -\frac{1}{4}\Big\}\Big\}\Big]$$

dim2sum = MBIntToSum[dim2int, Lk, {z2 → L, z1 → L}]

```
z2->R ( Re z2 > -1/4 )
Unable to find correct contour for z1.
{}
```

dim2sum = MBIntToSum[dim2int, {t → -10}, {z1 → L, z2 → L}]

```
Found c = -x (not a number): please complete kinematic's list.
Unable to find correct contour for z1.
{}
```

dim2sum = MBIntToSum[dim2int, Lk, {z1 → L, z2 → L}]

```
z1->R ( Re z1 > -1/2 )

z2->R ( Re z2 > -1/4 )

{MBsum[-((-1)^(-3 n1) eps (-x)^(n1-n2) y^n2 (-1 + 2 n1) !
      n2 ! (π^2 + 6 HarmonicNumber[n1] HarmonicNumber[-1 + n1 - n2] -
        12 HarmonicNumber[-1 + 2 n1] HarmonicNumber[-1 + n1 - n2] +
        3 HarmonicNumber[-1 + n1 - n2]^2 + 6 HarmonicNumber[n1] HarmonicNumber[n2] -
        12 HarmonicNumber[-1 + 2 n1] HarmonicNumber[n2] -
        3 HarmonicNumber[n2]^2 - 12 HarmonicNumber[n1] HarmonicNumber[1 + 2 n2] +
        24 HarmonicNumber[-1 + 2 n1] HarmonicNumber[1 + 2 n2] +
        12 HarmonicNumber[n2] HarmonicNumber[1 + 2 n2] -
        12 HarmonicNumber[1 + 2 n2]^2 + 3 HarmonicNumber[-1 + n1 - n2, 2] +
        3 HarmonicNumber[n2, 2] - 12 HarmonicNumber[1 + 2 n2, 2] -
        6 HarmonicNumber[n1] Log[-x] + 12 HarmonicNumber[-1 + 2 n1] Log[-x] -
        6 HarmonicNumber[-1 + n1 - n2] Log[-x] + 3 Log[-x]^2 + 6 HarmonicNumber[n1]
         Log[y] - 12 HarmonicNumber[-1 + 2 n1] Log[y] - 6 HarmonicNumber[n2] Log[y] +
        12 HarmonicNumber[1 + 2 n2] Log[y] - 3 Log[y]^2)) /
    (3 t^2 x n1 ! (-1 + n1 - n2) ! (1 + 2 n2) !), 2 n1 ≥
    1 && n2 ≥
    0 && n1 ≥
    1 + n2, {n1, n2}],
 MBsum[(2 (-1)^(-2 n1-n2) eps (-x)^(n1-n2) y^n2 (-1 + 2 n1) ! n2 ! (-n1 + n2) !
     (HarmonicNumber[n1] - 2 HarmonicNumber[-1 + 2 n1] +
       HarmonicNumber[-n1 + n2] - Log[-x])) /
   (t^2 x n1 ! (1 + 2 n2) !), 2 n1 ≥ 1 && n2 ≥ 0 &&
   n1 ≤ n2, {n1,
   n2}],
 MBsum[((-1)^(-3 n2) eps (-x)^n1 y^(-1-n1+n2) (-1 - n1 + n2) ! (-1 + 2 n2) !
     (π^2 + HarmonicNumber[n2]^2 - 2 HarmonicNumber[n2] HarmonicNumber[-1 - n1 + n2] +
       HarmonicNumber[-1 - n1 + n2]^2 - 4 HarmonicNumber[n2]
        HarmonicNumber[-1 + 2 n2] + 4 HarmonicNumber[-1 - n1 + n2]
        HarmonicNumber[-1 + 2 n2] + 4 HarmonicNumber[-1 + 2 n2]^2 +
       4 HarmonicNumber[n2] HarmonicNumber[-1 - 2 n1 + 2 n2] -
       4 HarmonicNumber[-1 - n1 + n2] HarmonicNumber[-1 - 2 n1 + 2 n2] -
       8 HarmonicNumber[-1 + 2 n2] HarmonicNumber[-1 - 2 n1 + 2 n2] +
       4 HarmonicNumber[-1 - 2 n1 + 2 n2]^2 + HarmonicNumber[n2, 2] -
       HarmonicNumber[-1 - n1 + n2, 2] - 4 HarmonicNumber[-1 + 2 n2, 2] +
       4 HarmonicNumber[-1 - 2 n1 + 2 n2, 2] - 2 HarmonicNumber[n2] Log[y] +
       2 HarmonicNumber[-1 - n1 + n2] Log[y] + 4 HarmonicNumber[-1 + 2 n2] Log[y] -
       4 HarmonicNumber[-1 - 2 n1 + 2 n2] Log[y] + Log[y]^2)) /
   (t^2 n1 ! n2 ! (-1 - 2 n1 + 2 n2) !), n1 ≥ 0 &&
   2 n2 ≥ 1 &&
   1 + n1 ≤ n2, {n1,
   n2}]}
```

MBintegrate[{dim2int}, Lk]

```
Shifting contours...
Performing 0 lower-dimensional integrations with NIntegrate
Higher-dimensional integrals
Preparing MBpart1eps1 (dim 2)
Running MBpart1eps1
```

$\{-0.0917189\ \text{eps}, \{8.79914\times 10^{-6}\ \text{eps}, 0\}\}$

DoAllMBSums[dim2sum, 50, Lk] // N

$-0.0917188\ \text{eps}$

ToString[dim2sum, InputForm, PageWidth → 60]

```
{MBsum[-(eps*(-x)^(n1 - n2)*y^n2*(-1 + 2*n1)!*n2!*
     (Pi^2 + 6*HarmonicNumber[n1]*HarmonicNumber[
        -1 + n1 - n2] - 12*HarmonicNumber[-1 + 2*n1]*
       HarmonicNumber[-1 + n1 - n2] +
      3*HarmonicNumber[-1 + n1 - n2]^2 +
      6*HarmonicNumber[n1]*HarmonicNumber[n2] -
      12*HarmonicNumber[-1 + 2*n1]*HarmonicNumber[n2] -
      3*HarmonicNumber[n2]^2 - 12*HarmonicNumber[n1]*
       HarmonicNumber[1 + 2*n2] +
      24*HarmonicNumber[-1 + 2*n1]*HarmonicNumber[
        1 + 2*n2] + 12*HarmonicNumber[n2]*
       HarmonicNumber[1 + 2*n2] -
      12*HarmonicNumber[1 + 2*n2]^2 +
      3*HarmonicNumber[-1 + n1 - n2, 2] +
      3*HarmonicNumber[n2, 2] - 12*HarmonicNumber[1 + 2*n2,
        2] - 6*HarmonicNumber[n1]*Log[-x] +
      12*HarmonicNumber[-1 + 2*n1]*Log[-x] -
      6*HarmonicNumber[-1 + n1 - n2]*Log[-x] +
      3*Log[-x]^2 + 6*HarmonicNumber[n1]*Log[y] -
      12*HarmonicNumber[-1 + 2*n1]*Log[y] -
      6*HarmonicNumber[n2]*Log[y] +
      12*HarmonicNumber[1 + 2*n2]*Log[y] - 3*Log[y]^2))/
   (3*(-1)^(3*n1)*t^2*x*n1!*(-1 + n1 - n2)!*(1 + 2*n2)!),
  2*n1 >= 1 && n2 >= 0 && n1 >= 1 + n2, {n1, n2}],
 MBsum[(2*(-1)^(-2*n1 - n2)*eps*(-x)^(n1 - n2)*y^n2*
    (-1 + 2*n1)!*n2!*(-n1 + n2)!*(HarmonicNumber[n1] -
     2*HarmonicNumber[-1 + 2*n1] + HarmonicNumber[
      -n1 + n2] - Log[-x]))/(t^2*x*n1!*(1 + 2*n2)!),
  2*n1 >= 1 && n2 >= 0 && n1 <= n2, {n1, n2}],
 MBsum[(eps*(-x)^n1*y^(-1 - n1 + n2)*(-1 - n1 + n2)!*
    (-1 + 2*n2)!*(Pi^2 + HarmonicNumber[n2]^2 -
     2*HarmonicNumber[n2]*HarmonicNumber[-1 - n1 + n2] +
     HarmonicNumber[-1 - n1 + n2]^2 - 4*HarmonicNumber[n2]*
      HarmonicNumber[-1 + 2*n2] +
     4*HarmonicNumber[-1 - n1 + n2]*HarmonicNumber[
       -1 + 2*n2] + 4*HarmonicNumber[-1 + 2*n2]^2 +
     4*HarmonicNumber[n2]*HarmonicNumber[
       -1 - 2*n1 + 2*n2] - 4*HarmonicNumber[-1 - n1 + n2]*
      HarmonicNumber[-1 - 2*n1 + 2*n2] -
     8*HarmonicNumber[-1 + 2*n2]*HarmonicNumber[
       -1 - 2*n1 + 2*n2] +
     4*HarmonicNumber[-1 - 2*n1 + 2*n2]^2 +
     HarmonicNumber[n2, 2] - HarmonicNumber[-1 - n1 + n2,
      2] - 4*HarmonicNumber[-1 + 2*n2, 2] +
     4*HarmonicNumber[-1 - 2*n1 + 2*n2, 2] -
     2*HarmonicNumber[n2]*Log[y] +
     2*HarmonicNumber[-1 - n1 + n2]*Log[y] +
     4*HarmonicNumber[-1 + 2*n2]*Log[y] -
     4*HarmonicNumber[-1 - 2*n1 + 2*n2]*Log[y] + Log[y]^2))/
   ((-1)^(3*n2)*t^2*n1!*n2!*(-1 - 2*n1 + 2*n2)!),
  n1 >= 0 && 2*n2 >= 1 && 1 + n1 <= n2, {n1, n2}]}
```