The new MC-2020 250 GeV data sets - and why you should use them

part 1 - samples, beam parameters & physics

Mikael Berggren[1]

[1]DESY, Hamburg

ILD general meeting, December 8, 2020

Outline

1. Why: MC requirements for linear colliders
2. How: Generating the full SM
   - Process classification
   - Physics generator
   - Generating beam properties
3. What: ILC 250 GeV Generation production
   - Whizard 1.95 $\rightarrow$ 2.8.5
   - Beams
   - Generation status
4. Conclusions

MC requirements for linear colliders

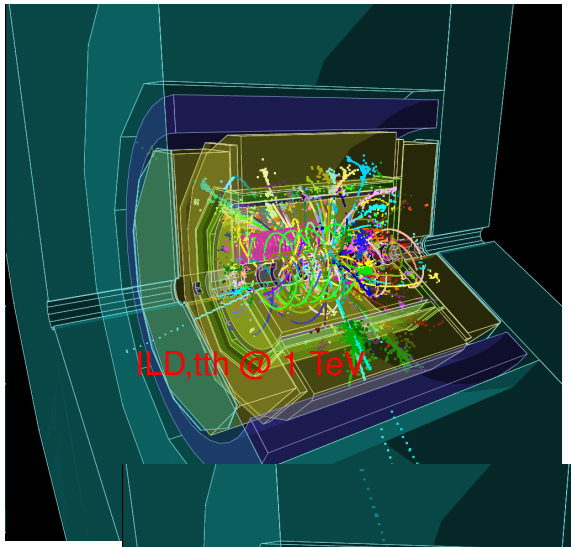
- Future LCs aim for extremely high precision measurements.
  - $\Rightarrow$ Need excellent detector, well controlled machine conditions
  - But also the best possible estimate of backgrounds.
- So: MC statistics or lacking channels must not be a major source of systematic errors $\Rightarrow$
  - All SM channels yielding at least a few events under the full lifetime of the projects need to be generated, with statistics largely exceeding that of the real data.
  - Also machine conditions need to be accurately taken into account.

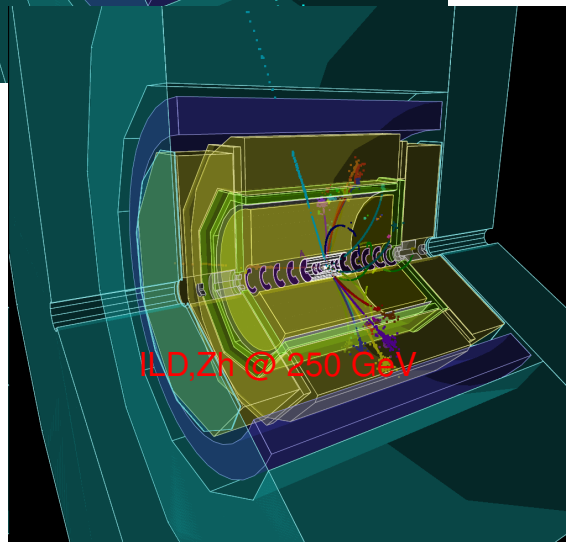
MC requirements for linear colliders

- Future LCs aim for extremely high precision measurements.
  - $\Rightarrow$ Need excellent detector, well controlled machine conditions
  - But also the best possible estimate of backgrounds.
- So: MC statistics or lacking channels must not be a major source of systematic errors $\Rightarrow$
  - All SM channels yielding at least a few events under the full lifetime of the projects need to be generated, with statistics largely exceeding that of the real data.
  - Also machine conditions need to be accurately taken into account.

Generating the full SM

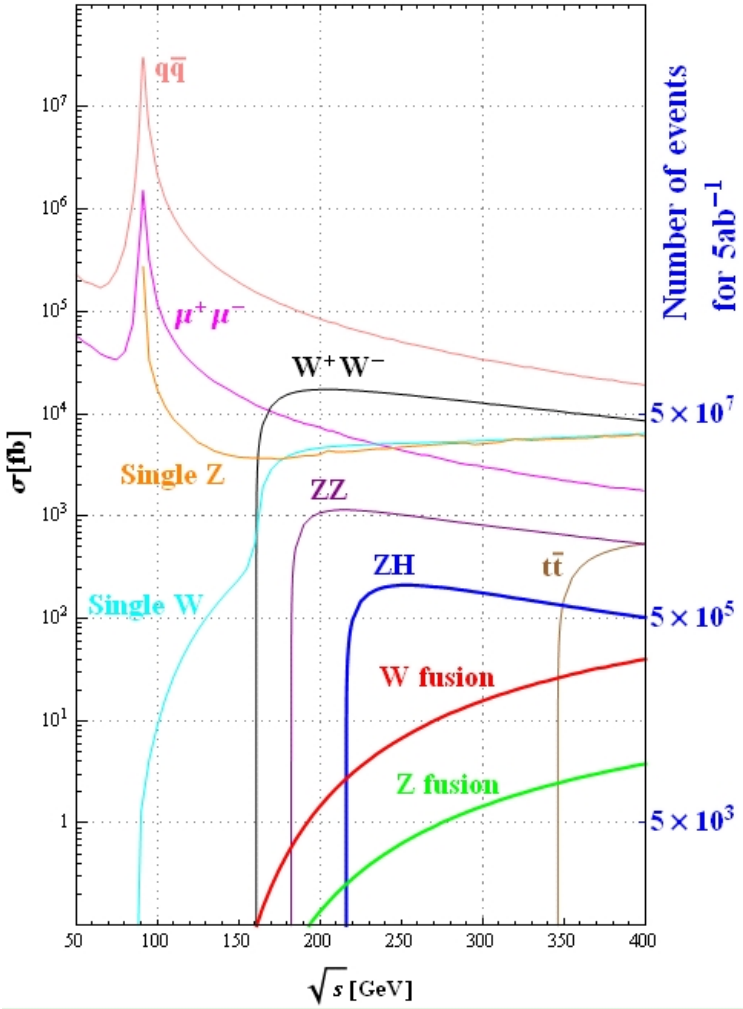
- At an LC, ALL events are interesting !
- Huge spread in cross-sections
- But for any given study, it might be a tiny cross-section one that dominates...
- ... or maybe a tiny fraction of a huge cross-section one.
- We want to make nice stacked histos of different backgrounds - different from analysis to analysis.
- $\Rightarrow$ Good idea to separate channels in an intelligent and well documented way





Generating the full SM

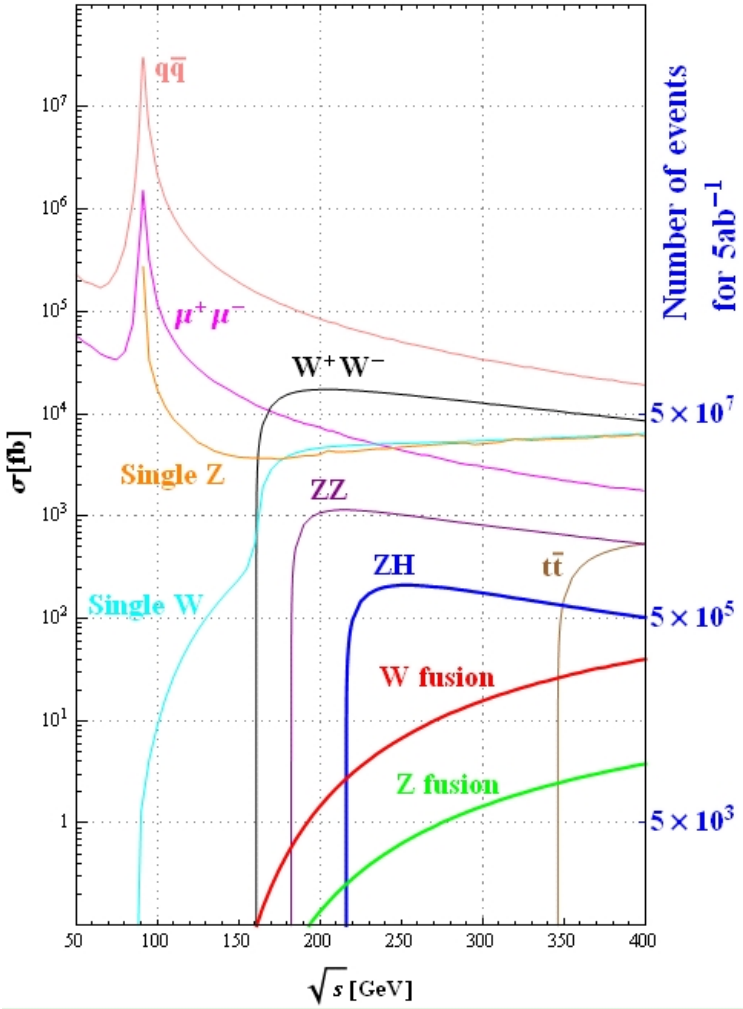
- At an LC, ALL events are interesting !
- Huge spread in cross-sections
- But for any given study, it might be a tiny cross-section one that dominates...
- ... or maybe a tiny fraction of a huge cross-section one.
- We want to make nice stacked histos of different backgrounds - different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent and well documented way



[from CepC, arXiv:1505.01008]

Generating the full SM

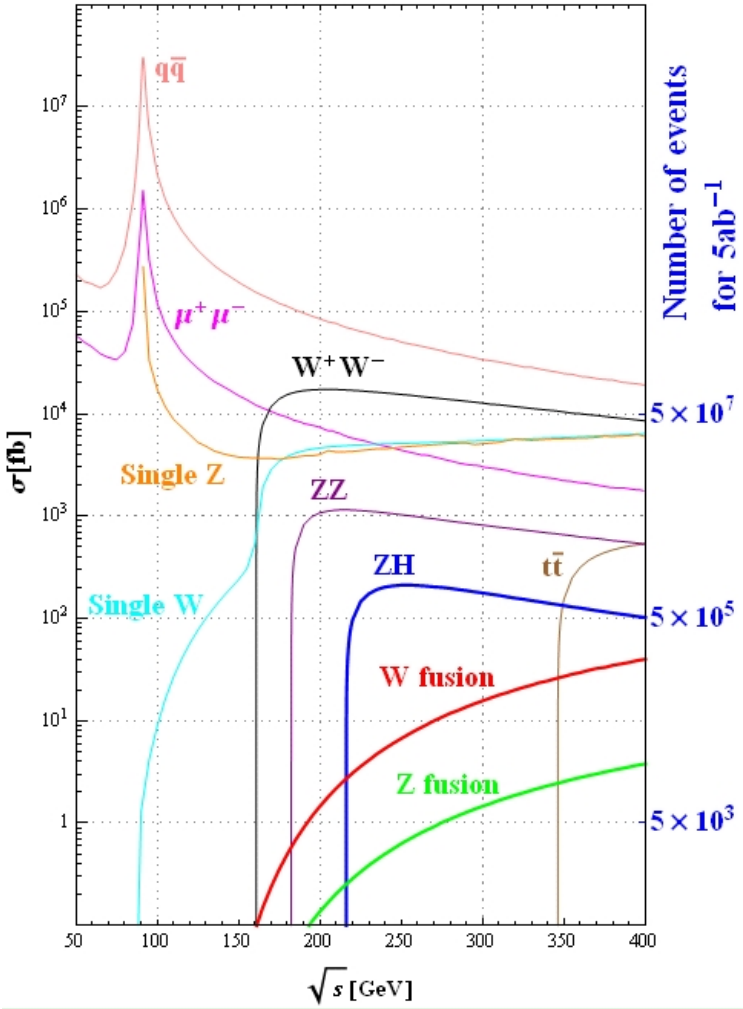
- At an LC, ALL events are interesting !
- Huge spread in cross-sections
- But for any given study, it might be a tiny cross-section one that dominates...
- ... or maybe a tiny fraction of a huge cross-section one.
- We want to make nice stacked histos of different backgrounds - different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent and well documented way



[from CepC, arXiv:1505.01008]

Generating the full SM

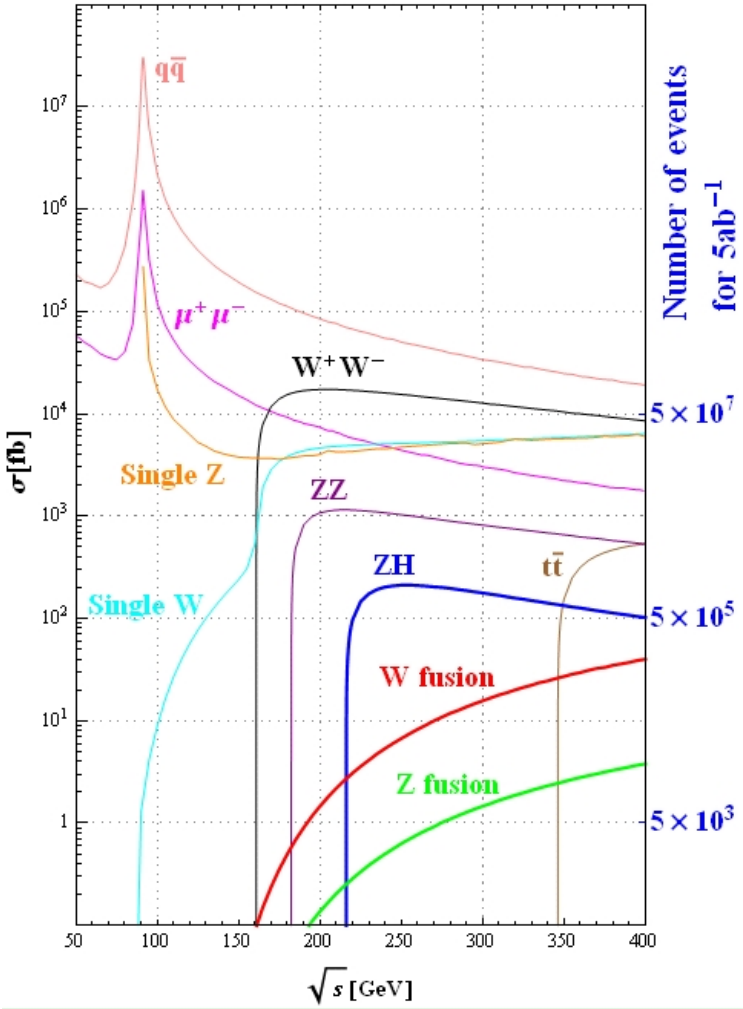
- At an LC, ALL events are interesting !
- Huge spread in cross-sections
- But for any given study, it might be a tiny cross-section one that dominates...
- ... or maybe a tiny fraction of a huge cross-section one.
- We want to make nice stacked histos of different backgrounds - different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent and well documented way



[from CepC, arXiv:1505.01008]

Generating the full SM

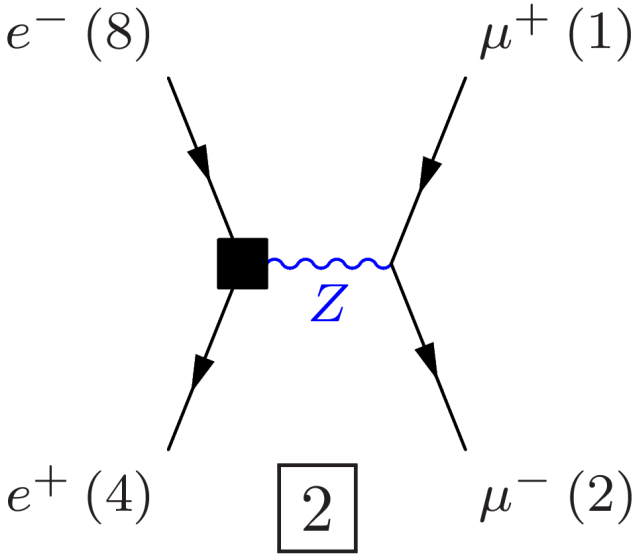
- At an LC, ALL events are interesting !
- Huge spread in cross-sections
- But for any given study, it might be a tiny cross-section one that dominates...
- ... or maybe a tiny fraction of a huge cross-section one.
- We want to make nice stacked histos of different backgrounds - different from analysis to analysis.
- $\Rightarrow$ Good idea to separate channels in an intelligent and well documented way



[from CepC, arXiv:1505.01008]

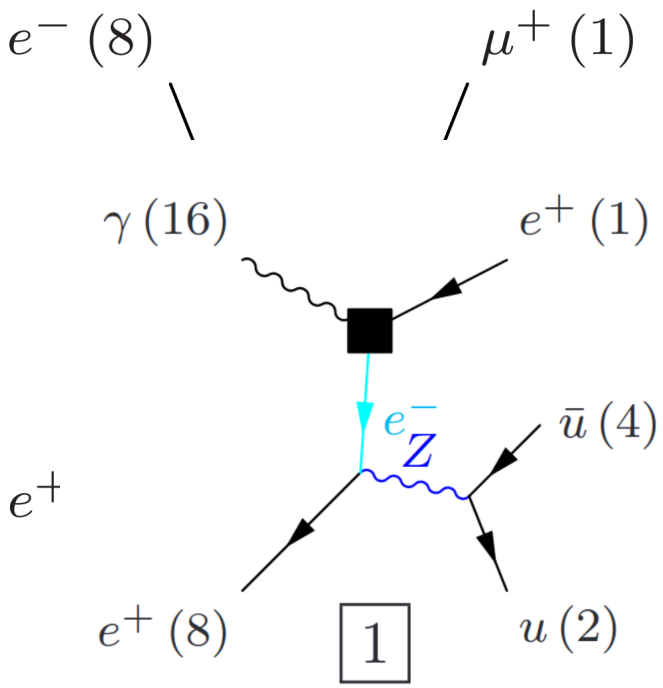
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



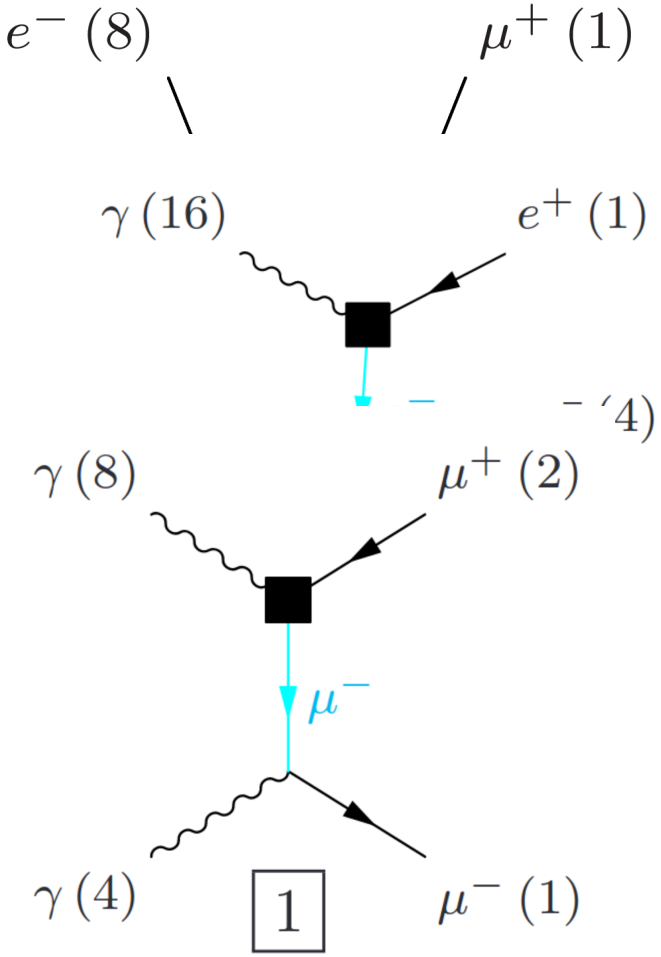
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



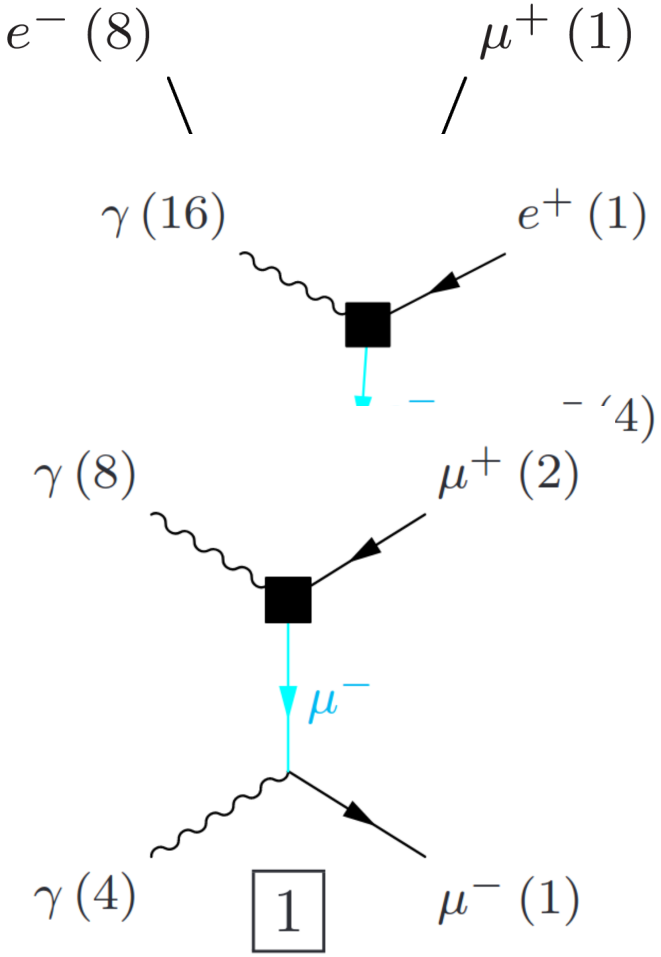
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



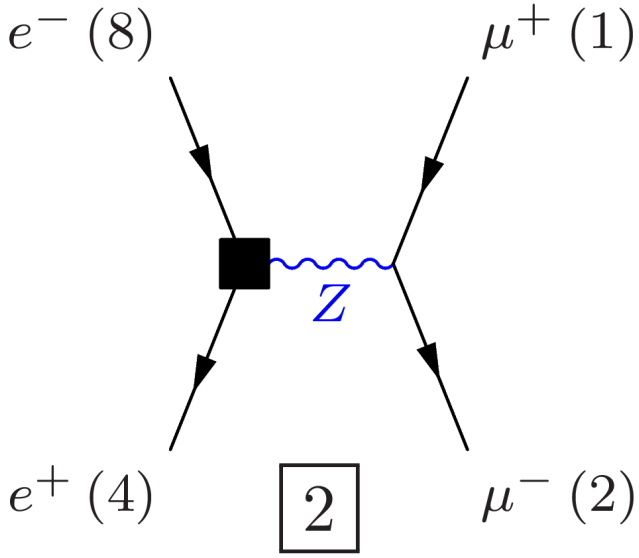
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



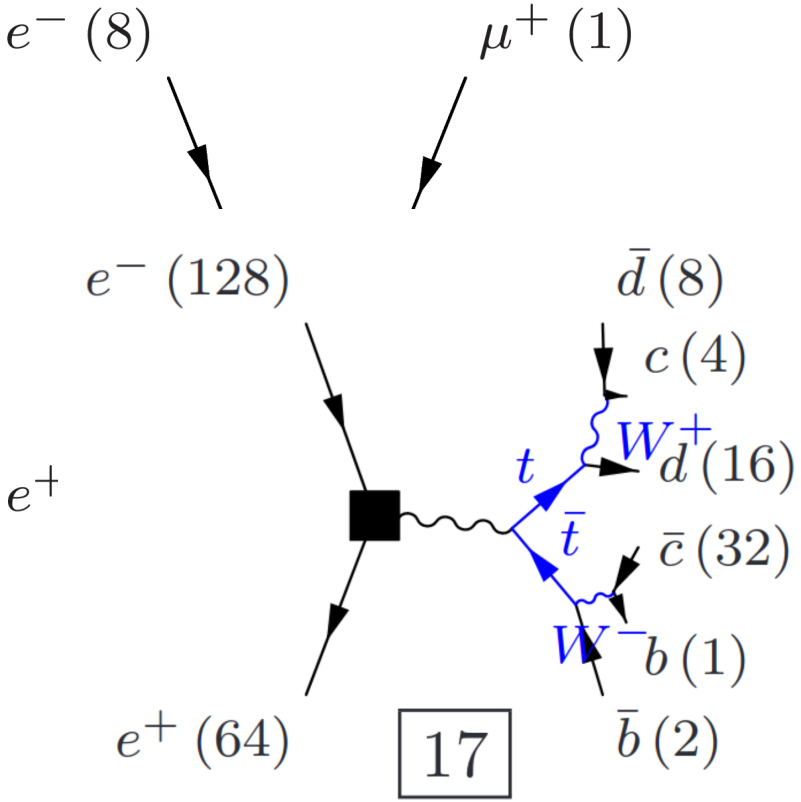
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



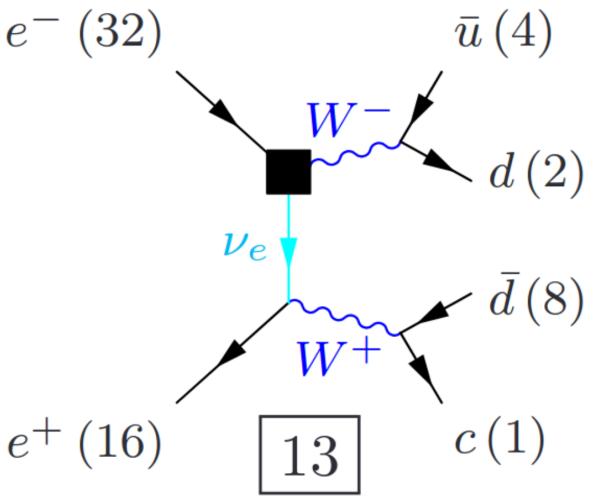
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptoni (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



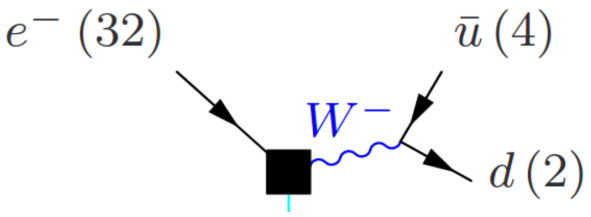
Process classification

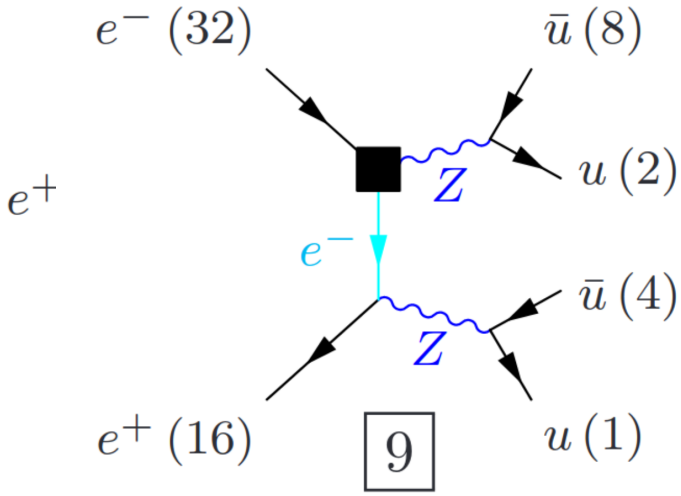
- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptonic (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



Process classification

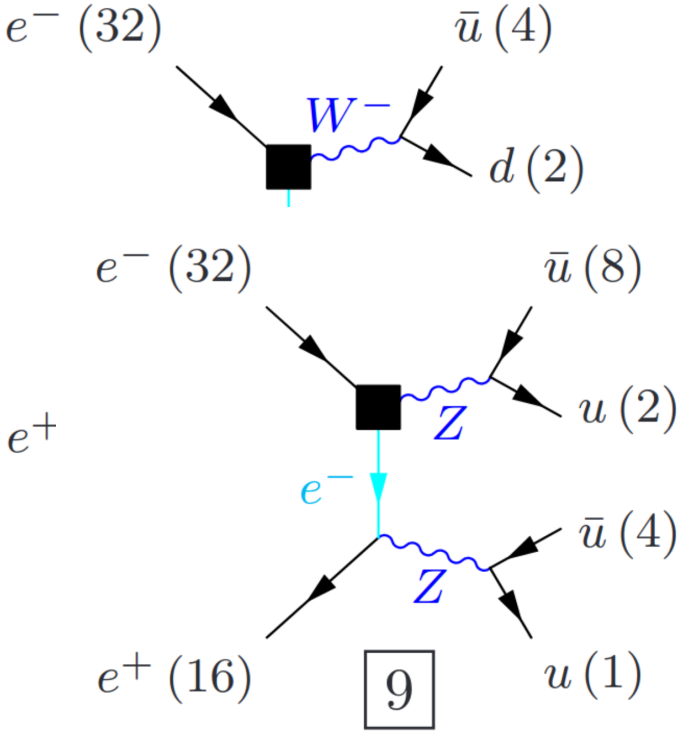
- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptonic (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.





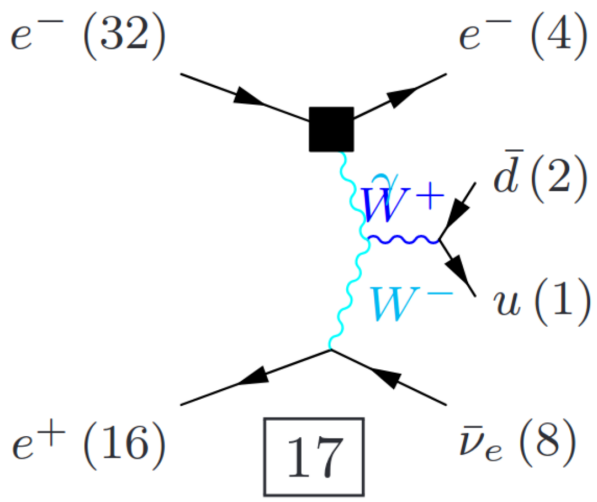
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptonic (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^- f\bar{f}$.



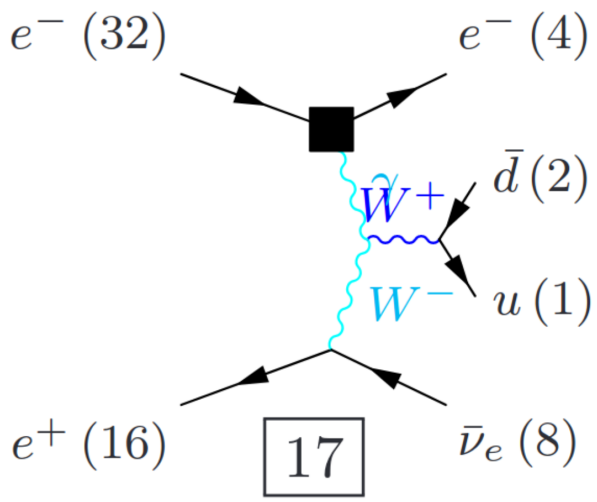
Process classification

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptonic (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^-f\bar{f}$.



Process classification

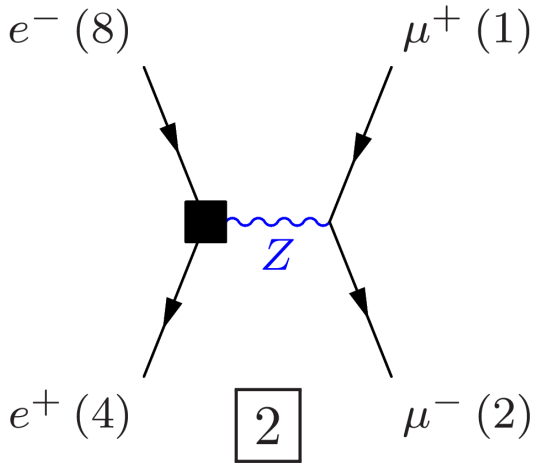
- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (1 to 8)
  - Flavour-grouping: W or Z, or ambiguous
  - leptonic, hadronic, semi-leptonic (+ neutrino only, for Z-leptonic)
- Special considerations
  - Eg. 4f with $|L_e|$=2 $\Rightarrow$ dominated by single W or single Z (t-channel !)
  - Avoid double-counting: Eg. $\gamma^*\gamma^* \to f\bar{f}$ vs. $e^+e^- \to e^+e^-f\bar{f}$.



Main generator: Whizard

Whizard is the generator of choice for e^+e^-

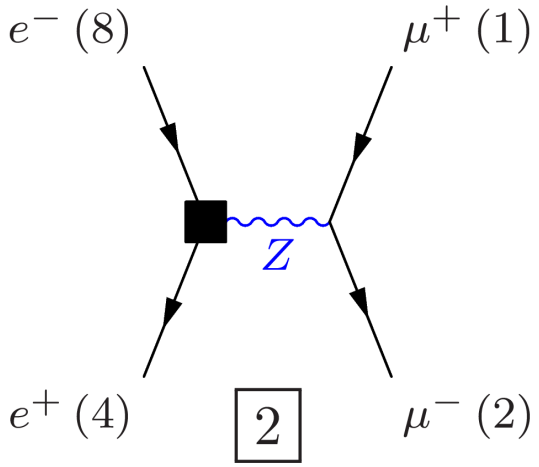
- Full matrix-element evaluation.
  Only at tree-level but:
  - Polarised beams.
  - Full helicity treatment.
  - Full colour flow, passed from the hard interaction to the P.S. code.
  - Can handle beam-spectrum, using Circe2.
  - Can handle polarised τ-decays, using TAUOLA [Comp Phys Comm,64,p275].
  - And: Can do $2 \to 8$ processes.
- ... which is more important than NLO for e^+e^- ! That was just one 5f tree-level diagram of ~ 500...



Main generator: Whizard

Whizard is the generator of choice for e^+e^-

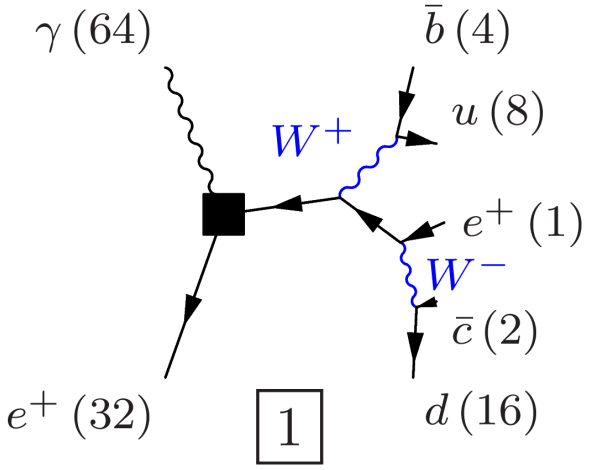
- Full matrix-element evaluation. Only at tree-level but:
  - Polarised beams.
  - Full helicity treatment.
  - Full colour flow, passed from the hard interaction to the P.S. code.
  - Can handle beam-spectrum, using Circe2.
  - Can handle polarised τ-decays, using TAUOLA [Comp Phys Comm,64,p275].
  - And: Can do $2 \to 8$ processes.
- ... which is more important than NLO for e^+e^- ! That was just one 5f tree-level diagram of ~ 500...



Main generator: Whizard

Whizard is the generator of choice for e^+e^-

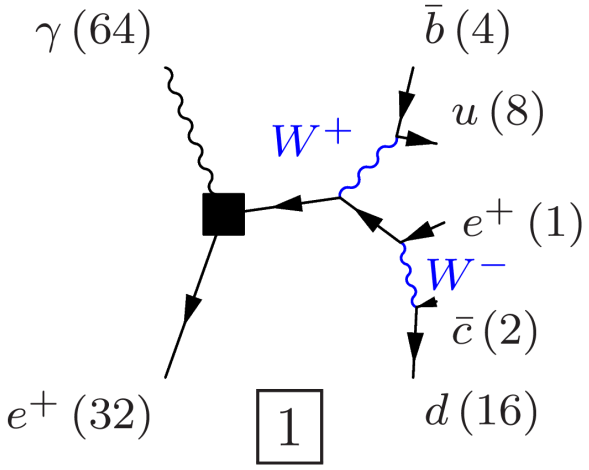
- Full matrix-element evaluation. Only at tree-level but:
  - Polarised beams.
  - Full helicity treatment.
  - Full colour flow, passed from the hard interaction to the P.S. code.
  - Can handle beam-spectrum, using Circe2.
  - Can handle polarised τ-decays, using TAUOLA [Comp Phys Comm,64,p275].
  - And: Can do $2 \to 8$ processes.
- ... which is more important than NLO for e^+e^- ! That was just one 5f tree-level diagram of ~ 500...



Main generator: Whizard

Whizard is the generator of choice for e^+e^-

- Full matrix-element evaluation. Only at tree-level but:
  - Polarised beams.
  - Full helicity treatment.
  - Full colour flow, passed from the hard interaction to the P.S. code.
  - Can handle beam-spectrum, using Circe2.
  - Can handle polarised τ-decays, using TAUOLA [Comp Phys Comm,64,p275].
  - And: Can do $2 \to 8$ processes.
- ... which is more important than NLO for e^+e^- ! That was just one 5f tree-level diagram of ~ 500...



Main generator: Whizard

Whizard is the generator of choice for e^+e^-.

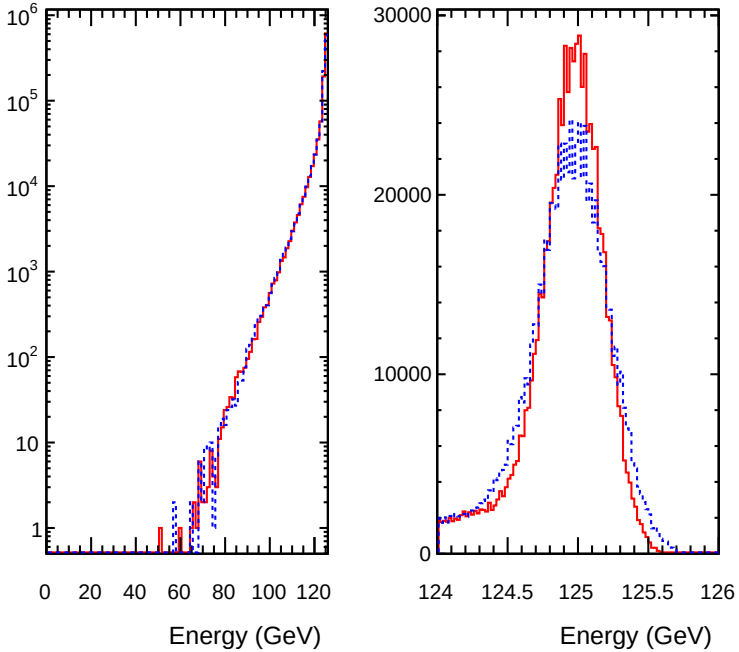
- Full matrix-element evaluation. Only at tree-level but:
  - Polarised beams.
  - Full helicity treatment.
  - Full colour flow, passed from the hard interaction to the P.S. code.
  - Can handle beam-spectrum, using Circe2.
  - Can handle polarised τ-decays, using TAUOLA [Comp Phys Comm,64,p275].
  - And: Can do $2 \to 8$ processes.
- ... which is more important than NLO for e^+e^- ! That was just one 5f tree-level diagram of ~ 500...

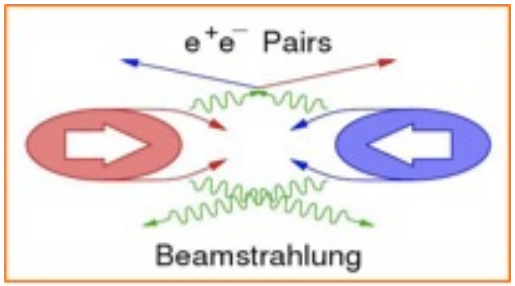
Main generator: Whizard

- The subsequent parton-shower and hadronisation is done by PYTHIA6.4.
  - LCGG has tuned hadronisation using input from OPAL at LEPII [Phys.Rept. 291 (1997) 107-217, D. Ward, private communication.].
- The process-definition given in the Whizard steering file (aka the *sindarin*) is also the driver for the scripts that organises the production: One ring to rule them all.
- Use powerful grouping and aliasing capabilities of sindarin to assure that no processes are over-looked.

Beam effects

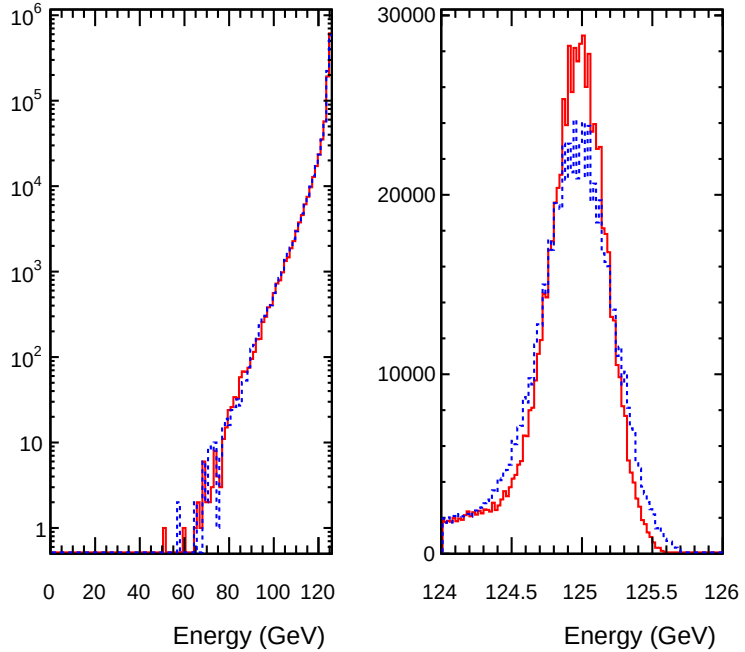
- Beam-spectrum.
  1. Incoming beam-spread
  2. But also: very strongly focused beams ⇒ Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig [CERN-PS-99-014-LP]. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Amount and spectrum of real photons
  - Distribution of interaction point

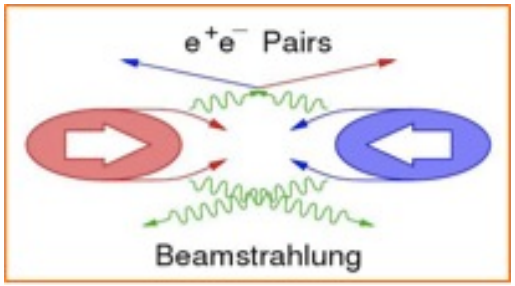




Beam effects

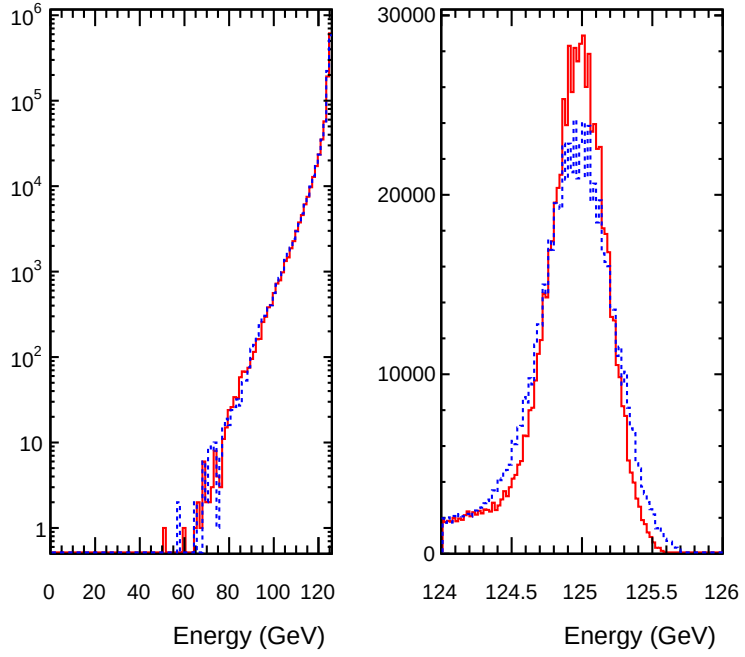
- Beam-spectrum.
  1. Incoming beam-spread
  2. But also: very strongly focused beams ⇒ Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig [CERN-PS-99-014-LP]. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Amount and spectrum of real photons
  - Distribution of interaction point

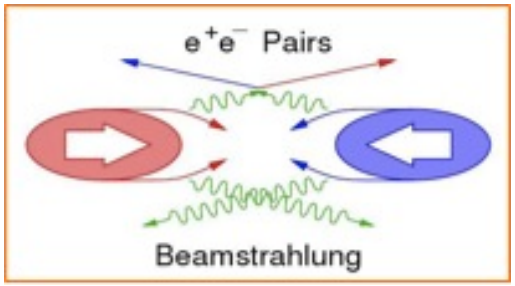




Beam effects

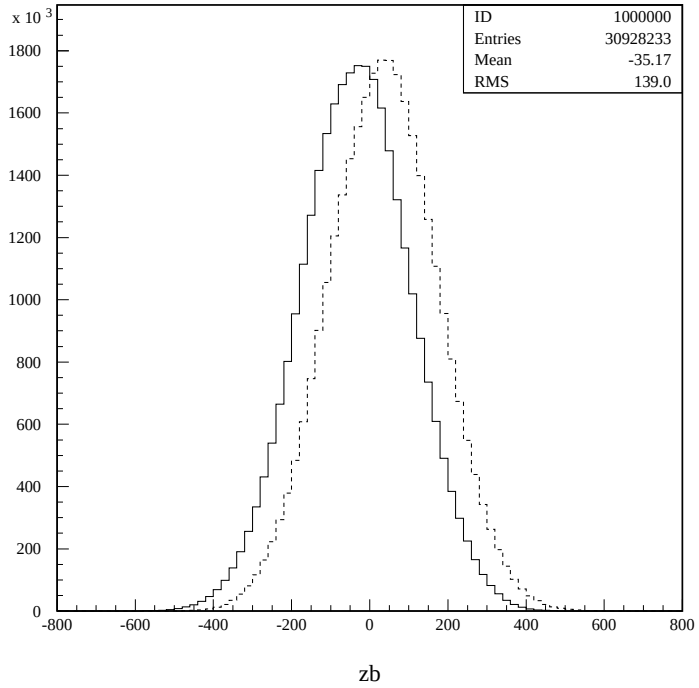
- Beam-spectrum.
  1. Incoming beam-spread
  2. But also: very strongly focused beams ⇒ Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig [CERN-PS-99-014-LP]. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Amount and spectrum of real photons
  - Distribution of interaction point

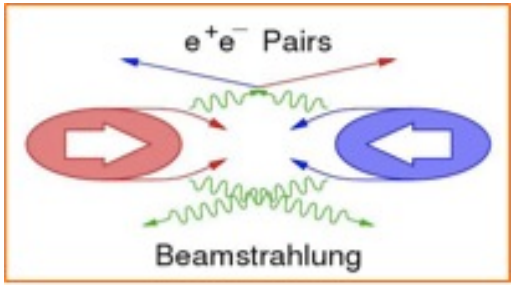




Beam effects

- Beam-spectrum.
  1. Incoming beam-spread
  2. But also: very strongly focused beams ⇒ Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig [CERN-PS-99-014-LP]. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Amount and spectrum of real photons
  - Distribution of interaction point

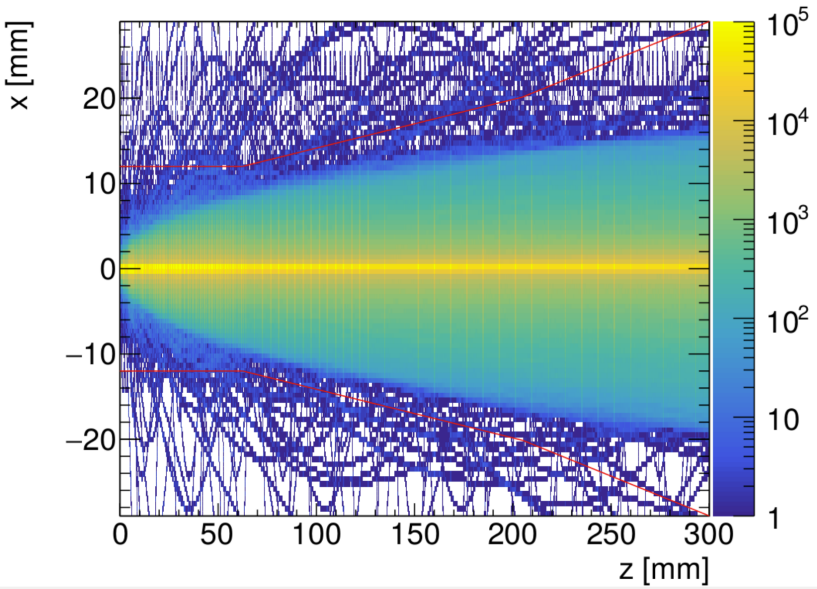




Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig can generate the full activity during a beam-crossing (a "BX").
- low-$p_\perp$ hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small $M_{\gamma\gamma}$ and multiplicity. NB: only $\mathcal{O}(1)$/BX !
  - ME can't do this, and PYTHIA is good down to $M_{\gamma\gamma} \sim 2$ GeV.
  - Below: fit to data - Custom generator developed by LCGG.
- Both types: Pre-generate pool of events, pick at random and overlay on main event.

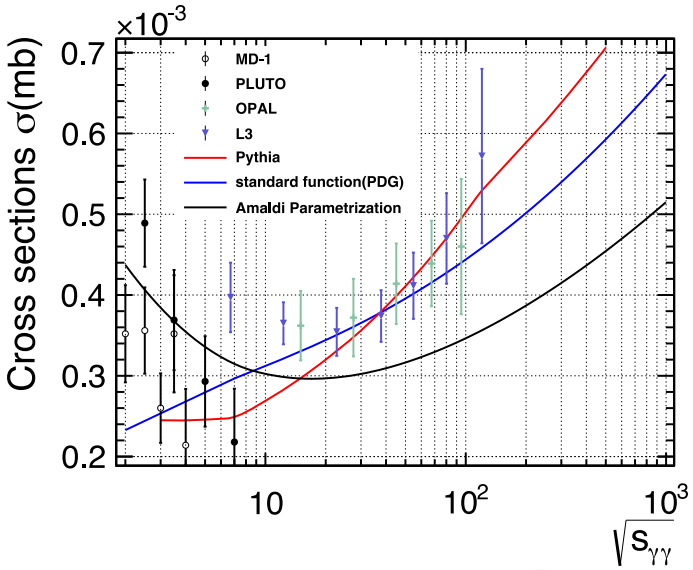


- For pairs: Do $\sim 10^5$ BXes, use fast detector simulation SGV to filter out track hitting tracking ($\sim$10/BX), rest used to build a map of background on BeamCal.

Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig can generate the full activity during a beam-crossing (a "BX").
- low-$p_\perp$ hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small $M_{\gamma\gamma}$ and multiplicity. NB: only $\mathcal{O}(1)$/BX !
  - ME can't do this, and PYTHIA is good down to $M_{\gamma\gamma} \sim 2$ GeV.
  - Below: fit to data - Custom generator developed by LCGG.
- Both types: Pre-generate pool of events, pick at random and overlay on main event.

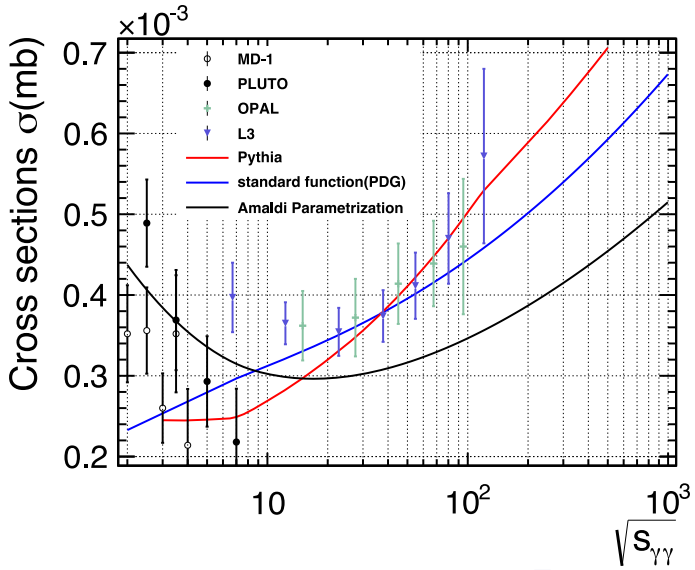


- For pairs: Do $\sim 10^5$ BXes, use fast detector simulation SGV to filter out track hitting tracking ($\sim$10/BX), rest used to build a map of background on BeamCal.

Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig can generate the full activity during a beam-crossing (a "BX").
- low-$p_\perp$ hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small $M_{\gamma\gamma}$ and multiplicity. NB: only $\mathcal{O}(1)$/BX !
  - ME can't do this, and PYTHIA is good down to $M_{\gamma\gamma} \sim 2$ GeV.
  - Below: fit to data - Custom generator developed by LCGG.
- Both types: Pre-generate pool of events, pick at random and overlay on main event.

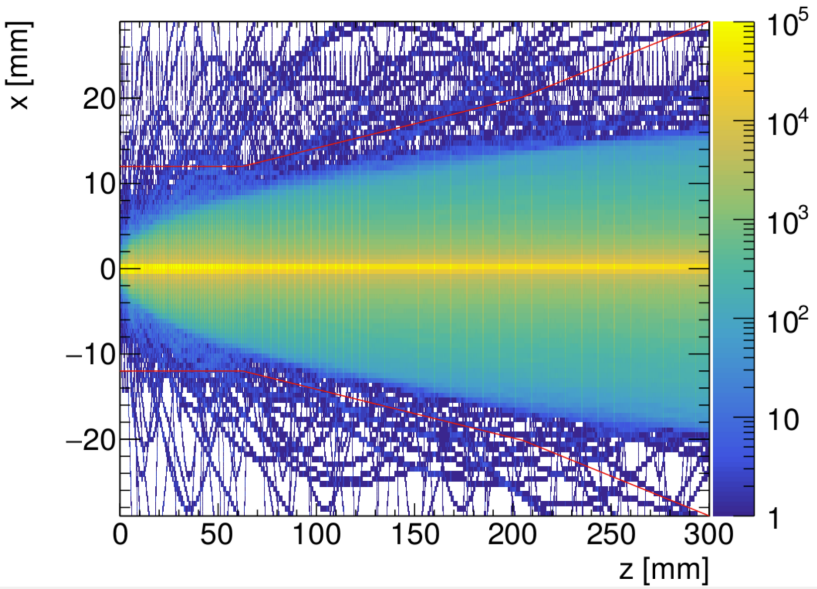


- For pairs: Do $\sim 10^5$ BXes, use fast detector simulation SGV to filter out track hitting tracking ($\sim$10/BX), rest used to build a map of background on BeamCal.

Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig can generate the full activity during a beam-crossing (a "BX").
- low-$p_\perp$ hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small $M_{\gamma\gamma}$ and multiplicity. NB: only $\mathcal{O}(1)$/BX !
  - ME can't do this, and PYTHIA is good down to $M_{\gamma\gamma} \sim 2$ GeV.
  - Below: fit to data - Custom generator developed by LCGG.
- Both types: Pre-generate pool of events, pick at random and overlay on main event.



- For pairs: Do $\sim 10^5$ BXes, use fast detector simulation SGV to filter out track hitting tracking ($\sim$10/BX), rest used to build a map of background on BeamCal.

ILC 250 GeV Generation production

Currently, ILC is doing a new full production at 250 GeV, 10 times larger than the one done for the DBD/IDR, and about twice the expected real data-set. NB: Also includes the Higgs signal.

proposal for statistics of 250 GeV generators

process\pol.	eL.pR	eR.pL	eL.pL	eR.pR
2f_l, 2f_h	5 ab^{-1}	5 ab^{-1}	1 ab^{-1}	1 ab^{-1}
all 4f				
all 6f	10K	10K	10K	10K
2f_bhabhag	1 ab^{-1}	1 ab^{-1}	1 ab^{-1}	1 ab^{-1}
h->inclusive	1 ab^{-1}	1 ab^{-1}	1 ab^{-1}	1 ab^{-1}
h->each mode (5x9 channels)	100K	100K	10K	10K

most of the irreducible background will then have x10 more than expectation at ILC250

aa_2f, aa_4f: 1 ab^{-1} each initial state

ILC 250 GeV production: Whizard 1.x → 2.x

- DBD was done with v. 1.95
- v2.x is a major re-write; we finally use v2.8.5. Many new features.
  - New, better steering
  - Things done by us now part of the main code:
    - Interface to FastJet (kt, Durham, Cambridge, Cone algorithms)
    - Interface with Tauola++ (tau decays)
    - Beam spectrum, ISR, Beam-strahlung, Photon PDF
    - Interface to Pythia6 (parton showering, hadronization)
  - Generated events directly in LCIO format.
  - Samples from new BSM models much easier to create, using tools like UFO.
  - And also:
    - Interface to Pythia8 (parton showering, hadronization)
    - Interface to Madgraph5 (matrix element generation)
    - Interface to Madgraph5aMC@NLO (matrix element generation, NLO)
    - Interface to Madgraph5aMC@NLO (matrix element generation, NLO)

ILC 250 GeV production: Whizard 1.x → 2.x

- DBD was done with v. 1.95
- v2.x is a major re-write; we finally use v2.8.5. Many new features.
  - New, better steering
  - Things done by us now part of the main code:
    - Interface to PYTHIA (parton-shower and hadronisation).
    - Interface TAUOLA (polarised τ-decays)
    - Beam-spectrum CIRCE2: Easier to study of impact of beam parameters
  - Generated events directly in LCIO format.
  - Samples from new BSM models much easier to create, using tools like UFO.
  - And also:
    - Internal parton-shower with matching of gluons between hard process- p.s.- hadronisation.
    - 8 fermion final states possible (ttH !). Was not (practically) possible with Whizard 1.95.
    - Improved treatment of $t\bar{t}$ threshold.

ILC 250 GeV production: Whizard 1.x → 2.x

- DBD was done with v. 1.95
- v2.x is a major re-write; we finally use v2.8.5. Many new features.
  - New, better steering
  - Things done by us now part of the main code:
    - Interface to PYTHIA (parton-shower and hadronisation).
    - Interface TAUOLA (polarised τ-decays)
    - Beam-spectrum CIRCE2: Easier to study of impact of beam parameters
  - Generated events directly in LCIO format.
  - Samples from new BSM models much easier to create, using tools like UFO.
  - And also:
    - Internal parton-shower with matching of gluons between hard process- p.s.- hadronisation.
    - 8 fermion final states possible (ttH !). Was not (practically) possible with Whizard 1.95.
    - Improved treatment of $t\bar{t}$ threshold.

Whizard 1.x → 2.x issues.

Polarised τ decays

- Whizard makes polarised τ:s ⇒ must decay polarised τ:s ⇒ TAUOLA.
- But: Pythia doesn't handle polarised decays ⇒ tweak needed.
- This was done by us with Whizard1, Whizard2 implementation of this was not correct.
- We helped the authors to get this right.

Colour-connection

- Whizard colour connects quarks ⇒ transfer to Pythia P.S. code.
- This was done by us with Whizard1, Whizard2 did not do it.
- We worked with the authors to fix this: Mark connection by artificial resonances in the event before handing over to Pythia.
- Caveat: Has ramifications on code analysing the structure, e.g. TrueJet or event re-scanning/re-weighting ⇒ W.I.P.

Whizard 1.x → 2.x issues.

Polarised τ decays

- Whizard makes polarised τ:s ⇒ must decay polarised τ:s ⇒ TAUOLA.
- But: Pythia doesn't handle polarised decays ⇒ tweak needed.
- This was done by us with Whizard1, Whizard2 implementation of this was not correct.
- We helped the authors to get this right.

Colour-connection

- Whizard colour connects quarks ⇒ transfer to Pythia P.S. code.
- This was done by us with Whizard1, Whizard2 did not do it.
- We worked with the authors to fix this: Mark connection by artificial resonances in the event before handing over to Pythia.
- Caveat: Has ramifications on code analysing the structure, e.g. TrueJet or event re-scanning/re-weighting ⇒ W.I.P.

Whizard 1.x → 2.x issues.

Polarised τ decays

- Whizard makes polarised τ:s ⇒ must decay polarised τ:s ⇒ TAUOLA.
- But: Pythia doesn't handle polarised decays ⇒ tweak needed.
- This was done by us with Whizard1, Whizard2 implementation of this was not correct.
- We helped the authors to get this right.

Colour-connection

- Whizard colour connects quarks ⇒ transfer to Pythia P.S. code.
- This was done by us with Whizard1, Whizard2 did not do it.
- We worked with the authors to fix this: Mark connection by artificial resonances in the event before handing over to Pythia.
- Caveat: Has ramifications on code analysing the structure, e.g. TrueJet or event re-scanning/re-weighting ⇒ W.I.P.

Whizard 1.x → 2.x issues.

Polarised τ decays

- Whizard makes polarised τ:s ⇒ must decay polarised τ:s ⇒ TAUOLA.
- But: Pythia doesn't handle polarised decays ⇒ tweak needed.
- This was done by us with Whizard1, Whizard2 implementation of this was not correct.
- We helped the authors to get this right.

Colour-connection

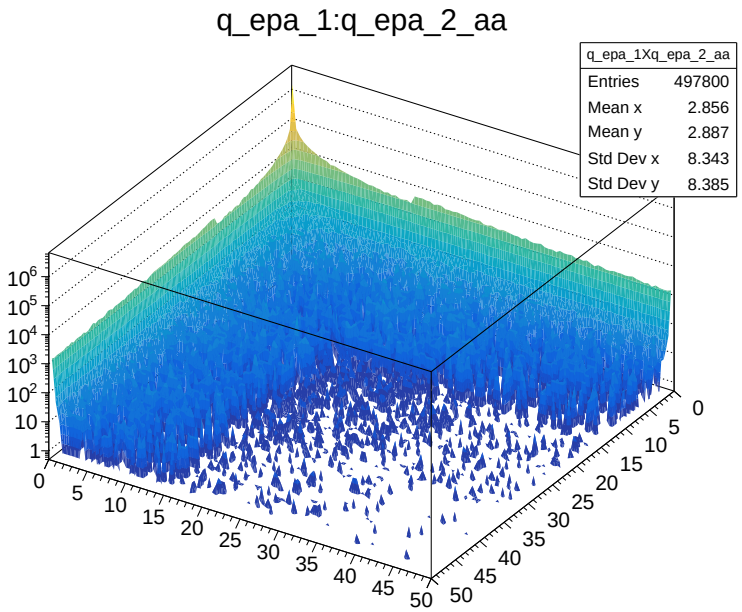
- Whizard colour connects quarks ⇒ transfer to Pythia P.S. code.
- This was done by us with Whizard1, Whizard2 did not do it.
- We worked with the authors to fix this: Mark connection by artificial resonances in the event before handing over to Pythia.
- Caveat: Has ramifications on code analysing the structure, e.g. TrueJet or event re-scanning/re-weighting ⇒ W.I.P.

Whizard 1.x → 2.x issues

Virtual incoming γ:s

- $e^+e^- \to e^+e^- f\bar{f}$ is generated in different ways, depending on the Q^2 between in- and out-going $e^{+(-)}$:
  1. Both high: as M.E. of $e^+e^- \to e^+e^- f\bar{f}$
  2. One high, the other low: as M.E. of $e^{+(-)}\gamma^* \to e^{+(-)} f\bar{f}$
  3. Both low: as M.E. of $\gamma^*\gamma^* \to f\bar{f}$

  where the γ^* is generated with the EPA off an incoming $e^{+(-)}$. The scattered $e^{+(-)}$ is present in the final state (the "beam-remnant").



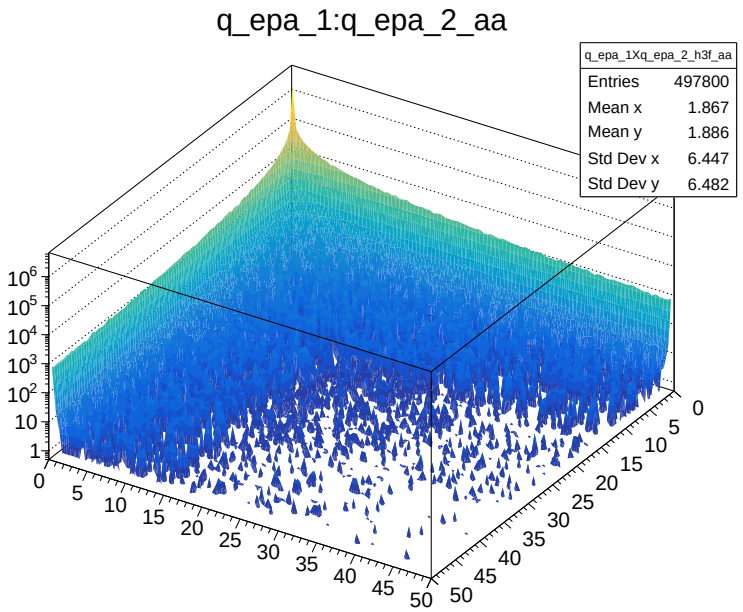
- In Whizard2, the cross-section of 2 is two times too high.
- Authors alerted, but for now these channels are being deferred in the production.

Whizard 1.x → 2.x issues

Virtual incoming γ:s

- $e^+e^- \to e^+e^- f\bar{f}$ is generated in different ways, depending on the Q^2 between in- and out-going $e^{+(-)}$:
  1. Both high: as M.E. of $e^+e^- \to e^+e^- f\bar{f}$
  2. One high, the other low: as M.E. of $e^{+(-)}\gamma^* \to e^{+(-)} f\bar{f}$
  3. Both low: as M.E. of $\gamma^*\gamma^* \to f\bar{f}$

  where the γ^* is generated with the EPA off an incoming $e^{+(-)}$. The scattered $e^{+(-)}$ is present in the final state (the "beam-remnant").



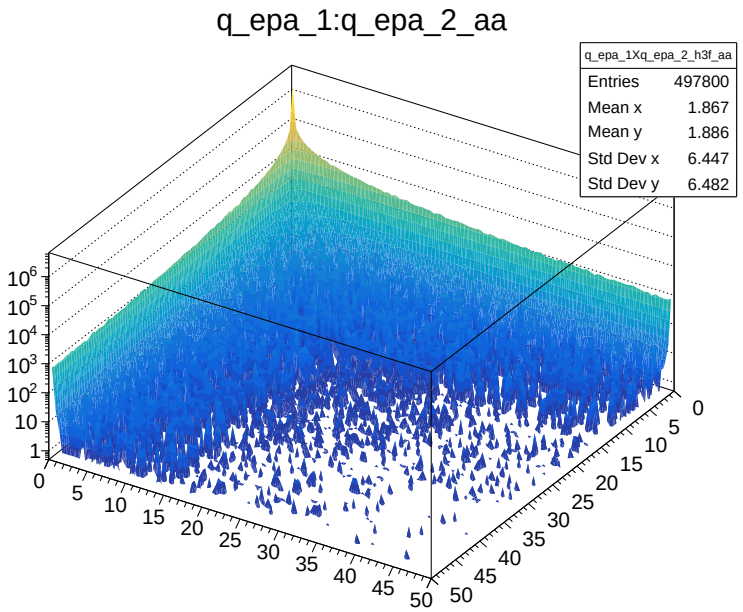
- In Whizard2, the cross-section of 2 is two times too high.
- Authors alerted, but for now these channels are being deferred in the production.

Whizard 1.x → 2.x issues

Virtual incoming γ:s

- $e^+e^- \to e^+e^- f\bar{f}$ is generated in different ways, depending on the Q^2 between in- and out-going $e^{+(-)}$:
  1. Both high: as M.E. of $e^+e^- \to e^+e^- f\bar{f}$
  2. One high, the other low: as M.E. of $e^{+(-)}\gamma^* \to e^{+(-)} f\bar{f}$
  3. Both low: as M.E. of $\gamma^*\gamma^* \to f\bar{f}$

  where the γ^* is generated with the EPA off an incoming $e^{+(-)}$. The scattered $e^{+(-)}$ is present in the final state (the "beam-remnant").



- In Whizard2, the cross-section of 2 is two times too high.
- Authors alerted, but for now these channels are being deferred in the production.

Beams: TDR ⇒ SetA

Beam parameters

- The beam-parameters of this production are different from the TDR ones !
- At TDR times, the focus was on 500 GeV, and the 250 GeV did not receive much attention.
- Since, the "SetA" beam-settings have been found to be optimal for the Higgs-factory stage.
- It buys more luminosity, but also more disruption ⇒ more background, larger tails.
- This alone is a reason that You should not use the 250-TDR samples anymore ! They simply do not correspond to the machine design on the table...

Beams: TDR ⇒ SetA

Beam parameters

- The beam-parameters of this production are different from the TDR ones !
- At TDR times, the focus was on 500 GeV, and the 250 GeV did not receive much attention.
- Since, the "SetA" beam-settings have been found to be optimal for the Higgs-factory stage.
- It buys more luminosity, but also more disruption ⇒ more background, larger tails.
- This alone is a reason that You should not use the 250-TDR samples anymore ! They simply do not correspond to the machine design on the table...

Beams: TDR ⇒ SetA

Beam parameters

- The beam-parameters of this production are different from the TDR ones !
- At TDR times, the focus was on 500 GeV, and the 250 GeV did not receive much attention.
- Since, the "SetA" beam-settings have been found to be optimal for the Higgs-factory stage.
- It buys more luminosity, but also more disruption ⇒ more background, larger tails.
- This alone is a reason that You should not use the 250-TDR samples anymore ! They simply do not correspond to the machine design on the table...

ILC 250 GeV production: Generation status

- For $\leq$ 5f (w/o virtual γ induced ones): 116 channels.
- As of today, 104 channels are done, producing 2.7 billion events in 15788 LCIO files details, occupying 5.4 TB. This used 7233 CPU hours, obtained in $\sim$ 10 days.
  - The remaining channels are only $\mathcal{O}(10000)$ events.
  - In addition, there are 96 channels with virtual γ:s to come, $\sim$0.5 billion events.
- In most cases: one channel = one generation job, but in some processes alone represent $\sim$ billion events: split in several jobs. In total, 478 jobs have been completed.
- At the end of each job, the events (in LCIO format), metadata, and input+log-file tarballs are uploaded to the grid.
- At the end of each channel, summary metadata of all jobs of the channel are uploaded to the grid. This triggers the simulation and reconstruction system under DIRAC to do it's thing.
- This documentation details is also available on the Web.

ILC 250 GeV production: Generation status

- For $\leq$ 5f (w/o virtual γ induced ones): 116 channels.
- As of today, 104 channels are done, producing 2.7 billion events in 15788 LCIO files details, occupying 5.4 TB. This used 7233 CPU hours, obtained in $\sim$ 10 days.
  - The remaining channels are only $\mathcal{O}(10000)$ events.
  - In addition, there are 96 channels with virtual γ:s to come, $\sim$0.5 billion events.
- In most cases: one channel = one generation job, but in some processes alone represent $\sim$ billion events: split in several jobs. In total, 478 jobs have been completed.
- At the end of each job, the events (in LCIO format), metadata, and input+log-file tarballs are uploaded to the grid.
- At the end of each channel, summary metadata of all jobs of the channel are uploaded to the grid. This triggers the simulation and reconstruction system under DIRAC to do it's thing.
- This documentation details is also available on the Web.

Conclusions

- The precision-goal of future e^+e^- colliders are such that it is not permissible that MC statistics or coverage could constitute an major systematic uncertainty.
- We showed how the generation of the full SM, with statistics well above the expected H20@250 data set is achieved.
  - It consists of bringing a large number of different codes together:
    - [illegible]
    - [illegible]
    - [illegible]
    - [illegible]
- This full data is organised and documented in a physics-oriented fashion, for the benefit of the end-user.
- Full simulation and reconstruction is on-going.

Conclusions

- The precision-goal of future e^+e^- colliders are such that it is not permissible that MC statistics or coverage could constitute an major systematic uncertainty.
- We showed how the generation of the full SM, with statistics well above the expected H20@250 data set is achieved.
  - It consists of bringing a large number of different codes together:
    - Whizard, Pythia, and Tauola for physics generation
    - GuineaPig and Circe2 for beam-properties
    - SGV+GuineaPig and the Peskin/Barklow generator for pile-up
    - In addition, input from machine-physics and data from LEPII was used.
- This full data is organised and documented in a physics-oriented fashion, for the benefit of the end-user.
- Full simulation and reconstruction is on-going.

Conclusions

- The precision-goal of future e^+e^- colliders are such that it is not permissible that MC statistics or coverage could constitute an major systematic uncertainty.
- We showed how the generation of the full SM, with statistics well above the expected H20@250 data set is achieved.
  - It consists of bringing a large number of different codes together:
    - Whizard, Pythia, and Tauola for physics generation
    - GuineaPig and Circe2 for beam-properties
    - SGV+GuineaPig and the Peskin/Barklow generator for pile-up
    - In addition, input from machine-physics and data from LEPII was used.
- This full data is organised and documented in a physics-oriented fashion, for the benefit of the end-user.
- Full simulation and reconstruction is on-going.

Conclusions

- The precision-goal of future e^+e^- colliders are such that it is not permissible that MC statistics or coverage could constitute an major systematic uncertainty.
- We showed how the generation of the full SM, with statistics well above the expected H20@250 data set is achieved.
  - It consists of bringing a large number of different codes together:
    - Whizard, Pythia, and Tauola for physics generation
    - GuineaPig and Circe2 for beam-properties
    - SGV+GuineaPig and the Peskin/Barklow generator for pile-up
    - In addition, input from machine-physics and data from LEPII was used.
- This full data is organised and documented in a physics-oriented fashion, for the benefit of the end-user.
- Full simulation and reconstruction is on-going.

Conclusions

- The precision-goal of future e^+e^- colliders are such that it is not permissible that MC statistics or coverage could constitute an major sys
- We show atistics well above th
  - It co s together:
    - pile-up
    - LEPII was
- This full cs-oriented fashion, f
- Full simu

The new samples not only represents a huge increase in statistics, and ameliorations in physics description, but are the only ones that correctly represents the current machine conditions @ 250 GeV.

They should replace the 250-TDR samples for any physics analysis ASAP.

BACKUP

BACKUP SLIDES

Documentation

Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions. It goes into:

- The event header:
  - Process-id, beam nature ($e^{+(-)}$ or γ), beam polarisation, beam spectrum, cross-section, run- and event-number of each event.
- The generator meta-data files:
  - Condenses job-specific information from Whizard logs.
  - Contains: process, cross section, polarisation, file-names, total number of events generated, total integrated luminosity, technicalities, ...
  - Browasble on the Web and uploaded to the Grid.
  - Once on the grid, the metadata is read by DIRAC to orchestrate the detector simulation.
- Steering-files, logs, pdf:s with diagrams, integration grids, output other than the events,... (Sufficient information to re-run):
  - On the Web, in full.
  - In tar files - parallel to generated files, also stored on the Grid.

Documentation

Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions. It goes into:

- The event header:
  - Process-id, beam nature ($e^{+(-)}$ or γ), beam polarisation, beam spectrum, cross-section, run- and event-number of each event.
- The generator meta-data files:
  - Condenses job-specific information from Whizard logs.
  - Contains: process, cross section, polarisation, file-names, total number of events generated, total integrated luminosity, technicalities, ...
  - Browasble on the Web and uploaded to the Grid.
  - Once on the grid, the metadata is read by DIRAC to orchestrate the detector simulation.
- Steering-files, logs, pdf:s with diagrams, integration grids, output other than the events,... (Sufficient information to re-run):
  - On the Web, in full.
  - In tar files - parallel to generated files, also stored on the Grid.

Documentation

Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions. It goes into:

- The event header:
  - Process-id, beam nature ($e^{+(-)}$ or γ), beam polarisation, beam spectrum, cross-section, run- and event-number of each event.
- The generator meta-data files:
  - Condenses job-specific information from Whizard logs.
  - Contains: process, cross section, polarisation, file-names, total number of events generated, total integrated luminosity, technicalities, ...
  - Browasble on the Web and uploaded to the Grid.
  - Once on the grid, the metadata is read by DIRAC to orchestrate the detector simulation.
- Steering-files, logs, pdf:s with diagrams, integration grids, output other than the events,... (Sufficient information to re-run):
  - On the Web, in full.
  - In tar files - parallel to generated files, also stored on the Grid.

Documentation

Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions. It goes into:

- The event header:
  - Process-id, beam nature ($e^{+(-)}$ or γ), beam polarisation, beam spectrum, cross-section, run- and event-number of each event.
- The generator meta-data files:
  - Condenses job-specific information from Whizard logs.
  - Contains: process, cross section, polarisation, file-names, total number of events generated, total integrated luminosity, technicalities, ...
  - Browasble on the Web and uploaded to the Grid.
  - Once on the grid, the metadata is read by DIRAC to orchestrate the detector simulation.
- Steering-files, logs, pdf:s with diagrams, integration grids, output other than the events,... (Sufficient information to re-run):
  - On the Web, in full.
  - In tar files - parallel to generated files, also stored on the Grid.

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA: [illegible]
    - P4f_ww_h: [illegible]
    - Gwhizard-2_8_5: [illegible]
    - eL.pR: [illegible]
    - I500066: [illegible]
    - 317: [illegible]
    - slcio: [illegible]
  - Unfortunately, this is a [illegible] file-name [illegible]
  - [illegible]
  - [illegible]

back

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: where and what

- The files can be found on
  - .../ilc/prod/ilc/mc-2020/generated/250-SetA/*
- Anatomy of the file-name:
  - E250-SetA.P4f_ww_h.Gwhizard-2_8_5.eL.pR.I500066.317.slcio
    - E250-SetA = E(nergy)[E_{CMS}]_[beam conditions]
    - P4f_ww_h =P(rocess)[(aa_[1])number of final state particles] _[NC/CC/mixed flavour combination]_[hadronic/leptonic/semi-leptonicfinal state]
    - Gwhizard-2_8_5 =G(enerator)[name and version of generator] (NB: Unfortunately, this tag is suppressed in the name of the DST files...)
    - eL = e(lectron beam)[polarisation (L/R) or γ type (W/B)]
    - pR = p(ositron beam)[polarisation (l/R) or γ type (W/B)]
    - I = I(detification number)[id (also run number)]

back

[1] added to $\gamma\gamma$ events to separate from e^+e^- events with the same number of fermions

The files: News

- The format is LCIO, the contents is one MCParticle collection/event
  - This is not the MCParticle you will see on the DST: Obviously, on the DST, particles created in the detector simulation and from overlay are added. But not only:
    - The crossing-angle boost is added.
    - The production point has been swimmed around.
    - For flying particles, the effect of the detector B-field and energy loss is added, so that the decay properties are modified.
- The event numbers are unique, without gaps.
  - The event header contains run and events number which is now unique, and carried on all the way to the DST.
- The header also contains the cross-section of the process. A GOTCHA: the true cross-section. It does not take into account that the luminosity for e^+e^-, $e^{+(-)}\gamma$ and $\gamma\gamma$ processes are different!
- The early parts of the event record (initial state, resonance insertions, status codes) is different.

The files: News

- The format is LCIO, the contents is one MCParticle collection/event
  - This is not the MCParticle you will see on the DST: Obviously, on the DST, particles created in the detector simulation and from overlay are added. But not only:
    - The crossing-angle boost is added.
    - The production point has been swimmed around.
    - For flying particles, the effect of the detector B-field and energy loss is added, so that the decay properties are modified.
- The event numbers are unique, without gaps.
  - The event header contains run and events number which is now unique, and carried on all the way to the DST.
- The header also contains the cross-section of the process. A GOTCHA: the true cross-section. It does not take into account that the luminosity for e^+e^-, $e^{+(-)}\gamma$ and $\gamma\gamma$ processes are different!
- The early parts of the event record (initial state, resonance insertions, status codes) is different.

The files: News

- The format is LCIO, the contents is one MCParticle collection/event
  - This is not the MCParticle you will see on the DST: Obviously, on the DST, particles created in the detector simulation and from overlay are added. But not only:
    - The crossing-angle boost is added.
    - The production point has been swimmed around.
    - For flying particles, the effect of the detector B-field and energy loss is added, so that the decay properties are modified.
- The event numbers are unique, without gaps.
  - The event header contains run and events number which is now unique, and carried on all the way to the DST.
- The header also contains the cross-section of the process. A GOTCHA: the true cross-section. It does not take into account that the luminosity for e^+e^-, $e^{+(-)}\gamma$ and $\gamma\gamma$ processes are different!
- The early parts of the event record (initial state, resonance insertions, status codes) is different.

The files: News

- The format is LCIO, the contents is one MCParticle collection/event
  - This is not the MCParticle you will see on the DST: Obviously, on the DST, particles created in the detector simulation and from overlay are added. But not only:
    - The crossing-angle boost is added.
    - The production point has been swimmed around.
    - For flying particles, the effect of the detector B-field and energy loss is added, so that the decay properties are modified.
- The event numbers are unique, without gaps.
  - The event header contains run and events number which is now unique, and carried on all the way to the DST.
- The header also contains the cross-section of the process. A GOTCHA: the true cross-section. It does not take into account that the luminosity for e^+e^-, $e^{+(-)}\gamma$ and $\gamma\gamma$ processes are different!
- The early parts of the event record (initial state, resonance insertions, status codes) is different.