```
/**********************************************************************************
 * Package: GSM                                                          *
 * Class  : WZFitter                                                     *
 *                                                                       *
 * Description:                                                          *
 *      Auxiliary Theory for electroweak formfactor for W decay          *
 *      W Boson decay / total width                                      *
 *                                                                       *
 * Sources:                                                              *
 *      - Bardin, Degrassi, The Standard Model in the Making, Oxford 1999       *
 *      - Bardin et al., ZFitter package dizet6_42.f                     *
 *                                                                       *
 * This class also contains code lines ported to C++ from the Fortran package     *
 * ZFITTER                                                               *
 *                                                                       *
 **********************************************************************************/
#include "TMath.h"

#include "Gfitter/GMath.h"
#include "Gfitter/GConstants.h"
#include "Gfitter/GTheory.h"
#include "Gfitter/GTheoryRef.h"
#include "Gfitter/GParameterRef.h"
#include "Gfitter/GReference.h"

#include "GSM/WZFitter.h"
#include "GSM/FermionPart.h"
#include "GSM/BosonPart.h"
#include "GSM/Vertex.h"

using namespace Gfitter;
using std::complex;

ClassImp(GSM::WZFitter)

GSM::WZFitter::WZFitter()
   : WBase(),
     m_isUpToDate_Update( kFALSE )
{
   SetTheoryName( GetName() );
   SetExistDerivative( kFALSE );

```

Hinweis:

Kommentare mit Hinweisen auf ZFitter sind grün markiert

Übereinstimmungen sind gelb markiert.

Auffällige Stellen bzw. Anmerkungen sind violett markiert

Anhang 2

zum Gutachten DESY ZFitter_GFitter vom 17.03.2014

Lübeck, den 17. März 2014

Dr. Ulrich Obermöller
-Sachverständiger-

```cpp
   BookTheory   ( "GSM::FermionPart"            , &t_fermionPart );
   BookTheory   ( "GSM::BosonPart"              , &t_bosonPart );
   BookTheory   ( "GSM::Vertex"                 , &t_Vertex );
   BookTheory   ( "GSM::RadiatorFunctions"      , &t_RadFun );
   BookTheory   ( "GSM::WMass"                  , & t_MW );
}

void GSM::WZFitter::UpdateLocalFlags( GReference& /* ref */ )
{
   m_isUpToDate_Update = kFALSE;
}

void GSM::WZFitter::Update()
{
   if (m_isUpToDate_Update) return;

   // now, it is uptodate (I mean... it will be)
   m_isUpToDate_Update = kTRUE;

   Double_t MW  = GetMW();
   Double_t MZ  = p_MZ;
   Double_t MW2 = MW*MW;
   Double_t MZ2 = MZ*MZ;

   m_R          = MW2/MZ2;

   // bosonic + ferminoic contribution
   // see dizet6_42.f line 3408-3410
   m_W_MW  = real( GetBosonPart().GetWbAtMW()  + GetFermionPart().GetWfAtMW() );
   m_W0    = real( GetBosonPart().GetWbAt0()   + GetFermionPart().GetWfAt0() );
   m_WF_MW = real( GetBosonPart().GetWbFAtMW() + GetFermionPart().GetWfFAtMW() );

   // eq. (10.67) of The Standard Model in the Making
   // without mass-corrections,
   // they are neglibile in relation to the W mass
    // see dizet6_42.f line 2849
   Double_t Gamma0 = ( p_GF*Gfitter::GMath::IPow( GetMW(),3 )
                     / (6.0*TMath::Pi()*Gfitter::GMath::Sqrt2()) );

   // decay into leptons
   Double_t RhoLep = GetRhoW( 1 );
```

Match 1

```fortran
3396        SUBROUTINE PROW (QI,ROW)
3397  *
3398        IMPLICIT REAL*8(A-H,O-W,Y-Z)
3399        IMPLICIT COMPLEX*16(X)
3400        COMMON/CDZCON/PI,PI2,F1,D3,ALFAI,AL4PI,AL2PI,AL1PI
3401        COMMON/CDZWSM/AMW2,AMZ2,R,R1,R12,R2,AMH2,RW,RW1,RW12,RW2,RZ,RZ1,
3402       *       RZ12,RZ2,ALR,ALRW,ALRZ,SW2M,CW2M,AKSX,R1W,R1W2
3403        COMMON/CDZWSC/SL2,SQ2,W0,W0F,Z0,Z0F,DWZ0R1,DWZ0F,XWM1,XWM1F,XZM1,
3404       &       XZM1F,XWZ1R1,XDWZ1F,XZFM1,XZFM1F,XAMM1,XAMM1F,XWFM1,XWFM1F
3405        COMMON/CDZVZW/V1ZZ,V1ZW,V2ZWW,V1WZ,V2WWZ,VTB
3406  *
3407        QIQJ=QI*(1.D0-QI)
3408        WM1A=DREAL(XWM1+XWM1F)
3409        W0A=W0+W0F
3410        WFM1A=DREAL(XWFM1+XWFM1F)
3411        ROW=1.D0+AL4PI/R1*(WM1A-W0A+WFM1A-7.D0/1.D0+5.D0/8.D0*R*R1W
3412       *   -9.D0/4.D0*R/R1*ALR+3.D0/4.D0/R+3.D0*R-3.D0/R*R12*QIQJ
3413       *   +(1.D0/2.D0/R-1.D0-2.D0*R12/R*QIQJ)*V1WZ
3414       *   +2.D0*R*V2WWZ+2.D0*R1*(77.D0/12.D0-2.D0/3.D0*PI2+109.D0/36.D0
3415       *   -3.D0/2.D0*QIQJ))
3416        PROW1=100.D0*(ROW-1.D0)
3417  *
3418        END
```

Match 2

```fortran
2842 ***********************************************************************
2843 *
2844 * W- CHAIN STARTS HERE, IT IS QUITE SIMILAR TO Z- CHAIN, FOR THIS
2845 * REASON ONLY BRIEF ADDITIONAL COMMENTS ARE ADDED BELOW
2846 *
2847       CALL VERTZW(0,0,1D0)
2848       AMW=SQRT(AMW2)
2849       CONSTW=GMU*AMW**3/6.D0/PI/SQRT(2.D0)
2850       GAM0T=0.D0
2851       GAM1T=0.D0
2852       DO 7 IND=1,2
2853       CALL PROW (AQFW(IND),ROW)
2854 * THIS SUBROUTINE RETURNS THE ONLY ONE ELECTROWEAK FORMFACTOR ROW
2855 * EXISTING IN THE W- DECAY CASE.
2856 * THE OTHER IMPORTANT DIFFERENCE FROM Z- CASE IS THAT IT IS IMPOSSIBLE T
2857 * DEFINE HERE QED- GAUGE INVARIANT SUBSET OF DIAGRAMS, FOR THIS REASON
2858 * ONLY TOTAL 1-LOOP GAMMAS AND WIDTHS ARE CALCULATED FOR W- DECAY
```

```cpp
      // decay into hadrons
      Double_t RhoQua = GetRhoW( Gfitter::GMath::TwoThird() );
      // some QCD corrections
      Double_t RadVu  = GetRadFun().GetRVq( Gfitter::GTypes::kUp, Gfitter::GMath::TwoThird() );

      // see dizet6_42.f line 2860-2869
      m_gammaW = Gamma0*( RhoLep*3.0 + RhoQua*6.0*RadVu );||~ keine Übereinstimmungen

      // now, parameters are up-to-date
      SetUpToDate();
   }

   // rho for W decay, just 1-loop, because it is impossible
   // to define QED-gauge invariant subset of diagrams
   Double_t GSM::WZFitter::GetRhoW( Double_t Charge )
   {
      Update();
      Double_t ChUpDo = Charge*(1.0 - Charge);

      // for better understanding
      // eq. (10.71) of The Standard Model in the Making
      // one and two point functions are replaced
      // by fermionic and bosonic contribution

      // see dizet6_42.f line 3411
      Double_t rho    = ( 1.0 + GConstants::alphaQED()/(4.0*TMath::Pi()*(1.0-m_R))
                      *(m_W_MW - m_W0 + m_WF_MW - 7.0 + 5/8.0*m_R*(1.0+m_R)
                       - 9/4.0*m_R/(1.0-m_R)*TMath::Log(m_R)
                       + 0.75/m_R + 3.0*m_R - 3.0/m_R*(1.0-m_R)*(1.0-m_R)*ChUpDo
                       + (0.5/m_R - 1.0 - 2.0*(1.0-m_R)*(1.0-m_R)/m_R*ChUpDo)*GetVertex().GetV1WZ()
                       + 2.0*m_R*GetVertex().GetV2WWZ() + 2.0*(1.0-m_R)
                       *(77/12.0 - 2/3.0*GMath::IPow( TMath::Pi(),2 ) + 109/36.0 - 3/2.0*ChUpDo)) );
      return rho;
   }
```

Match 3

Match 4

```fortran
2854  * THIS SUBROUTINE RETURNS THE ONLY ONE ELECTROWEAK FORMFACTOR ROW
2855  * EXISTING IN THE W- DECAY CASE.
2856  * THE OTHER IMPORTANT DIFFERENCE FROM Z- CASE IS THAT IT IS IMPOSSIBLE T
2857  * DEFINE HERE QED- GAUGE INVARIANT SUBSET OF DIAGRAMS, FOR THIS REASON
2858  * ONLY TOTAL 1-LOOP GAMMAS AND WIDTHS ARE CALCULATED FOR W- DECAY
2859  *
2860        GAM0I=CONSTW
2861        GAM1I=CONSTW*ROW*QCDCOR(IND-1)
2862        DELT1I=(GAM1I/GAM0I-1.D0)*100.D0
2863        GAM0T=GAM0T+GAM0I*MWFAC(IND)
2864        GAM1T=GAM1T+GAM1I*MWFAC(IND)
2865        PARTW(IND)= GAM1I*MWFAC(IND)*1D3
2866  7     CONTINUE
2867        DELT1T=(GAM1T/GAM0T-1.D0)*100.D0
2868        GAMW = GAM1T
2869        PARTW(3)=GAM1T*1D3
2870  *
2871        END
```

```fortran
3406  *
3407        QIQJ=QI*(1.D0-QI)
3408        WM1A=DREAL(XWM1+XWM1F)
3409        W0A=W0+W0F
3410        WFM1A=DREAL(XWFM1+XWFM1F)
3411        ROW=1.D0+AL4PI/R1*(WM1A-W0A+WFM1A-7.D0/1.D0+5.D0/8.D0*R*R1W
3412       *   -9.D0/4.D0*R/R1*ALR+3.D0/4.D0/R+3.D0*R-3.D0/R*R12*QIQJ
3413       *   +(1.D0/2.D0/R-1.D0-2.D0*R12/R*QIQJ)*V1WZ
3414       *   +2.D0*R*V2WWZ+2.D0*R1*(77.D0/12.D0-2.D0/3.D0*PI2+109.D0/36.D0
3415       *   -3.D0/2.D0*QIQJ))
3416        PROW1=100.D0*(ROW-1.D0)
3417  *
3418        END
```