Generating the full SM at ILC

Mikael Berggren[1]

[1]DESY, Hamburg

LCWS, Sendai, Japan , Oct 2019

Outline

Simulating the SM

Simulate the full SM? What's the problem?

- Just select a generator, and press <RET>, right?
- Noooo..., not really. Lots of details:
  - What collides (electrons or photons?)
    - How are they polarised?
    - What energy do they have ?
  - Where do they collide ?
    - Beam-spot properties
  - What else happens?
    - Beam-strahlung gives pairs
    - Do they hit anything ?
    - Maybe forward calorimetry ?
    - Or even the tracking system ?
    - Multiple interactions (pile-up) ?

Simulating the SM

Simulate the full SM? What's the problem?

- Just select a generator, and press <RET>, right?
- Noooo..., not really. Lots of details:
  - What collides (electrons or photons?)
    - How are they polarised?
    - What energy do they have?
  - Where do they collide ?
    - Beam-spot properties
  - What else happens?
    - Beam-strahlung pairs
    - Do they hit anything ?

Simulating the SM

Simulate the full SM? What's the problem?

- Just select a generator, and press <RET>, right?
- Noooo..., not really. Lots of details:
  - What collides (electrons or photons?)
    - How are they polarised?
    - What energy do they have ?
  - Where do they collide ?
    - Beam-spot properties
  - What else happens?
    - Beam-strahlung gives pairs
    - Do they hit anything ?
    - Maybe forward calorimetry ?
    - Or even the tracking system ?
    - Multiple interactions (pile-up) ?

Simulating the SM

Simulate the full SM? What's the problem?

- Just select a generator, and press <RET>, right?
- Noooo..., not really. Lots of details:
  - What collides (electrons or photons?)
    - How are they polarised?
    - What energy do they have ?
  - Where do they collide ?
    - Beam-spot properties
  - What else happens?
    - Beam-strahlung gives pairs
    - Do they hit anything ?
    - Maybe forward calorimetry ?
    - Or even the tracking system ?
    - Multiple interactions (pile-up) ?

Simulating the SM

Simulate the full SM? What's the problem?

- Just select a generator, and press <RET>, right?
- Noooo..., not really. Lots of details:
    - What collides (electrons or photons?)
        - How are they polarised?
        - What energy do they have ?
    - Where do they collide ?
        - Beam-spot properties
    - What else happens?
        - Beam-strahlung gives pairs
        - Do they hit anything ?
        - Maybe forward calorimetry ?
        - Or even the tracking system ?
        - Multiple interactions (pile-up) ?

Simulating the SM

In addition, the physics channels themselves:

- Huge spread in cross-sections
- But for any given study, it might be a low-cross-section one that dominates.
- ... or maybe a tiny fraction of a huge cross-section one.
- Analysers wants to make nice stacked histos of different background-sources - probably different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent way

Simulating the SM

In addition, the physics channels themselves:

- Huge spread in cross-sections
- But for any given study, it might be a low-cross-section one that dominates.
- ... or maybe a tiny fraction of a huge cross-section one.
- Analysers wants to make nice stacked histos of different background-sources - probably different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent way

Simulating the SM

In addition, the physics channels themselves:

- Huge spread in cross-sections
- But for any given study, it might be a low-cross-section one that dominates.
- ... or maybe a tiny fraction of a huge cross-section one.
- Analysers wants to make nice stacked histos of different background-sources - probably different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent way

Simulating the SM

In addition, the physics channels themselves:

- Huge spread in cross-sections
- But for any given study, it might be a low-cross-section one that dominates.
- ... or maybe a tiny fraction of a huge cross-section one.
- Analysers wants to make nice stacked histos of different background-sources - probably different from analysis to analysis.
- ⇒ Good idea to separate channels in an intelligent way

Simulating the SM

In addition, the physics channels themselves:

- Huge spread in cross-sections
- But for any given study, it might be a low-cross-section one that dominates.
- ... or maybe a tiny fraction of a huge cross-section one.
- Analysers wants to make nice stacked histos of different background-sources - probably different from analysis to analysis.
- $\Rightarrow$ Good idea to separate channels in an intelligent way

Beam effects

- Beam-spectrum.
  1. Incoming beam-spread
  2. Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Incoming beam-spread from damping-rings and ondulator: External input from machine-scientists.
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Distribution of interaction point
  - Amount and spectrum of real photons

Beam effects

- Beam-spectrum.
  1. Incoming beam-spread
  2. Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Incoming beam-spread from damping-rings and ondulator: External input from machine-scientists.
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Distribution of interaction point
  - Amount and spectrum of real photons

Beam effects

- Beam-spectrum.
  1. Incoming beam-spread
  2. Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Incoming beam-spread from damping-rings and ondulator: External input from machine-scientists.
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Distribution of interaction point
  - Amount and spectrum of real photons

Beam effects

- Beam-spectrum.
  1. Incoming beam-spread
  2. Beam-beam interactions
- Photons
  1. How many photons?
  2. Are they virtual or real?
- Incoming beam-spread from damping-rings and ondulator: External input from machine-scientists.
- Need beam-beam interaction simulation input.
- Simulate interaction region: GuineaPig. Gives:
  - Beam-spectrum for electrons and positrons independently
  - Distribution of interaction point
  - Amount and spectrum of real photons

Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig also gives us this.
- low-p$_\perp$t hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small invariant masses
  - ME can't do this, so need different generator
  - PYTHIA is good down to $\sim$ 2 GeV
  - A lot happens below that, but is basically not known theoretically $\Rightarrow$ need to fit to data
  - Data is scarce, and ambiguous ...

Spurious interactions ("pile-up")

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig also gives us this.
- low-$p_\perp$t hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small invariant masses
  - ME can't do this, so need different generator
  - PYTHIA is good down to $\sim$ 2 GeV
  - A lot happens below that, but is basically not known theoretically $\Rightarrow$ need to fit to data
  - Data is scarce, and ambiguous ...

Spurious interactions (“pile-up”)

Two types:

- Pair-background: Pair-creation of photons in the beam by the strong fields. GuineaPig also gives us this.
- low-p$_\perp$t hadrons, ie. $\gamma^{(*)}\gamma^{(*)}$ interaction with small invariant masses
  - ME can’t do this, so need different generator
  - PYTHIA is good down to $\sim$ 2 GeV
  - A lot happens below that, but is basically not known theoretically $\Rightarrow$ need to fit to data
  - Data is scarce, and ambiguous ...

Spurious interactions ("pile-up")

- These backgrounds need to be passed on to simulation, but in a different mode.
- Eg. can't simulate $\sim 10^5$ pairs on each physics event.
- Actually, can't generate that either: time for 1 BX 5-10 minutes
- Find the few tracks that do hit the tracking ($<$ 100/BX). Do $\sim$ 100000 BXes, and pick a random one from the pool to overlay to each physics event.
- Also, use some ($\mathcal{O}(100)$) BXes to simulate pairs hitting the BeamCal, to build a map of the background, to be used in the BeamCal simulation.
- Similar for low-$p_\perp$ hadrons, but here also the number per BX is random, and their production point.

Spurious interactions ("pile-up")

- These backgrounds need to be passed on to simulation, but in a different mode.
- Eg. can't simulate $\sim 10^5$ pairs on each physics event.
- Actually, can't generate that either: time for 1 BX 5-10 minutes
- Find the few tracks that do hit the tracking (< 100/BX). Do $\sim$ 100000 BXes, and pick a random one from the pool to overlay to each physics event.
- Also, use some ($\mathcal{O}(100)$) BXes to simulate pairs hitting the BeamCal, to build a map of the background, to be used in the BeamCal simulation.
- Similar for low-$p_\perp$ hadrons, but here also the number per BX is random, and their production point.

Spurious interactions (“pile-up”)

- These backgrounds need to be passed on to simulation, but in a different mode.
- Eg. can’t simulate $\sim 10^5$ pairs on each physics event.
- Actually, can’t generate that either: time for 1 BX 5-10 minutes
- Find the few tracks that do hit the tracking ($<$ 100/BX). Do $\sim$ 100000 BXes, and pick a random one from the pool to overlay to each physics event.
- Also, use some ($\mathcal{O}(100)$) BXes to simulate pairs hitting the BeamCal, to build a map of the background, to be used in the BeamCal simulation.
- Similar for low-$p_\perp$ hadrons, but here also the number per BX is random, and their production point.

Process classification

The classes

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (0 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

Process classification

The classes

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (0 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

Process classification

The classes

- Initial state
  - ee, eγ or $\gamma\gamma$
  - e - polarisation and γ type (real or virtual)
- Final state
  - Number of fermions (0 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

The generator group

- A group under the GDE was formed to select generator to generate physics events for the DBD benchmarks, including all relevant backgrounds.
- Mandate was for the DBD.
- However, ILD decided to get back into business, because:
  - New, better Whizard.
  - Developments in reconstruction.
  - Lack of statistics in some cases.
  - Things that went wrong in DBD & friends: Double-counted channels, beam-spot, un-physical $\gamma\gamma$ events, ..
- New, informal group (M.B., J. Tian, P. Roloff, Whizard authors.)
- Hot off the press: Now getting mandated by LCC (under the SW group).
- *Any news on this ?*

The generator group

- A group under the GDE was formed to select generator to generate physics events for the DBD benchmarks, including all relevant backgrounds.
- Mandate was for the DBD.
- However, ILD decided to get back into business, because:
  - New, better Whizard.
  - Developments in reconstruction.
  - Lack of statistics in some cases.
  - Things that went wrong in DBD & friends: Double-counted channels, beam-spot, un-physical $\gamma\gamma$ events, ..
- New, informal group (M.B., J. Tian, P. Roloff, Whizard authors.)
- Hot off the press: Now getting mandated by LCC (under the SW group).
- *Any news on this ?*

The generator group

- A group under the GDE was formed to select generator to generate physics events for the DBD benchmarks, including all relevant backgrounds.
- Mandate was for the DBD.
- However, ILD decided to get back into business, because:
  - New, better Whizard.
  - Developments in reconstruction.
  - Lack of statistics in some cases.
  - Things that went wrong in DBD & friends: Double-counted channels, beam-spot, un-physical $\gamma\gamma$ events, ..
- New, informal group (M.B., J. Tian, P. Roloff, Whizard authors.)
- Hot off the press: Now getting mandated by LCC (under the SW group).
- *Any news on this ?*

Whizard 2.8 status

- Whizard remains the generator of choice for e^+e^-.
- DBD was done with v. 1.95
- v2.x is a major re-write. Many new features.
  - New, better steering
  - Things done by us now part of the main code:
    - Interface to PYTHIA6 (parton-shower and hadronisation).
    - Interface TAUOLA (polarised τ-decays)
    - Beam-spectrum CIRCE2
  - Internal parton-shower with matching of gluons between hard process- p.s.- hadronisation.
  - Samples from new BSM models much easier to create, using tools like SARAH.
  - 8 fermion final states possible (ttH !). Was not (practically) possible with Whizard 1.95.
- Current version: 2.8

Whizard 2.8 status

- Whizard remains the generator of choice for e^+e^-.
- DBD was done with v. 1.95
- v2.x is a major re-write. Many new features.
  - New, better steering
  - Things done by us now part of the main code:
    - Interface to PYTHIAS (parton-shower and hadronisation).
    - Interface TAUOLA (polarised τ-decays)
    - Beam-spectrum CIRCE2
  - Internal parton-shower with matching of gluons between hard process- p.s.- hadronisation.
  - Samples from new BSM models much easier to create, using tools like SARAH.
  - 8 fermion final states possible (*ttH* !). Was not (practically) possible with Whizard 1.95.
- Current version: 2.8

Status of Whizard-2 for ILC : Some issues

- Technical requirements for mass production in batch mode:
  - Steering by command line options (Work-around in place).
  - *Should take some time to go through my scripts these days*
- For fast simulation: need callable interface (ie. no intermediate generator files).
- Need to re-visit and validate generator-level cuts.
  - In Whizard 1, built-in default cuts used (in most cases).
  - In Whizard 2, cuts are entirely up to the user. Need to formulate, and compare results.
  - Particularly delicate: cuts that separate same final state produced in two ways, to avoid double-counting.
    - Eg. $e^+e^- \to e^+e^- f\bar{f}$, either as $e^+e^- \to \gamma^*\gamma^* e^+e^- \to e^+e^- f\bar{f}$ or $e^+e^- \to ZZ \to e^+e^- f\bar{f}$.
- *Need to scrutinize cut setup*

Status of Whizard-2 for ILC : Some issues

- Technical requirements for mass production in batch mode:
  - Steering by command line options (Work-around in place).
  - *Should take some time to go through my scripts these days*
- For fast simulation: need callable interface (ie. no intermediate generator files).
- Need to re-visit and validate generator-level cuts.
  - In Whizard 1, built-in default cuts used (in most cases).
  - In Whizard 2, cuts are entirely up to the user. Need to formulate, and compare results.
  - Particularly delicate: cuts that separate same final state produced in two ways, to avoid double-counting.
    - Eg. $e^+e^- \to e^+e^- f\bar{f}$, either as $e^+e^- \to \gamma^*\gamma^* e^+e^- \to e^+e^- f\bar{f}$ or $e^+e^- \to ZZ \to e^+e^- f\bar{f}$.
- *Need to scrutinize cut setup*

Status of Whizard-2 for ILC : Some issues

- Technical requirements for mass production in batch mode:
  - Steering by command line options (Work-around in place).
  - *Should take some time to go through my scripts these days*
- For fast simulation: need callable interface (ie. no intermediate generator files).
- Need to re-visit and validate generator-level cuts.
  - In Whizard 1, built-in default cuts used (in most cases).
  - In Whizard 2, cuts are entirely up to the user. Need to formulate, and compare results.
  - Particularly delicate: cuts that separate same final state produced in two ways, to avoid double-counting.
    - Eg. $\mathrm{e}^+\mathrm{e}^- \rightarrow e^+e^- f\bar{f}$, either as $\mathrm{e}^+\mathrm{e}^- \rightarrow \gamma^*\gamma^* e^+e^- \rightarrow e^+e^- f\bar{f}$ or $\mathrm{e}^+\mathrm{e}^- \rightarrow ZZ \rightarrow e^+e^- f\bar{f}$.
- *Need to scrutinize cut setup*

Status of Whizard-2 for ILC : Some more issues

- 4f-SingleW leptonic and semileptonic bombs in integration if the incomming positron is R. If it is L, the result from the DBD is recuperated.
- I do get the same result for e^+e^- at 500 as we got in the DBD, but for γe and $\gamma\gamma$, the results differ. Both for W and B γ.
- In particular for 5f, there are some channels that are wildely different.
- Question: Is the event-header information in LCIO output as it should be?

Status of Whizard-2 for ILC : Other issues

- gluon matching between ME and PS:
  - Now: no gluons included in ME (setting α_s = 0) to avoid double-counting with (unmatched) parton shower in PYTHIA.
  - Whizard 2 can do it's own parton-shower with "MLM matching".
  - For now: stick with old scheme for mass-production, but make dedicated comparisons for multi-jet final states (eg $t\bar{t}$ background for ttH or ZZH)
- 8 fermions:
  - Whizard 1: Choked on 8 fermion.
  - Whizard 2: Scalable option: generated "byte code" running in a "virtual machine" (think Java!). Should work for 8f. Need to validate.
- Medium-term wishes:
  - γ ISR/FSR matching
  - Work out priority processes for EW-NLO (!)
- Generator group meets in Tokyo next week.

Status of Whizard-2 for ILC : Other issues

- gluon matching between ME and PS:
  - Now: no gluons included in ME (setting $\alpha_s = 0$) to avoid double-counting with (unmatched) parton shower in PYTHIA.
  - Whizard 2 can do it's own parton-shower with "MLM matching".
  - For now: stick with old scheme for mass-production, but make dedicated comparisons for multi-jet final states (eg $t\bar{t}$ background for ttH or ZZH)
- 8 fermions:
  - Whizard 1: Choked on 8 fermion.
  - Whizard 2: Scalable option: generated "byte code" running in a "virtual machine" (think Java!). Should work for 8f. Need to validate.
- Medium-term wishes:
  - γ ISR/FSR matching
  - Work out priority processes for EW-NLO (!)
- Generator group meets in Tokyo next week.

Status of Whizard-2 for ILC : Other issues

- gluon matching between ME and PS:
  - Now: no gluons included in ME (setting α_s = 0) to avoid double-counting with (unmatched) parton shower in PYTHIA.
  - Whizard 2 can do it's own parton-shower with "MLM matching".
  - For now: stick with old scheme for mass-production, but make dedicated comparisons for multi-jet final states (eg $t\bar{t}$ background for ttH or ZZH)
- 8 fermions:
  - Whizard 1: Choked on 8 fermion.
  - Whizard 2: Scalable option: generated "byte code" running in a "virtual machine" (think Java!). Should work for 8f. Need to validate.
- Medium-term wishes:
  - γ ISR/FSR matching
  - Work out priority processes for EW-NLO (!)
- Generator group meets in Tokyo next week.

Status of Whizard-2 for ILC : Other issues

- gluon matching between ME and PS:
  - Now: no gluons included in ME (setting α_s = 0) to avoid double-counting with (unmatched) parton shower in PYTHIA.
  - Whizard 2 can do it's own parton-shower with "MLM matching".
  - For now: stick with old scheme for mass-production, but make dedicated comparisons for multi-jet final states (eg $t\bar{t}$ background for ttH or ZZH)
- 8 fermions:
  - Whizard 1: Choked on 8 fermion.
  - Whizard 2: Scalable option: generated "byte code" running in a "virtual machine" (think Java!). Should work for 8f. Need to validate.
- Medium-term wishes:
  - γ ISR/FSR matching
  - Work out priority processes for EW-NLO (!)
- Generator group meets in Tokyo next week.

Other generators

In general it would be nice to also have other generators,

- BHWide for better Bhabhas.
- BDK/BDKRC for $\gamma\gamma \to \ell\ell$
- Pythia8, MadGraph, Sherpa (See talk by Price tomorrow) for double-checks.
- Pythia8 instead of Pythia6 for hadronisation.

Alas, not much progress on this ...

ILC 250 GeV Generation production

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

1f, 3f, others: 100K each initial state

ILC 250 GeV Generation production

Beam-conditions etc:

- Beam background with GuineaPig, 100000 BXes (DONE)
- *Ship Circe2 files (have them for 250, 500 and 1 TeV) with Whizard ?*
  - Pair background
    - Need to create files with real tracks
    - One event with 1 BX
    - SGV is used to do this.
  - Beam-spectrum and Circe2 parametrisation.
  - Beam-spot size and position.
  - Input for BeamCal background maps.
- aa_lowpt for “pile-up” (DONE)(T. Barklow)
  - Events to overlay.
  - Average number per BX evaluated.

ILC 250 GeV Generation production: Space and Time

- Once Whizard is fully validated
  - Largest fraction of generation is the initial integration, has already been done for $\leq$ 5f: over-night for $\leq$ 5f.
  - One process per job - afterwards producing large sample is fast, but not yet fully evaluated.
  - Disk-space might be an issue, but here the switch from un-compressed stdhep output to compressed LCIO output is expected to help a lot.
  - A new issue wrt DBD, not yet addressed: Samples now so big that even a single channel/polarisation might need to be in separate jobs, so splitting procedure and synchronisation on run- and event-numbers between independent jobs must be worked out.
  - *Do we have what is needed for this on the Whizard side?*

ILC 250 GeV Generation production: Space and Time

- Once Whizard is fully validated
  - Largest fraction of generation is the initial integration, has already been done for $\leq$ 5f: over-night for $\leq$ 5f.
  - One process per job - afterwards producing large sample is fast, but not yet fully evaluated.
  - Disk-space might be an issue, but here the switch from un-compressed stdhep output to compressed LCIO output is expected to help a lot.
  - A new issue wrt DBD, not yet addressed: Samples now so big that even a single channel/polarisation might need to be in separate jobs, so splitting procedure and synchronisation on run- and event-numbers between independent jobs must be worked out.
  - *Do we have what is needed for this on the Whizard side?*

ILC Generation production

Generator meta-data files:

- Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions.
- Condenses job-specific information from Whizard logs.
- Contain: process, cross section, polarisation, files, ...
- To be stored on Web and in Grid-SE (and GitHub?)

Steering-files, logs, integration grids output other than the events,...:

- On the Web, in full (and GitHub?)
- In tar files - parallel to generated files, also to be stored in Grid-SE.

ILC Generation production

Generator meta-data files:

- Created by generation job, driven by the contents of the process-definition Sindarin script and common conditions.
- Condenses job-specific information from Whizard logs.
- Contain: process, cross section, polarisation, files, ...
- To be stored on Web and in Grid-SE (and GitHub?)

Steering-files, logs, integration grids output other than the events,...:

- On the Web, in full (and GitHub?)
- In tar files - parallel to generated files, also to be stored in Grid-SE.

Conclusions and Outlook

- Lots of goodies in Whizard 2
- A few final issues to rub out, then ready to go to production mode for 250 GeV.
- All auxiliary stuff (beam-spectra, pile-up, beam-spots, BeamCal backgrounds, ...) are done.
- Some logistics to work out, then massive statistics can be generated, much bigger than the DBD.
- For the timescale: Your input would be most appreciated: Lot's of purely technical things could be done down-stream in simulation and reconstruction (new compiler versions, versions of Root and Geant4, ...)
- ... but these could be reduced to mere bug fixes, if there is a strong push to get going ASAP!

Conclusions and Outlook

- Lots of goodies in Whizard 2
- A few final issues to rub out, then ready to go to production mode for 250 GeV.
- All auxiliary stuff (beam-spectra, pile-up, beam-spots, BeamCal backgrounds, ...) are done.
- Some logistics to work out, then massive statistics can be generated, much bigger than the DBD.
- For the timescale: Your input would be most appreciated: Lot's of purely technical things could be done down-stream in simulation and reconstruction (new compiler versions, versions of Root and Geant4, ...)
- ... but these could be reduced to mere bug fixes, if there is a strong push to get going ASAP!

Conclusions and Outlook

- Lots of goodies in Whizard 2
- A few final issues to rub out, then ready to go to production mode for 250 GeV.
- All auxiliary stuff (beam-spectra, pile-up, beam-spots, BeamCal backgrounds, ...) are done.
- Some logistics to work out, then massive statistics can be generated, much bigger than the DBD.
- For the timescale: Your input would be most appreciated: Lot's of purely technical things could be done down-stream in simulation and reconstruction (new compiler versions, versions of Root and Geant4, ...)
- ... but these could be reduced to mere bug fixes, if there is a strong push to get going ASAP!

Conclusions and Outlook

- Lots of goodies in Whizard 2
- A few final issues to rub out, then ready to go to production mode for 250 GeV.
- All auxiliary stuff (beam-spectra, pile-up, beam-spots, BeamCal backgrounds, ...) are done.
- Some logistics to work out, then massive statistics can be generated, much bigger than the DBD.
- For the timescale: Your input would be most appreciated: Lot's of purely technical things could be done down-stream in simulation and reconstruction (new compiler versions, versions of Root and Geant4, ...)
- ... but these could be reduced to mere bug fixes, if there is a strong push to get going ASAP!