UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

The ATLAS simulation infrastructure

Aad, G.; et al., [Unknown]; Bentvelsen, S.; Bobbink, G.J.; Bos, K.; Boterenbrood, H.; Colijn, A.P.; de Jong, P.; Doxiadis, A.; Ferrari, P.; Garitaonandia, H.; Gosselink, M.; Hartjes, F.; Hessey, N.P.; Igonkina, O.; Kayl, M.S.; Klous, S.; Kluit, P.; Koffeman, E.; Koutsman, A.; Lee, H.; Liebig, W.; Mechnich, J.; Mussche, I.; Ottersbach, J.P.; Rijpstra, M.; Ruckstuhl, N.; Salamanna, G.; Sandstroem, R.; Snuverink, J.; Tsiakiris, M.; Turlay, E.; van der Graaf, H.; van der Kraaij, E.; van der Poel, E.; van Kesteren, Z.; van Vulpen, I.; Verkerke, W.; Vermeulen, J.C.; Vreeswijk, M.

DOI
10.1140/epjc/s10052-010-1429-9

Publication date
2010

Document Version
Final published version

Published in
European Physical Journal C

Link to publication

Citation for published version (APA):
Aad, G., et al., U., Bentvelsen, S., Bobbink, G. J., Bos, K., Boterenbrood, H., Colijn, A. P., de Jong, P., Doxiadis, A., Ferrari, P., Garitaonandia, H., Gosselink, M., Hartjes, F., Hessey, N. P., Igonkina, O., Kayl, M. S., Klous, S., Kluit, P., Koffeman, E., ... Vreeswijk, M. (2010). The ATLAS simulation infrastructure. *European Physical Journal C*, *70*(3), 823-874. https://doi.org/10.1140/epjc/s10052-010-1429-9

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

Download date:05 Sept 2026

Eur. Phys. J. C (2010) 70: 823–874
DOI 10.1140/epjc/s10052-010-1429-9

THE EUROPEAN PHYSICAL JOURNAL C

Special Article - Tools for Experiment and Theory

The ATLAS Simulation Infrastructure

The ATLAS Collaboration[*,**]

G. Aad[48], **B. Abbott**[111], **J. Abdallah**[11], **A.A. Abdelalim**[49], **A. Abdesselam**[118], **O. Abdinov**[10], **B. Abi**[112], **M. Abolins**[88], **H. Abramowicz**[152], **H. Abreu**[115], **B.S. Acharya**[163a,163b], **D.L. Adams**[24], **T.N. Addy**[56], **J. Adelman**[174], **C. Adorisio**[36a,36b], **P. Adragna**[75], **T. Adye**[129], **S. Aefsky**[22], **J.A. Aguilar-Saavedra**[124b], **M. Aharrouche**[81], **S.P. Ahlen**[21], **F. Ahles**[48], **A. Ahmad**[147], **H. Ahmed**[2], **M. Ahsan**[40], **G. Aielli**[133a,133b], **T. Akdogan**[18a], **T.P.A. Åkesson**[79], **G. Akimoto**[154], **A.V. Akimov**[94], **A. Aktas**[48], **M.S. Alam**[1], **M.A. Alam**[76], **S. Albrand**[55], **M. Aleksa**[29], **I.N. Aleksandrov**[65], **C. Alexa**[25a], **G. Alexander**[152], **G. Alexandre**[49], **T. Alexopoulos**[9], **M. Alhroob**[20], **M. Aliev**[15], **G. Alimonti**[89a], **J. Alison**[120], **M. Aliyev**[10], **P.P. Allport**[73], **S.E. Allwood-Spiers**[53], **J. Almond**[82], **A. Aloisio**[102a,102b], **R. Alon**[170], **A. Alonso**[79], **M.G. Alviggi**[102a,102b], **K. Amako**[66], **C. Amelung**[22], **A. Amorim**[124a], **G. Amorós**[166], **N. Amram**[152], **C. Anastopoulos**[139], **T. Andeen**[29], **C.F. Anders**[48], **K.J. Anderson**[30], **A. Andreazza**[89a,89b], **V. Andrei**[58a], **X.S. Anduaga**[70], **A. Angerami**[34], **F. Anghinolfi**[29], **N. Anjos**[124a], **A. Annovi**[47], **A. Antonaki**[8], **M. Antonelli**[47], **S. Antonelli**[19a,19b], **J. Antos**[144b], **B. Antunovic**[41], **F. Anulli**[132a], **S. Aoun**[83], **G. Arabidze**[8], **I. Aracena**[143], **Y. Arai**[66], **A.T.H. Arce**[14], **J.P. Archambault**[28], **S. Arfaoui**[29,a], **J.-F. Arguin**[14], **T. Argyropoulos**[9], **M. Arik**[18a], **A.J. Armbruster**[87], **O. Arnaez**[4], **C. Arnault**[115], **A. Artamonov**[95], **D. Arutinov**[20], **M. Asai**[143], **S. Asai**[154], **R. Asfandiyarov**[171], **S. Ask**[82], **B. Åsman**[145a,145b], **D. Asner**[28], **L. Asquith**[77], **K. Assamagan**[24], **A. Astbury**[168], **A. Astvatsatourov**[52], **G. Atoian**[174], **B. Auerbach**[174], **K. Augsten**[127], **M. Aurousseau**[4], **N. Austin**[73], **G. Avolio**[162], **R. Avramidou**[9], **D. Axen**[167], **C. Ay**[54], **G. Azuelos**[93,b], **Y. Azuma**[154], **M.A. Baak**[29], **A.M. Bach**[14], **H. Bachacou**[136], **K. Bachas**[29], **M. Backes**[49], **E. Badescu**[25a], **P. Bagnaia**[132a,132b], **Y. Bai**[32a], **T. Bain**[157], **J.T. Baines**[129], **O.K. Baker**[174], **M.D. Baker**[24], **S. Baker**[77], **F. Baltasar Dos Santos Pedrosa**[29], **E. Banas**[38], **P. Banerjee**[93], **S. Banerjee**[168], **D. Banfi**[89a,89b], **A. Bangert**[137], **V. Bansal**[168], **S.P. Baranov**[94], **S. Baranov**[65], **A. Barashkou**[65], **T. Barber**[27], **E.L. Barberio**[86], **D. Barberis**[50a,50b], **M. Barbero**[20], **D.Y. Bardin**[65], **T. Barillari**[99], **M. Barisonzi**[173], **T. Barklow**[143], **N. Barlow**[27], **B.M. Barnett**[129], **R.M. Barnett**[14], **A. Baroncelli**[134a], **A.J. Barr**[118], **F. Barreiro**[80], **J. Barreiro Guimarães da Costa**[57], **P. Barrillon**[115], **R. Bartoldus**[143], **D. Bartsch**[20], **R.L. Bates**[53], **L. Batkova**[144a], **J.R. Batley**[27], **A. Battaglia**[16], **M. Battistin**[29], **F. Bauer**[136], **H.S. Bawa**[143], **M. Bazalova**[125], **B. Beare**[157], **T. Beau**[78], **P.H. Beauchemin**[118], **R. Beccherle**[50a], **N. Becerici**[18a], **P. Bechtle**[41], **G.A. Beck**[75], **H.P. Beck**[16], **M. Beckingham**[48], **K.H. Becks**[173], **A.J. Beddall**[18c], **A. Beddall**[18c], **V.A. Bednyakov**[65], **C. Bee**[83], **M. Begel**[24], **S. Behar Harpaz**[151], **P.K. Behera**[63], **M. Beimforde**[99], **C. Belanger-Champagne**[165], **P.J. Bell**[49], **W.H. Bell**[49], **G. Bella**[152], **L. Bellagamba**[19a], **F. Bellina**[29], **M. Bellomo**[119a], **A. Belloni**[57], **K. Belotskiy**[96], **O. Beltramello**[29], **S. Ben Ami**[151], **O. Benary**[152], **D. Benchekroun**[135a], **M. Bendel**[81], **B.H. Benedict**[162], **N. Benekos**[164], **Y. Benhammou**[152], **G.P. Benincasa**[124a], **D.P. Benjamin**[44], **M. Benoit**[115], **J.R. Bensinger**[22], **K. Benslama**[130], **S. Bentvelsen**[105], **M. Beretta**[47], **D. Berge**[29], **E. Bergeaas Kuutmann**[41], **N. Berger**[4], **F. Berghaus**[168], **E. Berglund**[49], **J. Beringer**[14], **P. Bernat**[115], **R. Bernhard**[48], **C. Bernius**[77], **T. Berry**[76], **A. Bertin**[19a,19b], **M.I. Besana**[89a,89b], **N. Besson**[136], **S. Bethke**[99], **R.M. Bianchi**[48], **M. Bianco**[72a,72b], **O. Biebel**[98], **J. Biesiada**[14], **M. Biglietti**[132a,132b], **H. Bilokon**[47], **M. Bindi**[19a,19b], **S. Binet**[115], **A. Bingul**[18c], **C. Bini**[132a,132b], **C. Biscarat**[179], **U. Bitenc**[48], **K.M. Black**[57], **R.E. Blair**[5], **J.-B. Blanchard**[115], **G. Blanchot**[29], **C. Blocker**[22], **A. Blondel**[49], **W. Blum**[81], **U. Blumenschein**[54], **G.J. Bobbink**[105], **A. Bocci**[44], **M. Boehler**[41], **J. Boek**[173], **N. Boelaert**[79], **S. Böser**[77], **J.A. Bogaerts**[29], **A. Bogouch**[90,*], **C. Bohm**[145a], **J. Bohm**[125], **V. Boisvert**[76], **T. Bold**[162c], **V. Boldea**[25a], **V.G. Bondarenko**[96], **M. Bondioli**[162], **M. Boonekamp**[136], **S. Bordoni**[78], **C. Borer**[16], **A. Borisov**[128], **G. Borissov**[71], **I. Borjanovic**[72a], **S. Borroni**[132a,132b], **K. Bos**[105], **D. Boscherini**[19a], **M. Bosman**[11], **H. Boterenbrood**[105], **J. Bouchami**[93], **J. Boudreau**[123], **E.V. Bouhova-Thacker**[71], **C. Boulahouache**[123], **C. Bourdarios**[115], **A. Boveia**[30], **J. Boyd**[29], **I.R. Boyko**[65], **I. Bozovic-Jelisavcic**[12b], **J. Bracinik**[17], **A. Braem**[29], **P. Branchini**[134a], **G.W. Brandenburg**[57], **A. Brandt**[7], **G. Brandt**[41], **O. Brandt**[54], **U. Bratzler**[155], **B. Brau**[84], **J.E. Brau**[114], **H.M. Braun**[173], **B. Brelier**[157], **J. Bremer**[29], **R. Brenner**[165], **S. Bressler**[151], **D. Britton**[53], **F.M. Brochu**[27], **I. Brock**[20], **R. Brock**[88], **E. Brodet**[152], **C. Bromberg**[88], **G. Brooijmans**[34], **W.K. Brooks**[31b], **G. Brown**[82], **P.A. Bruckman de Renstrom**[38], **D. Bruncko**[144b], **R. Bruneliere**[48], **S. Brunet**[41], **A. Bruni**[19a], **G. Bruni**[19a], **M. Bruschi**[19a], **F. Bucci**[49], **J. Buchanan**[118], **P. Buchholz**[141], **A.G. Buckley**[45], **I.A. Budagov**[65], **B. Budick**[108], **V. Büscher**[81], **L. Bugge**[117], **O. Bulekov**[96], **M. Bunse**[42], **T. Buran**[117], **H. Burckhart**[29], **S. Burdin**[73], **T. Burgess**[13],

S. Burke[129], E. Busato[33], P. Bussey[53], C.P. Buszello[165], F. Butin[29], B. Butler[143], J.M. Butler[21], C.M. Buttar[53], J.M. Butterworth[77], T. Byatt[77], J. Caballero[24], S. Cabrera Urbán[166], D. Caforio[19a,19b], O. Cakir[3a], P. Calafiura[14], G. Calderini[78], P. Calfayan[98], R. Calkins[106a], L.P. Caloba[23a], D. Calvet[33], P. Camarri[133a,133b], D. Cameron[117], S. Campana[29], M. Campanelli[77], V. Canale[102a,102b], F. Canelli[30], A. Canepa[158a], J. Cantero[80], L. Capasso[102a,102b], M.D.M. Capeans Garrido[29], I. Caprini[25a], M. Caprini[25a], M. Capua[36a,36b], R. Caputo[147], C. Caramarcu[25a], R. Cardarelli[133a], T. Carli[29], G. Carlino[102a], L. Carminati[89a,89b], B. Caron[2,b], S. Caron[48], G.D. Carrillo Montoya[171], S. Carron Montero[157], A.A. Carter[75], J.R. Carter[27], J. Carvalho[124a], D. Casadei[108], M.P. Casado[11], M. Cascella[122a,122b], A.M. Castaneda Hernandez[171], E. Castaneda-Miranda[171], V. Castillo Gimenez[166], N.F. Castro[124b], G. Cataldi[72a], A. Catinaccio[29], J.R. Catmore[71], A. Cattai[29], G. Cattani[133a,133b], S. Caughron[34], D. Cauz[163a,163c], P. Cavalleri[78], D. Cavalli[89a], M. Cavalli-Sforza[11], V. Cavasinni[122a,122b], F. Ceradini[134a,134b], A.S. Cerqueira[23a], A. Cerri[29], L. Cerrito[75], F. Cerutti[47], S.A. Cetin[18b], A. Chafaq[135a], D. Chakraborty[106a], K. Chan[2], J.D. Chapman[27], J.W. Chapman[87], E. Chareyre[78], D.G. Charlton[17], V. Chavda[82], S. Cheatham[71], S. Chekanov[5], S.V. Chekulaev[158a], G.A. Chelkov[65], H. Chen[24], S. Chen[32c], X. Chen[171], A. Cheplakov[65], V.F. Chepurnov[65], R. Cherkaoui El Moursli[135d], V. Tcherniatine[24], D. Chesneanu[25a], E. Cheu[6], S.L. Cheung[157], L. Chevalier[136], F. Chevallier[136], V. Chiarella[47], G. Chiefari[102a,102b], L. Chikovani[51], J.T. Childers[58a], A. Chilingarov[71], G. Chiodini[72a], V. Chizhov[65], G. Choudalakis[30], S. Chouridou[137], I.A. Christidi[77], A. Christov[48], D. Chromek-Burckhart[29], M.L. Chu[150], J. Chudoba[125], G. Ciapetti[132a,132b], A.K. Ciftci[3a], R. Ciftci[3a], D. Cinca[33], V. Cindro[74], M.D. Ciobotaru[162], C. Ciocca[19a,19b], A. Ciocio[14], M. Cirilli[87], M. Citterio[89a], A. Clark[49], P.J. Clark[45], W. Cleland[123], J.C. Clemens[83], B. Clement[55], C. Clement[145a,145b], Y. Coadou[83], M. Cobal[163a,163c], A. Coccaro[50a,50b], J. Cochran[64], J. Coggeshall[164], E. Cogneras[179], A.P. Colijn[105], C. Collard[115], N.J. Collins[17], C. Collins-Tooth[53], J. Collot[55], G. Colon[84], P. Conde Muiño[124a], E. Coniavitis[165], M. Consonni[104], S. Constantinescu[25a], C. Conta[119a,119b], F. Conventi[102a,d], M. Cooke[34], B.D. Cooper[75], A.M. Cooper-Sarkar[118], N.J. Cooper-Smith[76], K. Copic[34], T. Cornelissen[50a,50b], M. Corradi[19a], F. Corriveau[85,e], A. Corso-Radu[162], A. Cortes-Gonzalez[164], G. Cortiana[99], G. Costa[89a], M.J. Costa[166], D. Costanzo[139], T. Costin[30], D. Côté[41], R. Coura Torres[23a], L. Courneyea[168], G. Cowan[76], C. Cowden[27], B.E. Cox[82], K. Cranmer[108], J. Cranshaw[5], M. Cristinziani[20], G. Crosetti[36a,36b], R. Crupi[72a,72b], S. Crépé-Renaudin[55], C. Cuenca Almenar[174], T. Cuhadar Donszelmann[139], M. Curatolo[47], C.J. Curtis[17], P. Cwetanski[61], Z. Czyczula[174], S. D'Auria[53], M. D'Onofrio[73], A. D'Orazio[99], C. Da Via[82], W. Dabrowski[37], T. Dai[87], C. Dallapiccola[84], S.J. Dallison[129,*], C.H. Daly[138], M. Dam[35], H.O. Danielsson[29], D. Dannheim[99], V. Dao[49], G. Darbo[50a], G.L. Darlea[25b], W. Davey[86], T. Davidek[126], N. Davidson[86], R. Davidson[71], M. Davies[93], A.R. Davison[77], I. Dawson[139], R.K. Daya[39], K. De[7], R. de Asmundis[102a], S. De Castro[19a,19b], P.E. De Castro Faria Salgado[24], S. De Cecco[78], J. de Graat[98], N. De Groot[104], P. de Jong[105], L. De Mora[71], M. De Oliveira Branco[29], D. De Pedis[132a], A. De Salvo[132a], U. De Sanctis[163a,163c], A. De Santo[148], J.B. De Vivie De Regie[115], G. De Zorzi[132a,132b], S. Dean[77], D.V. Dedovich[65], J. Degenhardt[120], M. Dehchar[118], C. Del Papa[163a,163c], J. Del Peso[80], T. Del Prete[122a,122b], A. Dell'Acqua[29], L. Dell'Asta[89a,89b], M. Della Pietra[102a,d], D. della Volpe[102a,102b], M. Delmastro[29], P.A. Delsart[55], C. Deluca[147], S. Demers[174], M. Demichev[65], B. Demirkoz[11], J. Deng[162], W. Deng[24], S.P. Denisov[128], J.E. Derkaoui[135c], F. Derue[78], P. Dervan[73], K. Desch[20], P.O. Deviveiros[157], A. Dewhurst[129], B. DeWilde[147], S. Dhaliwal[157], R. Dhullipudi[24,f], A. Di Ciaccio[133a,133b], L. Di Ciaccio[4], A. Di Domenico[132a,132b], A. Di Girolamo[29], B. Di Girolamo[29], S. Di Luise[134a,134b], A. Di Mattia[88], R. Di Nardo[133a,133b], A. Di Simone[133a,133b], R. Di Sipio[19a,19b], M.A. Diaz[31a], F. Diblen[18c], E.B. Diehl[87], J. Dietrich[48], T.A. Dietzsch[58a], S. Diglio[115], K. Dindar Yagci[39], J. Dingfelder[48], C. Dionisi[132a,132b], P. Dita[25a], S. Dita[25a], F. Dittus[29], F. Djama[83], R. Djilkibaev[108], T. Djobava[51], M.A.B. do Vale[23a], A. Do Valle Wemans[124a], T.K.O. Doan[4], D. Dobos[29], E. Dobson[29], M. Dobson[162], C. Doglioni[118], T. Doherty[53], J. Dolejsi[126], I. Dolenc[74], Z. Dolezal[126], B.A. Dolgoshein[96], T. Dohmae[154], M. Donega[120], J. Donini[55], J. Dopke[173], A. Doria[102a], A. Dos Anjos[171], A. Dotti[122a,122b], M.T. Dova[70], A. Doxiadis[105], A.T. Doyle[53], Z. Drasal[126], M. Dris[9], J. Dubbert[99], E. Duchovni[170], G. Duckeck[98], A. Dudarev[29], F. Dudziak[115], M. Dührssen[29], L. Duflot[115], M.-A. Dufour[85], M. Dunford[30], H. Duran Yildiz[3b], A. Dushkin[22], R. Duxfield[139], M. Dwuznik[37], M. Düren[52], W.L. Ebenstein[44], J. Ebke[98], S. Eckweiler[81], K. Edmonds[81], C.A. Edwards[76], K. Egorov[61], W. Ehrenfeld[41], T. Ehrich[99], T. Eifert[29], G. Eigen[13], K. Einsweiler[14], E. Eisenhandler[75], T. Ekelof[165], M. El Kacimi[4], M. Ellert[165], S. Elles[4], F. Ellinghaus[81], K. Ellis[75], N. Ellis[29], J. Elmsheuser[98], M. Elsing[29], D. Emeliyanov[129], R. Engelmann[147], A. Engl[98], B. Epp[62], A. Eppig[87], J. Erdmann[54], A. Ereditato[16], D. Eriksson[145a], I. Ermoline[88], J. Ernst[1], M. Ernst[24], J. Ernwein[136], D. Errede[164], S. Errede[164], E. Ertel[81], M. Escalier[115], C. Escobar[166], X. Espinal Curull[11], B. Esposito[47], A.I. Etienvre[136], E. Etzion[152], H. Evans[61], L. Fabbri[19a,19b], C. Fabre[29], K. Facius[35], R.M. Fakhrutdinov[128], S. Falciano[132a], Y. Fang[171],

M. Fanti[89a,89b], A. Farbin[7], A. Farilla[134a], J. Farley[147], T. Farooque[157], S.M. Farrington[118], P. Farthouat[29], P. Fassnacht[29], D. Fassouliotis[8], B. Fatholahzadeh[157], L. Fayard[115], F. Fayette[54], R. Febbraro[33], P. Federic[144a], O.L. Fedin[121], W. Fedorko[29], L. Feligioni[83], C.U. Felzmann[86], C. Feng[32d], E.J. Feng[30], A.B. Fenyuk[128], J. Ferencei[144b], J. Ferland[93], B. Fernandes[124a], W. Fernando[109], S. Ferrag[53], J. Ferrando[118], V. Ferrara[41], A. Ferrari[165], P. Ferrari[105], R. Ferrari[119a], A. Ferrer[166], M.L. Ferrer[47], D. Ferrere[49], C. Ferretti[87], M. Fiascaris[118], F. Fiedler[81], A. Filipčič[74], A. Filippas[9], F. Filthaut[104], M. Fincke-Keeler[168], M.C.N. Fiolhais[124a], L. Fiorini[11], A. Firan[39], G. Fischer[41], M.J. Fisher[109], M. Flechl[165], I. Fleck[141], J. Fleckner[81], P. Fleischmann[172], S. Fleischmann[20], T. Flick[173], L.R. Flores Castillo[171], M.J. Flowerdew[99], T. Fonseca Martin[76], A. Formica[136], A. Forti[82], D. Fortin[158a], D. Fournier[115], A.J. Fowler[44], K. Fowler[137], H. Fox[71], P. Francavilla[122a,122b], S. Franchino[119a,119b], D. Francis[29], M. Franklin[57], S. Franz[29], M. Fraternali[119a,119b], S. Fratina[120], J. Freestone[82], S.T. French[27], R. Froeschl[29], D. Froidevaux[29], J.A. Frost[27], C. Fukunaga[155], E. Fullana Torregrosa[5], J. Fuster[166], C. Gabaldon[80], O. Gabizon[170], T. Gadfort[24], S. Gadomski[49], G. Gagliardi[50a,50b], P. Gagnon[61], C. Galea[98], E.J. Gallas[118], M.V. Gallas[29], V. Gallo[16], B.J. Gallop[129], P. Gallus[125], E. Galyaev[40], K.K. Gan[109], Y.S. Gao[143,g], A. Gaponenko[14], M. Garcia-Sciveres[14], C. García[166], J.E. García Navarro[49], R.W. Gardner[30], N. Garelli[29], H. Garitaonandia[105], V. Garonne[29], C. Gatti[47], G. Gaudio[119a], V. Gautard[136], P. Gauzzi[132a,132b], I.L. Gavrilenko[94], C. Gay[167], G. Gaycken[20], E.N. Gazis[9], P. Ge[32d], C.N.P. Gee[129], Ch. Geich-Gimbel[20], K. Gellerstedt[145a,145b], C. Gemme[50a], M.H. Genest[98], S. Gentile[132a,132b], F. Georgatos[9], S. George[76], A. Gershon[152], H. Ghazlane[135d], N. Ghodbane[33], B. Giacobbe[19a], S. Giagu[132a,132b], V. Giakoumopoulou[8], V. Giangiobbe[122a,122b], F. Gianotti[29], B. Gibbard[24], A. Gibson[157], S.M. Gibson[118], L.M. Gilbert[118], M. Gilchriese[14], V. Gilewsky[91], D.M. Gingrich[2,b], J. Ginzburg[152], N. Giokaris[8], M.P. Giordani[163a,163c], R. Giordano[102a,102b], F.M. Giorgi[15], P. Giovannini[99], P.F. Giraud[29], P. Girtler[62], D. Giugni[89a], P. Giusti[19a], B.K. Gjelsten[117], L.K. Gladilin[97], C. Glasman[80], A. Glazov[41], K.W. Glitza[173], G.L. Glonti[65], J. Godfrey[142], J. Godlewski[29], M. Goebel[41], T. Göpfert[43], C. Goeringer[81], C. Gössling[42], T. Göttfert[99], V. Goggi[119a,119b,,h], S. Goldfarb[87], D. Goldin[39], T. Golling[174], A. Gomes[124a], L.S. Gomez Fajardo[41], R. Gonçalo[76], L. Gonella[20], C. Gong[32b], S. González de la Hoz[166], M.L. Gonzalez Silva[26], S. Gonzalez-Sevilla[49], J.J. Goodson[147], L. Goossens[29], H.A. Gordon[24], I. Gorelov[103], G. Gorfine[173], B. Gorini[29], E. Gorini[72a,72b], A. Gorišek[74], E. Gornicki[38], B. Gosdzik[41], M. Gosselink[105], M.I. Gostkin[65], I. Gough Eschrich[162], M. Gouighri[135a], D. Goujdami[135a], M.P. Goulette[49], A.G. Goussiou[138], C. Goy[4], I. Grabowska-Bold[162,c], P. Grafström[29], K.-J. Grahn[146], S. Grancagnolo[15], V. Grassi[147], V. Gratchev[121], N. Grau[34], H.M. Gray[34,i], J.A. Gray[147], E. Graziani[134a], B. Green[76], T. Greenshaw[73], Z.D. Greenwood[24,f], I.M. Gregor[41], P. Grenier[143], E. Griesmayer[46], J. Griffiths[138], N. Grigalashvili[65], A.A. Grillo[137], K. Grimm[147], S. Grinstein[11], Y.V. Grishkevich[97], M. Groh[99], M. Groll[81], E. Gross[170], J. Grosse-Knetter[54], J. Groth-Jensen[79], K. Grybel[141], C. Guicheney[33], A. Guida[72a,72b], T. Guillemin[4], H. Guler[85,j], J. Gunther[125], B. Guo[157], A. Gupta[30], Y. Gusakov[65], A. Gutierrez[93], P. Gutierrez[111], N. Guttman[152], O. Gutzwiller[171], C. Guyot[136], C. Gwenlan[118], C.B. Gwilliam[73], A. Haas[143], S. Haas[29], C. Haber[14], H.K. Hadavand[39], D.R. Hadley[17], P. Haefner[99], R. Härtel[99], Z. Hajduk[38], H. Hakobyan[175], J. Haller[41,k], K. Hamacher[173], A. Hamilton[49], S. Hamilton[160], L. Han[32b], K. Hanagaki[116], M. Hance[120], C. Handel[81], P. Hanke[58a], J.R. Hansen[35], J.B. Hansen[35], J.D. Hansen[35], P.H. Hansen[35], T. Hansl-Kozanecka[137], P. Hansson[143], K. Hara[159], G.A. Hare[137], T. Harenberg[173], R.D. Harrington[21], O.M. Harris[138], K. Harrison[17], J. Hartert[48], F. Hartjes[105], A. Harvey[56], S. Hasegawa[101], Y. Hasegawa[140], K. Hashemi[22], S. Hassani[136], S. Haug[16], M. Hauschild[29], R. Hauser[88], M. Havranek[125], C.M. Hawkes[17], R.J. Hawkings[29], T. Hayakawa[67], H.S. Hayward[73], S.J. Haywood[129], S.J. Head[82], V. Hedberg[79], L. Heelan[28], S. Heim[88], B. Heinemann[14], S. Heisterkamp[35], L. Helary[4], M. Heller[115], S. Hellman[145a,145b], C. Helsens[11], T. Hemperek[20], R.C.W. Henderson[71], M. Henke[58a], A. Henrichs[54], A.M. Henriques Correia[29], S. Henrot-Versille[115], C. Hensel[54], T. Henß[173], Y. Hernández Jiménez[166], A.D. Hershenhorn[151], G. Herten[48], R. Hertenberger[98], L. Hervas[29], N.P. Hessey[105], E. Higón-Rodriguez[166], J.C. Hill[27], K.H. Hiller[41], S. Hillert[145a,145b], S.J. Hillier[17], I. Hinchliffe[14], E. Hines[120], M. Hirose[116], F. Hirsch[42], D. Hirschbuehl[173], J. Hobbs[147], N. Hod[152], M.C. Hodgkinson[139], P. Hodgson[139], A. Hoecker[29], M.R. Hoeferkamp[103], J. Hoffman[39], D. Hoffmann[83], M. Hohlfeld[81], T. Holy[127], J.L. Holzbauer[88], Y. Homma[67], T. Horazdovsky[127], T. Hori[67], C. Horn[143], S. Horner[48], S. Horvat[99], J.-Y. Hostachy[55], S. Hou[150], A. Hoummada[135a], T. Howe[39], J. Hrivnac[115], T. Hryn'ova[4], P.J. Hsu[174], S.-C. Hsu[14], G.S. Huang[111], Z. Hubacek[127], F. Hubaut[83], F. Huegging[20], E.W. Hughes[34], G. Hughes[71], M. Hurwitz[30], U. Husemann[41], N. Huseynov[10], J. Huston[88], J. Huth[57], G. Iacobucci[102a], G. Iakovidis[9], I. Ibragimov[141], L. Iconomidou-Fayard[115], J. Idarraga[158b], P. Iengo[4], O. Igonkina[105], Y. Ikegami[66], M. Ikeno[66], Y. Ilchenko[39], D. Iliadis[153], T. Ince[168], P. Ioannou[8], M. Iodice[134a], A. Irles Quiles[166], A. Ishikawa[67], M. Ishino[66], R. Ishmukhametov[39], T. Isobe[154], V. Issakov[174,*], C. Issever[118],

S. Istin[18a], Y. Itoh[101], A.V. Ivashin[128], W. Iwanski[38], H. Iwasaki[66], J.M. Izen[40], V. Izzo[102a], B. Jackson[120], J.N. Jackson[73], P. Jackson[143], M.R. Jaekel[29], V. Jain[61], K. Jakobs[48], S. Jakobsen[35], J. Jakubek[127], D.K. Jana[111], E. Jansen[104], A. Jantsch[99], M. Janus[48], R.C. Jared[171], G. Jarlskog[79], L. Jeanty[57], I. Jen-La Plante[30], P. Jenni[29], P. Jez[35], S. Jézéquel[4], W. Ji[79], J. Jia[147], Y. Jiang[32b], M. Jimenez Belenguer[29], S. Jin[32a], O. Jinnouchi[156], D. Joffe[39], M. Johansen[145a,145b], K.E. Johansson[145a], P. Johansson[139], S. Johnert[41], K.A. Johns[6], K. Jon-And[145a,145b], G. Jones[82], R.W.L. Jones[71], T.J. Jones[73], P.M. Jorge[124a], J. Joseph[14], V. Juranek[125], P. Jussel[62], V.V. Kabachenko[128], M. Kaci[166], A. Kaczmarska[38], M. Kado[115], H. Kagan[109], M. Kagan[57], S. Kaiser[99], E. Kajomovitz[151], S. Kalinin[173], L.V. Kalinovskaya[65], A. Kalinowski[130], S. Kama[41], N. Kanaya[154], M. Kaneda[154], V.A. Kantserov[96], J. Kanzaki[66], B. Kaplan[174], A. Kapliy[30], J. Kaplon[29], D. Kar[43], M. Karagounis[20], M. Karagoz Unel[118], V. Kartvelishvili[71], A.N. Karyukhin[128], L. Kashif[57], A. Kasmi[39], R.D. Kass[109], A. Kastanas[13], M. Kastoryano[174], M. Kataoka[4], Y. Kataoka[154], E. Katsoufis[9], J. Katzy[41], V. Kaushik[6], K. Kawagoe[67], T. Kawamoto[154], G. Kawamura[81], M.S. Kayl[105], F. Kayumov[94], V.A. Kazanin[107], M.Y. Kazarinov[65], J.R. Keates[82], R. Keeler[168], P.T. Keener[120], R. Kehoe[39], M. Keil[54], G.D. Kekelidze[65], M. Kelly[82], M. Kenyon[53], O. Kepka[125], N. Kerschen[29], B.P. Kerševan[74], S. Kersten[173], K. Kessoku[154], M. Khakzad[28], F. Khalil-zada[10], H. Khandanyan[164], A. Khanov[112], D. Kharchenko[65], A. Khodinov[147], A. Khomich[58a], G. Khoriauli[20], N. Khovanskiy[65], V. Khovanskiy[95], E. Khramov[65], J. Khubua[51], H. Kim[7], M.S. Kim[2], P.C. Kim[143], S.H. Kim[159], O. Kind[15], P. Kind[173], B.T. King[73], J. Kirk[129], G.P. Kirsch[118], L.E. Kirsch[22], A.E. Kiryunin[99], D. Kisielewska[37], T. Kittelmann[123], H. Kiyamura[67], E. Kladiva[144b], M. Klein[73], U. Klein[73], K. Kleinknecht[81], M. Klemetti[85], A. Klier[170], A. Klimentov[24], R. Klingenberg[42], E.B. Klinkby[44], T. Klioutchnikova[29], P.F. Klok[104], S. Klous[105], E.-E. Kluge[58a], T. Kluge[73], P. Kluit[105], M. Klute[54], S. Kluth[99], N.S. Knecht[157], E. Kneringer[62], B.R. Ko[44], T. Kobayashi[154], M. Kobel[43], B. Koblitz[29], M. Kocian[143], A. Kocnar[113], P. Kodys[126], K. Köneke[41], A.C. König[104], S. Koenig[81], L. Köpke[81], F. Koetsveld[104], P. Koevesarki[20], T. Koffas[29], E. Koffeman[105], F. Kohn[54], Z. Kohout[127], T. Kohriki[66], H. Kolanoski[15], V. Kolesnikov[65], I. Koletsou[4], J. Koll[88], D. Kollar[29], S. Kolos[162,l], S.D. Kolya[82], A.A. Komar[94], J.R. Komaragiri[142], T. Kondo[66], T. Kono[41,k], R. Konoplich[108], S.P. Konovalov[94], N. Konstantinidis[77], S. Koperny[37], K. Korcyl[38], K. Kordas[153], A. Korn[14], I. Korolkov[11], E.V. Korolkova[139], V.A. Korotkov[128], O. Kortner[99], P. Kostka[41], V.V. Kostyukhin[20], S. Kotov[99], V.M. Kotov[65], K.Y. Kotov[107], C. Kourkoumelis[8], A. Koutsman[105], R. Kowalewski[168], H. Kowalski[41], T.Z. Kowalski[37], W. Kozanecki[136], A.S. Kozhin[128], V. Kral[127], V.A. Kramarenko[97], G. Kramberger[74], M.W. Krasny[78], A. Krasznahorkay[108], A. Kreisel[152], F. Krejci[127], J. Kretzschmar[73], N. Krieger[54], P. Krieger[157], K. Kroeninger[54], H. Kroha[99], J. Kroll[120], J. Kroseberg[20], J. Krstic[12a], U. Kruchonak[65], H. Krüger[20], Z.V. Krumshteyn[65], T. Kubota[154], S. Kuehn[48], A. Kugel[58c], T. Kuhl[173], D. Kuhn[62], V. Kukhtin[65], Y. Kulchitsky[90], S. Kuleshov[31b], C. Kummer[98], M. Kuna[83], J. Kunkle[120], A. Kupco[125], H. Kurashige[67], M. Kurata[159], L.L. Kurchaninov[158a], Y.A. Kurochkin[90], V. Kus[125], R. Kwee[15], L. La Rotonda[36a,36b], J. Labbe[4], C. Lacasta[166], F. Lacava[132a,132b], H. Lacker[15], D. Lacour[78], V.R. Lacuesta[166], E. Ladygin[65], R. Lafaye[4], B. Laforge[78], T. Lagouri[80], S. Lai[48], M. Lamanna[29], C.L. Lampen[6], W. Lampl[6], E. Lancon[136], U. Landgraf[48], M.P.J. Landon[75], J.L. Lane[82], A.J. Lankford[162], F. Lanni[24], K. Lantzsch[29], A. Lanza[119a], S. Laplace[4], C. Lapoire[83], J.F. Laporte[136], T. Lari[89a], A. Larner[118], M. Lassnig[29], P. Laurelli[47], W. Lavrijsen[14], P. Laycock[73], A.B. Lazarev[65], A. Lazzaro[89a,89b], O. Le Dortz[78], E. Le Guirriec[83], E. Le Menedeu[136], M. Le Vine[24], A. Lebedev[64], C. Lebel[93], T. LeCompte[5], F. Ledroit-Guillon[55], H. Lee[105], J.S.H. Lee[149], S.C. Lee[150], M. Lefebvre[168], M. Legendre[136], B.C. LeGeyt[120], F. Legger[98], C. Leggett[14], M. Lehmacher[20], G. Lehmann Miotto[29], X. Lei[6], R. Leitner[126], D. Lellouch[170], J. Lellouch[78], V. Lendermann[58a], K.J.C. Leney[73], T. Lenz[173], G. Lenzen[173], B. Lenzi[136], K. Leonhardt[43], C. Leroy[93], J.-R. Lessard[168], C.G. Lester[27], A. Leung Fook Cheong[171], J. Levêque[83], D. Levin[87], L.J. Levinson[170], M. Leyton[14], H. Li[171], S. Li[41], X. Li[87], Z. Liang[39], Z. Liang[150,m], B. Liberti[133a], P. Lichard[29], M. Lichtnecker[98], K. Lie[164], W. Liebig[105], J.N. Lilley[17], H. Lim[5], A. Limosani[86], M. Limper[63], S.C. Lin[150], J.T. Linnemann[88], E. Lipeles[120], L. Lipinsky[125], A. Lipniacka[13], T.M. Liss[164], D. Lissauer[24], A. Lister[49], A.M. Litke[137], C. Liu[28], D. Liu[150,n], H. Liu[87], J.B. Liu[87], M. Liu[32b], T. Liu[39], Y. Liu[32b], M. Livan[119a,119b], A. Lleres[55], S.L. Lloyd[75], E. Lobodzinska[41], P. Loch[6], W.S. Lockman[137], S. Lockwitz[174], T. Loddenkoetter[20], F.K. Loebinger[82], A. Loginov[174], C.W. Loh[167], T. Lohse[15], K. Lohwasser[48], M. Lokajicek[125], R.E. Long[71], L. Lopes[124a], D. Lopez Mateos[34,i], M. Losada[161], P. Loscutoff[14], X. Lou[40], A. Lounis[115], K.F. Loureiro[109], L. Lovas[144a], J. Love[21], P.A. Love[71], A.J. Lowe[61], F. Lu[32a], H.J. Lubatti[138], C. Luci[132a,132b], A. Lucotte[55], A. Ludwig[43], D. Ludwig[41], I. Ludwig[48], F. Luehring[61], L. Luisa[163a,163c], D. Lumb[48], L. Luminari[132a], E. Lund[117], B. Lund-Jensen[146], B. Lundberg[79], J. Lundberg[29], J. Lundquist[35], D. Lynn[24], J. Lys[14], E. Lytken[79], H. Ma[24], L.L. Ma[171], J.A. Macana Goia[93], G. Maccarrone[47], A. Macchiolo[99], B. Maček[74], J. Machado Miguens[124a], R. Mackeprang[35], R.J. Madaras[14], W.F. Mader[43], R. Maenner[58c], T. Maeno[24],

P. Mättig[173], S. Mättig[41], P.J. Magalhaes Martins[124a], E. Magradze[51], Y. Mahalalel[152], K. Mahboubi[48], A. Mahmood[1], C. Maiani[132a,132b], C. Maidantchik[23a], A. Maio[124a], S. Majewski[24], Y. Makida[66], M. Makouski[128], N. Makovec[115], Pa. Malecki[38], P. Malecki[38], V.P. Maleev[121], F. Malek[55], U. Mallik[63], D. Malon[5], S. Maltezos[9], V. Malyshev[107], S. Malyukov[65], M. Mambelli[30], R. Mameghani[98], J. Mamuzic[41], L. Mandelli[89a], I. Mandić[74], R. Mandrysch[15], J. Maneira[124a], P.S. Mangeard[88], I.D. Manjavidze[65], P.M. Manning[137], A. Manousakis-Katsikakis[8], B. Mansoulie[136], A. Mapelli[29], L. Mapelli[29], L. March[80], J.F. Marchand[4], F. Marchese[133a,133b], G. Marchiori[78], M. Marcisovsky[125], C.P. Marino[61], F. Marroquim[23a], Z. Marshall[34,i], S. Marti-Garcia[166], A.J. Martin[75], A.J. Martin[174], B. Martin[29], B. Martin[88], F.F. Martin[120], J.P. Martin[93], T.A. Martin[17], B. Martin dit Latour[49], M. Martinez[11], V. Martinez Outschoorn[57], A. Martini[47], A.C. Martyniuk[82], F. Marzano[132a], A. Marzin[136], L. Masetti[20], T. Mashimo[154], R. Mashinistov[96], J. Masik[82], A.L. Maslennikov[107], I. Massa[19a,19b], N. Massol[4], A. Mastroberardino[36a,36b], T. Masubuchi[154], P. Matricon[115], H. Matsunaga[154], T. Matsushita[67], C. Mattravers[118,o], S.J. Maxfield[73], A. Mayne[139], R. Mazini[150], M. Mazur[48], M. Mazzanti[89a], J. McDonald[85], S.P. McKee[87], A. McCarn[164], R.L. McCarthy[147], N.A. McCubbin[129], K.W. McFarlane[56], H. McGlone[53], G. Mchedlidze[51], S.J. McMahon[129], R.A. McPherson[168,e], A. Meade[84], J. Mechnich[105], M. Mechtel[173], M. Medinnis[41], R. Meera-Lebbai[111], T.M. Meguro[116], S. Mehlhase[41], A. Mehta[73], K. Meier[58a], B. Meirose[48], C. Melachrinos[30], B.R. Mellado Garcia[171], L. Mendoza Navas[161], Z. Meng[150,p], S. Menke[99], E. Meoni[11], P. Mermod[118], L. Merola[102a,102b], C. Meroni[89a], F.S. Merritt[30], A.M. Messina[29], J. Metcalfe[103], A.S. Mete[64], J.-P. Meyer[136], J. Meyer[172], J. Meyer[54], T.C. Meyer[29], W.T. Meyer[64], J. Miao[32d], S. Michal[29], L. Micu[25a], R.P. Middleton[129], S. Migas[73], L. Mijović[74], G. Mikenberg[170], M. Mikestikova[125], M. Mikuž[74], D.W. Miller[143], W.J. Mills[167], C.M. Mills[57], A. Milov[170], D.A. Milstead[145a,145b], D. Milstein[170], A.A. Minaenko[128], M. Miñano[166], I.A. Minashvili[65], A.I. Mincer[108], B. Mindur[37], M. Mineev[65], Y. Ming[130], L.M. Mir[11], G. Mirabelli[132a], S. Misawa[24], S. Miscetti[47], A. Misiejuk[76], J. Mitrevski[137], V.A. Mitsou[166], P.S. Miyagawa[82], J.U. Mjörnmark[79], D. Mladenov[22], T. Moa[145a,145b], S. Moed[57], V. Moeller[27], K. Mönig[41], N. Möser[20], W. Mohr[48], S. Mohrdieck-Möck[99], R. Moles-Valls[166], J. Molina-Perez[29], J. Monk[77], E. Monnier[83], S. Montesano[89a,89b], F. Monticelli[70], R.W. Moore[2], C. Mora Herrera[49], A. Moraes[53], A. Morais[124a], J. Morel[54], G. Morello[36a,36b], D. Moreno[161], M. Moreno Llácer[166], P. Morettini[50a], M. Morii[57], A.K. Morley[86], G. Mornacchi[29], S.V. Morozov[96], J.D. Morris[75], H.G. Moser[99], M. Mosidze[51], J. Moss[109], R. Mount[143], E. Mountricha[136], S.V. Mouraviev[94], E.J.W. Moyse[84], M. Mudrinic[12b], F. Mueller[58a], J. Mueller[123], K. Mueller[20], T.A. Müller[98], D. Muenstermann[42], A. Muir[167], Y. Munwes[152], R. Murillo Garcia[162], W.J. Murray[129], I. Mussche[105], E. Musto[102a,102b], A.G. Myagkov[128], M. Myska[125], J. Nadal[11], K. Nagai[159], K. Nagano[66], Y. Nagasaka[60], A.M. Nairz[29], K. Nakamura[154], I. Nakano[110], H. Nakatsuka[67], G. Nanava[20], A. Napier[160], M. Nash[77,q], N.R. Nation[21], T. Nattermann[20], T. Naumann[41], G. Navarro[161], S.K. Nderitu[20], H.A. Neal[87], E. Nebot[80], P. Nechaeva[94], A. Negri[119a,119b], G. Negri[29], A. Nelson[64], T.K. Nelson[143], S. Nemecek[125], P. Nemethy[108], A.A. Nepomuceno[23a], M. Nessi[29], M.S. Neubauer[164], A. Neusiedl[81], R.M. Neves[108], P. Nevski[24], F.M. Newcomer[120], R.B. Nickerson[118], R. Nicolaidou[136], L. Nicolas[139], G. Nicoletti[47], B. Nicquevert[29], F. Niedercorn[115], J. Nielsen[137], A. Nikiforov[15], K. Nikolaev[65], I. Nikolic-Audit[78], K. Nikolopoulos[8], H. Nilsen[48], P. Nilsson[7], A. Nisati[132a], T. Nishiyama[67], R. Nisius[99], L. Nodulman[5], M. Nomachi[116], I. Nomidis[153], M. Nordberg[29], B. Nordkvist[145a,145b], D. Notz[41], J. Novakova[126], M. Nozaki[66], M. Nožička[41], I.M. Nugent[158a], A.-E. Nuncio-Quiroz[20], G. Nunes Hanninger[20], T. Nunnemann[98], E. Nurse[77], D.C. O'Neil[142], V. O'Shea[53], F.G. Oakham[28,b], H. Oberlack[99], A. Ochi[67], S. Oda[154], S. Odaka[66], J. Odier[83], H. Ogren[61], A. Oh[82], S.H. Oh[44], C.C. Ohm[145a,145b], T. Ohshima[101], H. Ohshita[140], T. Ohsugi[59], S. Okada[67], H. Okawa[162], Y. Okumura[101], T. Okuyama[154], A.G. Olchevski[65], M. Oliveira[124a], D. Oliveira Damazio[24], J. Oliver[57], E. Oliveira Garcia[166], D. Olivito[120], A. Olszewski[38], J. Olszowska[38], C. Omachi[67,r], A. Onofre[124a], P.U.E. Onyisi[30], C.J. Oram[158a], M.J. Oreglia[30], Y. Oren[152], D. Orestano[134a,134b], I. Orlov[107], C. Oropeza Barrera[53], R.S. Orr[157], E.O. Ortega[130], B. Osculati[50a,50b], R. Ospanov[120], C. Osuna[11], J.P. Ottersbach[105], F. Ould-Saada[117], A. Ouraou[136], Q. Ouyang[32a], M. Owen[82], S. Owen[139], A. Oyarzun[31b], V.E. Ozcan[77], K. Ozone[66], N. Ozturk[7], A. Pacheco Pages[11], C. Padilla Aranda[11], E. Paganis[139], C. Pahl[63], F. Paige[24], K. Pajchel[117], S. Palestini[29], D. Pallin[33], A. Palma[124a], J.D. Palmer[17], Y.B. Pan[171], E. Panagiotopoulou[9], B. Panes[31a], N. Panikashvili[87], S. Panitkin[24], D. Pantea[25a], M. Panuskova[125], V. Paolone[123], Th.D. Papadopoulou[9], S.J. Park[54], W. Park[24,s], M.A. Parker[27], S.I. Parker[14], F. Parodi[50a,50b], J.A. Parsons[34], U. Parzefall[48], E. Pasqualucci[132a], A. Passeri[134a], F. Pastore[134a,134b], Fr. Pastore[29], G. Pásztor[49,t], S. Pataraia[99], J.R. Pater[82], S. Patricelli[102a,102b], A. Patwa[24], T. Pauly[29], L.S. Peak[149], M. Pecsy[144a], M.I. Pedraza Morales[171], S.V. Peleganchuk[107], H. Peng[171], A. Penson[34], J. Penwell[61], M. Perantoni[23a], K. Perez[34,i], E. Perez Codina[11], M.T. Pérez García-Estañ[166], V. Perez Reale[34], L. Perini[89a,89b], H. Pernegger[29], R. Perrino[72a],

S. Persembe[3a], P. Perus[115], V.D. Peshekhonov[65], B.A. Petersen[29], T.C. Petersen[35], E. Petit[83], C. Petridou[153], E. Petrolo[132a], F. Petrucci[134a,134b], D. Petschull[41], M. Petteni[142], R. Pezoa[31b], A. Phan[86], A.W. Phillips[27], G. Piacquadio[29], M. Piccinini[19a,19b], R. Piegaia[26], J.E. Pilcher[30], A.D. Pilkington[82], J. Pina[124a], M. Pinamonti[163a,163c], J.L. Pinfold[2], B. Pinto[124a], C. Pizio[89a,89b], R. Placakyte[41], M. Plamondon[168], M.-A. Pleier[24], A. Poblaguev[174], S. Poddar[58a], F. Podlyski[33], P. Poffenberger[168], L. Poggioli[115], M. Pohl[49], F. Polci[55], G. Polesello[119a], A. Policicchio[138], A. Polini[19a], J. Poll[75], V. Polychronakos[24], D. Pomeroy[22], K. Pommès[29], P. Ponsot[136], L. Pontecorvo[132a], B.G. Pope[88], G.A. Popeneciu[25a], D.S. Popovic[12a], A. Poppleton[29], J. Popule[125], X. Portell Bueso[48], R. Porter[162], G.E. Pospelov[99], S. Pospisil[127], M. Potekhin[24], I.N. Potrap[99], C.J. Potter[148], C.T. Potter[85], K.P. Potter[82], G. Poulard[29], J. Poveda[171], R. Prabhu[20], P. Pralavorio[83], S. Prasad[57], R. Pravahan[7], L. Pribyl[29], D. Price[61], L.E. Price[5], P.M. Prichard[73], D. Prieur[123], M. Primavera[72a], K. Prokofiev[29], F. Prokoshin[31b], S. Protopopescu[24], J. Proudfoot[5], X. Prudent[43], H. Przysiezniak[4], S. Psoroulas[20], E. Ptacek[114], C. Puigdengoles[11], J. Purdham[87], M. Purohit[24,s], P. Puzo[115], Y. Pylypchenko[117], M. Qi[32c], J. Qian[87], W. Qian[129], Z. Qin[41], A. Quadt[54], D.R. Quarrie[14], W.B. Quayle[171], F. Quinonez[31a], M. Raas[104], V. Radeka[24], V. Radescu[58b], B. Radics[20], T. Rador[18a], F. Ragusa[89a,89b], G. Rahal[179], A.M. Rahimi[109], S. Rajagopalan[24], M. Rammensee[48], M. Rammes[141], F. Rauscher[98], E. Rauter[99], M. Raymond[29], A.L. Read[117], D.M. Rebuzzi[119a,119b], A. Redelbach[172], G. Redlinger[24], R. Reece[120], K. Reeves[40], E. Reinherz-Aronis[152], A. Reinsch[114], I. Reisinger[42], D. Reljic[12a], C. Rembser[29], Z.L. Ren[150], P. Renkel[39], S. Rescia[24], M. Rescigno[132a], S. Resconi[89a], B. Resende[136], P. Reznicek[126], R. Rezvani[157], A. Richards[77], R.A. Richards[88], R. Richter[99], E. Richter-Was[38,u], M. Ridel[78], M. Rijpstra[105], M. Rijssenbeek[147], A. Rimoldi[119a,119b], L. Rinaldi[19a], R.R. Rios[39], I. Riu[11], F. Rizatdinova[112], E. Rizvi[75], D.A. Roa Romero[161], S.H. Robertson[85,e], A. Robichaud-Veronneau[49], D. Robinson[27], J.E.M. Robinson[77], M. Robinson[114], A. Robson[53], J.G. Rocha de Lima[106a], C. Roda[122a,122b], D. Roda Dos Santos[29], D. Rodriguez[161], Y. Rodriguez Garcia[15], S. Roe[29], O. Røhne[117], V. Rojo[1], S. Rolli[160], A. Romaniouk[96], V.M. Romanov[65], G. Romeo[26], D. Romero Maltrana[31a], L. Roos[78], E. Ros[166], S. Rosati[138], G.A. Rosenbaum[157], L. Rosselet[49], V. Rossetti[11], L.P. Rossi[50a], M. Rotaru[25a], J. Rothberg[138], D. Rousseau[115], C.R. Royon[136], A. Rozanov[83], Y. Rozen[151], X. Ruan[115], B. Ruckert[98], N. Ruckstuhl[105], V.I. Rud[97], G. Rudolph[62], F. Rühr[58a], F. Ruggieri[134a], A. Ruiz-Martinez[64], L. Rumyantsev[65], Z. Rurikova[48], N.A. Rusakovich[65], J.P. Rutherfoord[6], C. Ruwiedel[20], P. Ruzicka[125], Y.F. Ryabov[121], P. Ryan[88], G. Rybkin[115], S. Rzaeva[10], A.F. Saavedra[149], H.F.-W. Sadrozinski[137], R. Sadykov[65], H. Sakamoto[154], G. Salamanna[105], A. Salamon[133a], M.S. Saleem[111], D. Salihagic[99], A. Salnikov[143], J. Salt[166], B.M. Salvachua Ferrando[5], D. Salvatore[36a,36b], F. Salvatore[148], A. Salvucci[47], A. Salzburger[29], D. Sampsonidis[153], B.H. Samset[117], H. Sandaker[13], H.G. Sander[81], M.P. Sanders[98], M. Sandhoff[173], P. Sandhu[157], R. Sandstroem[105], S. Sandvoss[173], D.P.C. Sankey[129], B. Sanny[173], A. Sansoni[47], C. Santamarina Rios[85], C. Santoni[33], R. Santonico[133a,133b], J.G. Saraiva[124a], T. Sarangi[171], E. Sarkisyan-Grinbaum[7], F. Sarri[122a,122b], O. Sasaki[66], N. Sasao[68], I. Satsounkevitch[90], G. Sauvage[4], P. Savard[157,b], A.Y. Savine[6], V. Savinov[123], L. Sawyer[24,f], D.H. Saxon[53], L.P. Says[33], C. Sbarra[19a,19b], A. Sbrizzi[19a,19b], D.A. Scannicchio[29], J. Schaarschmidt[43], P. Schacht[99], U. Schäfer[81], S. Schaetzel[58b], A.C. Schaffer[115], D. Schaile[98], R.D. Schamberger[147], A.G. Schamov[107], V.A. Schegelsky[121], D. Scheirich[87], M. Schernau[162], M.I. Scherzer[14], C. Schiavi[50a,50b], J. Schieck[99], M. Schioppa[36a,36b], S. Schlenker[29], E. Schmidt[48], K. Schmieden[20], C. Schmitt[81], M. Schmitz[20], M. Schott[29], D. Schouten[142], J. Schovancova[125], M. Schram[85], A. Schreiner[63], C. Schroeder[81], N. Schroer[58c], M. Schroers[173], J. Schultes[173], H.-C. Schultz-Coulon[58a], J.W. Schumacher[43], M. Schumacher[48], B.A. Schumm[137], Ph. Schune[136], C. Schwanenberger[82], A. Schwartzman[143], Ph. Schwemling[78], R. Schwienhorst[88], R. Schwierz[43], J. Schwindling[136], W.G. Scott[129], J. Searcy[114], E. Sedykh[121], E. Segura[11], S.C. Seidel[103], A. Seiden[137], F. Seifert[43], J.M. Seixas[23a], G. Sekhniaidze[102a], D.M. Seliverstov[121], B. Sellden[145a], N. Semprini-Cesari[19a,19b], C. Serfon[98], L. Serin[115], R. Seuster[99], H. Severini[111], M.E. Sevior[86], A. Sfyrla[164], E. Shabalina[54], M. Shamim[114], L.Y. Shan[32a], J.T. Shank[21], Q.T. Shao[86], M. Shapiro[14], P.B. Shatalov[95], K. Shaw[139], D. Sherman[29], P. Sherwood[77], A. Shibata[108], M. Shimojima[100], T. Shin[56], A. Shmeleva[94], M.J. Shochet[30], M.A. Shupe[6], P. Sicho[125], A. Sidoti[15], F. Siegert[77], J. Siegrist[14], Dj. Sijacki[12a], O. Silbert[170], J. Silva[124a], Y. Silver[152], D. Silverstein[143], S.B. Silverstein[145a], V. Simak[127], Lj. Simic[12a], S. Simion[115], B. Simmons[77], M. Simonyan[35], P. Sinervo[157], N.B. Sinev[114], V. Sipica[141], G. Siragusa[81], A.N. Sisakyan[65], S.Yu. Sivoklokov[97], J. Sjoelin[145a,145b], T.B. Sjursen[13], K. Skovpen[107], P. Skubic[111], M. Slater[17], T. Slavicek[127], K. Sliwa[160], J. Sloper[29], T. Sluka[125], V. Smakhtin[170], S.Yu. Smirnov[96], Y. Smirnov[24], L.N. Smirnova[97], O. Smirnova[79], B.C. Smith[57], D. Smith[143], K.M. Smith[53], M. Smizanska[71], K. Smolek[127], A.A. Snesarev[94], S.W. Snow[82], J. Snow[111], J. Snuverink[105], S. Snyder[24], M. Soares[124a], R. Sobie[168,e], J. Sodomka[127], A. Soffer[152], C.A. Solans[166], M. Solar[127], J. Solc[127], E. Solfaroli Camillocci[132a,132b], A.A. Solodkov[128], O.V. Solovyanov[128], R. Soluk[2], J. Sondericker[24],

V. Sopko[127], B. Sopko[127], M. Sosebee[7], A. Soukharev[107], S. Spagnolo[72a,72b], F. Spanò[34], E. Spencer[137], R. Spighi[19a], G. Spigo[29], F. Spila[132a,132b], R. Spiwoks[29], M. Spousta[126], T. Spreitzer[142], B. Spurlock[7], R.D.St. Denis[53], T. Stahl[141], J. Stahlman[120], R. Stamen[58a], S.N. Stancu[162], E. Stanecka[29], R.W. Stanek[5], C. Stanescu[134a], S. Stapnes[117], E.A. Starchenko[128], J. Stark[55], P. Staroba[125], P. Starovoitov[91], J. Stastny[125], P. Stavina[144a], G. Stavropoulos[14], G. Steele[53], P. Steinbach[43], P. Steinberg[24], I. Stekl[127], B. Stelzer[142], H.J. Stelzer[41], O. Stelzer-Chilton[158a], H. Stenzel[52], K. Stevenson[75], G.A. Stewart[53], M.C. Stockton[29], K. Stoerig[48], G. Stoicea[25a], S. Stonjek[99], P. Strachota[126], A.R. Stradling[7], A. Straessner[43], J. Strandberg[87], S. Strandberg[14], A. Strandlie[117], M. Strauss[111], P. Strizenec[144b], R. Ströhmer[172], D.M. Strom[114], R. Stroynowski[39], J. Strube[129], B. Stugu[13], D.A. Soh[150,v], D. Su[143], Y. Sugaya[116], T. Sugimoto[101], C. Suhr[106a], M. Suk[126], V.V. Sulin[94], S. Sultansoy[3d], T. Sumida[29], X.H. Sun[32d], J.E. Sundermann[48], K. Suruliz[163a,163b], S. Sushkov[11], G. Susinno[36a,36b], M.R. Sutton[139], T. Suzuki[154], Y. Suzuki[66], I. Sykora[144a], T. Sykora[126], T. Szymocha[38], J. Sánchez[166], D. Ta[20], K. Tackmann[29], A. Taffard[162], R. Tafirout[158a], A. Taga[117], Y. Takahashi[101], H. Takai[24], R. Takashima[69], H. Takeda[67], T. Takeshita[140], M. Talby[83], A. Talyshev[107], M.C. Tamsett[76], J. Tanaka[154], R. Tanaka[115], S. Tanaka[131], S. Tanaka[66], S. Tapprogge[81], D. Tardif[157], S. Tarem[151], F. Tarrade[24], G.F. Tartarelli[89a], P. Tas[126], M. Tasevsky[125], E. Tassi[36a,36b], M. Tatarkhanov[14], C. Taylor[77], F.E. Taylor[92], G.N. Taylor[86], R.P. Taylor[168], W. Taylor[158b], P. Teixeira-Dias[76], H. Ten Kate[29], P.K. Teng[150], Y.D. Tennenbaum-Katan[151], S. Terada[66], K. Terashi[154], J. Terron[80], M. Terwort[41k], M. Testa[47], R.J. Teuscher[157,e], M. Thioye[174], S. Thoma[48], J.P. Thomas[17], E.N. Thompson[84], P.D. Thompson[17], P.D. Thompson[157], R.J. Thompson[82], A.S. Thompson[53], E. Thomson[120], R.P. Thun[87], T. Tic[125], V.O. Tikhomirov[94], Y.A. Tikhonov[107], P. Tipton[174], F.J. Tique Aires Viegas[29], S. Tisserant[83], B. Toczek[37], T. Todorov[4], S. Todorova-Nova[160], B. Toggerson[162], J. Tojo[66], S. Tokár[144a], K. Tokushuku[66], K. Tollefson[88], L. Tomasek[125], M. Tomasek[125], M. Tomoto[101], L. Tompkins[14], K. Toms[103], A. Tonoyan[13], C. Topfel[16], N.D. Topilin[65], E. Torrence[114], E. Torró Pastor[166], J. Toth[83,t], F. Touchard[83], D.R. Tovey[139], T. Trefzger[172], L. Tremblet[29], A. Tricoli[29], I.M. Trigger[158a], S. Trincaz-Duvoid[78], T.N. Trinh[78], M.F. Tripiana[70], N. Triplett[64], W. Trischuk[157], A. Trivedi[24,s], B. Trocmé[55], C. Troncon[89a], A. Trzupek[38], C. Tsarouchas[9], J.C.-L. Tseng[118], M. Tsiakiris[105], P.V. Tsiareshka[90], D. Tsionou[139], G. Tsipolitis[9], V. Tsiskaridze[51], E.G. Tskhadadze[51], I.I. Tsukerman[95], V. Tsulaia[123], J.-W. Tsung[20], S. Tsuno[66], D. Tsybychev[147], J.M. Tuggle[30], D. Turecek[127], I. Turk Cakir[3e], E. Turlay[105], P.M. Tuts[34], M.S. Twomey[138], M. Tylmad[145a,145b], M. Tyndel[129], K. Uchida[116], I. Ueda[154], M. Ugland[13], M. Uhlenbrock[20], M. Uhrmacher[54], F. Ukegawa[159], G. Unal[29], A. Undrus[24], G. Unel[162], Y. Unno[66], D. Urbaniec[34], E. Urkovsky[152], P. Urquijo[49,w], P. Urrejola[31a], G. Usai[7], M. Uslenghi[119a,119b], L. Vacavant[83], V. Vacek[127], B. Vachon[85], S. Vahsen[14], P. Valente[132a], S. Valentinetti[19a,19b], S. Valkar[126], E. Valladolid Gallego[166], S. Vallecorsa[151], J.A. Valls Ferrer[166], R. Van Berg[120], H. van der Graaf[105], E. van der Kraaij[105], E. van der Poel[105], D. van der Ster[29], N. van Eldik[84], P. van Gemmeren[5], Z. van Kesteren[105], I. van Vulpen[105], W. Vandelli[29], A. Vaniachine[5], P. Vankov[73], F. Vannucci[78], R. Vari[132a], E.W. Varnes[6], D. Varouchas[14], A. Vartapetian[7], K.E. Varvell[149], L. Vasilyeva[94], V.I. Vassilakopoulos[56], F. Vazeille[33], C. Vellidis[8], F. Veloso[124a], S. Veneziano[132a], A. Ventura[72a,72b], D. Ventura[138], M. Venturi[48], N. Venturi[16], V. Vercesi[119a], M. Verducci[172], W. Verkerke[105], J.C. Vermeulen[105], M.C. Vetterli[142,b], I. Vichou[164], T. Vickey[118], G.H.A. Viehhauser[118], M. Villa[19a,19b], E.G. Villani[129], M. Villaplana Perez[166], E. Vilucchi[47], M.G. Vincter[28], E. Vinek[29], V.B. Vinogradov[65], S. Viret[33], J. Virzi[14], A. Vitale[19a,19b], O. Vitells[170], I. Vivarelli[48], F. Vives Vaque[11], S. Vlachos[9], M. Vlasak[127], N. Vlasov[20], A. Vogel[20], P. Vokac[127], M. Volpi[11], H. von der Schmitt[99], J. von Loeben[99], H. von Radziewski[48], E. von Toerne[20], V. Vorobel[126], V. Vorwerk[11], M. Vos[166], R. Voss[29], T.T. Voss[173], J.H. Vossebeld[73], N. Vranjes[12a], M. Vranjes Milosavljevic[12a], V. Vrba[125], M. Vreeswijk[105], T. Vu Anh[81], D. Vudragovic[12a], R. Vuillermet[29], I. Vukotic[115], P. Wagner[120], J. Walbersloh[42], J. Walder[71], R. Walker[98], W. Walkowiak[141], R. Wall[174], C. Wang[44], H. Wang[171], J. Wang[55], S.M. Wang[150], A. Warburton[85], C.P. Ward[27], M. Warsinsky[48], R. Wastie[118], P.M. Watkins[17], A.T. Watson[17], M.F. Watson[17], G. Watts[138], S. Watts[82], A.T. Waugh[149], B.M. Waugh[77], M.D. Weber[16], M. Weber[129], M.S. Weber[16], P. Weber[58a], A.R. Weidberg[118], J. Weingarten[54], C. Weiser[48], H. Wellenstein[22], P.S. Wells[29], M. Wen[47], T. Wenaus[24], S. Wendler[123], T. Wengler[82], S. Wenig[29], N. Wermes[20], M. Werner[48], P. Werner[29], M. Werth[162], U. Werthenbach[141], M. Wessels[58a], K. Whalen[28], A. White[7], M.J. White[27], S. White[24], S.R. Whitehead[118], D. Whiteson[162], D. Whittington[61], F. Wicek[115], D. Wicke[81], F.J. Wickens[129], W. Wiedenmann[171], M. Wielers[129], P. Wienemann[20], C. Wiglesworth[73], L.A.M. Wiik[48], A. Wildauer[166], M.A. Wildt[41k], H.G. Wilkens[29], E. Williams[34], H.H. Williams[120], S. Willocq[84], J.A. Wilson[17], M.G. Wilson[143], A. Wilson[87], I. Wingerter-Seez[4], F. Winklmeier[29], M. Wittgen[143], M.W. Wolter[38], H. Wolters[124a], B.K. Wosiek[38], J. Wotschack[29], M.J. Woudstra[84], K. Wraight[53], C. Wright[53], D. Wright[143], B. Wrona[73], S.L. Wu[171], X. Wu[49], E. Wulf[34], B.M. Wynne[45], L. Xaplanteris[9], S. Xella[35], S. Xie[48], D. Xu[139], N. Xu[171], M. Yamada[159],

A. Yamamoto[66], K. Yamamoto[64], S. Yamamoto[154], T. Yamamura[154], J. Yamaoka[44], T. Yamazaki[154], Y. Yamazaki[67], Z. Yan[21], H. Yang[87], U.K. Yang[82], Z. Yang[145a,145b], W.-M. Yao[14], Y. Yao[14], Y. Yasu[66], J. Ye[39], S. Ye[24], M. Yilmaz[3c], R. Yoosoofmiya[123], K. Yorita[169], R. Yoshida[5], C. Young[143], S.P. Youssef[21], D. Yu[24], J. Yu[7], L. Yuan[78], A. Yurkewicz[147], R. Zaidan[63], A.M. Zaitsev[128], Z. Zajacova[29], V. Zambrano[47], L. Zanello[132a,132b], A. Zaytsev[107], C. Zeitnitz[173], M. Zeller[174], A. Zemla[38], C. Zendler[20], O. Zenin[128], T. Zenis[144a], Z. Zenonos[122a,122b], S. Zenz[14], D. Zerwas[115], G. Zevi della Porta[57], Z. Zhan[32d], H. Zhang[83], J. Zhang[5], Q. Zhang[5], X. Zhang[32d], L. Zhao[108], T. Zhao[138], Z. Zhao[32b], A. Zhemchugov[65], J. Zhong[150x], B. Zhou[87], N. Zhou[34], Y. Zhou[150], C.G. Zhu[32d], H. Zhu[41], Y. Zhu[171], X. Zhuang[98], V. Zhuravlov[99], R. Zimmermann[20], S. Zimmermann[20], S. Zimmermann[48], M. Ziolkowski[141], L. Živković[34], G. Zobernig[171], A. Zoccoli[19a,19b], M. zur Nedden[15], V. Zutshi[106a]

★CERN, 1211 Geneva 23, Switzerland
[1]University at Albany, 1400 Washington Ave, Albany, NY 12222, USA
[2]University of Alberta, Department of Physics, Centre for Particle Physics, Edmonton, AB T6G 2G7, Canada
[3]Ankara University[(a)], Faculty of Sciences, Department of Physics, 061000 Tandogan, Ankara; Dumlupinar University[(b)], Faculty of Arts and Sciences, Department of Physics, Kutahya; Gazi University[(c)], Faculty of Arts and Sciences, Department of Physics, 06500 Teknikokullar, Ankara; TOBB University of Economics and Technology[(d)], Faculty of Arts and Sciences, Division of Physics, 06560 Sogutozu, Ankara; Turkish Atomic Energy Authority[(e)], 06530 Lodumlu, Ankara, Turkey
[4]LAPP, Université de Savoie, CNRS/IN2P3, Annecy-le-Vieux, France
[5]Argonne National Laboratory, High Energy Physics Division, 9700 S. Cass Avenue, Argonne, IL 60439, USA
[6]University of Arizona, Department of Physics, Tucson, AZ 85721, USA
[7]The University of Texas at Arlington, Department of Physics, Box 19059, Arlington, TX 76019, USA
[8]University of Athens, Nuclear & Particle Physics, Department of Physics, Panepistimiopouli, Zografou, 15771 Athens, Greece
[9]National Technical University of Athens, Physics Department, 9-Iroon Polytechniou, 15780 Zografou, Greece
[10]Institute of Physics, Azerbaijan Academy of Sciences, H. Javid Avenue 33, 143 Baku, Azerbaijan
[11]Institut de Física d'Altes Energies, IFAE, Edifici Cn, Universitat Autònoma de Barcelona, 08193 Bellaterra (Barcelona), Spain
[12]University of Belgrade[(a)], Institute of Physics, P.O. Box 57, 11001 Belgrade; Vinca Institute of Nuclear Sciences[(b)], Mihajla Petrovica Alasa 12-14, 11001 Belgrade, Serbia
[13]University of Bergen, Department for Physics and Technology, Allegaten 55, 5007 Bergen, Norway
[14]Lawrence Berkeley National Laboratory and University of California, Physics Division, MS50B-6227, 1 Cyclotron Road, Berkeley, CA 94720, USA
[15]Humboldt University, Institute of Physics, Berlin, Newtonstr. 15, 12489 Berlin, Germany
[16]University of Bern, Albert Einstein Center for Fundamental Physics, Laboratory for High Energy Physics, Sidlerstrasse 5, 3012 Bern, Switzerland
[17]University of Birmingham, School of Physics and Astronomy, Edgbaston, Birmingham B15 2TT, UK
[18]Bogazici University[(a)], Faculty of Sciences, Department of Physics, 80815 Bebek-Istanbul; Dogus University[(b)], Faculty of Arts and Sciences, Department of Physics, 34722 Kadikoy, Istanbul; [(c)]Gaziantep University, Faculty of Engineering, Department of Physics Engineering, 27310 Sehitkamil, Gaziantep; Istanbul Technical University[(d)], Faculty of Arts and Sciences, Department of Physics, 34469 Maslak, Istanbul, Turkey
[19]INFN Sezione di Bologna[(a)]; Università di Bologna, Dipartimento di Fisica[(b)], viale C. Berti Pichat, 6/2, 40127 Bologna, Italy
[20]University of Bonn, Physikalisches Institut, Nussallee 12, 53115 Bonn, Germany
[21]Boston University, Department of Physics, 590 Commonwealth Avenue, Boston, MA 02215, USA
[22]Brandeis University, Department of Physics, MS057, 415 South Street, Waltham, MA 02454, USA
[23]Universidade Federal do Rio De Janeiro, COPPE/EE/IF [(a)], Caixa Postal 68528, Ilha do Fundao, 21945-970 Rio de Janeiro; [(b)]Universidade de Sao Paulo, Instituto de Fisica, R.do Matao Trav. R.187, Sao Paulo, SP, 05508-900, Brazil
[24]Brookhaven National Laboratory, Physics Department, Bldg. 510A, Upton, NY 11973, USA
[25]National Institute of Physics and Nuclear Engineering[(a)], Str. Atomistilor 407, P.O. Box MG-6, 077125 Bucharest-Magurele; University Politehnica Bucharest[(b)], Rectorat AN 001, 313 Splaiul Independentei, sector 6, 060042 Bucuresti; West University[(c)] in Timisoara, Bd. Vasile Parvan 4, Timisoara, Romania
[26]Universidad de Buenos Aires, FCEyN, Dto. Fisica, Pab I, C. Universitaria, 1428 Buenos Aires, Argentina
[27]University of Cambridge, Cavendish Laboratory, J J Thomson Avenue, Cambridge CB3 0HE, UK
[28]Carleton University, Department of Physics, 1125 Colonel By Drive, Ottawa, ON K1S 5B6, Canada
[29]CERN, 1211 Geneva 23, Switzerland
[30]University of Chicago, Enrico Fermi Institute, 5640 S. Ellis Avenue, Chicago, IL 60637, USA
[31]Pontificia Universidad Católica de Chile, Facultad de Fisica, Departamento de Fisica[(a)], Avda. Vicuna Mackenna 4860, San Joaquin, Santiago; Universidad Técnica Federico Santa María, Departamento de Física[(b)], Avda. Espãna 1680, Casilla 110-V, Valparaíso, Chile
[32]Institute of High Energy Physics, Chinese Academy of Sciences[(a)], P.O. Box 918, 19 Yuquan Road, Shijing Shan District, Beijing 100049; University of Science & Technology of China (USTC), Department of Modern Physics[(b)], Hefei, Anhui 230026; Nanjing University, Department of Physics[(c)], 22 Hankou Road, Nanjing, 210093; Shandong University, High Energy Physics Group[(d)], Jinan, Shandong 250100, China
[33]Laboratoire de Physique Corpusculaire, Clermont Université, Université Blaise Pascal, CNRS/IN2P3, 63177 Aubiere Cedex, France
[34]Columbia University, Nevis Laboratory, 136 So. Broadway, Irvington, NY 10533, USA
[35]University of Copenhagen, Niels Bohr Institute, Blegdamsvej 17, 2100 Kobenhavn 0, Denmark
[36]INFN Gruppo Collegato di Cosenza[(a)]; Università della Calabria, Dipartimento di Fisica[(b)], 87036 Arcavacata di Rende, Italy
[37]Faculty of Physics and Applied Computer Science of the AGH-University of Science and Technology, (FPACS, AGH-UST), al. Mickiewicza 30, 30059 Cracow, Poland

[38]The Henryk Niewodniczanski Institute of Nuclear Physics, Polish Academy of Sciences, ul. Radzikowskiego 152, 31342 Krakow, Poland
[39]Southern Methodist University, Physics Department, 106 Fondren Science Building, Dallas, TX 75275-0175, USA
[40]University of Texas at Dallas, 800 West Campbell Road, Richardson, TX 75080-3021, USA
[41]DESY, Notkestr. 85, 22603 Hamburg and Platanenallee 6, 15738 Zeuthen, Germany
[42]TU Dortmund, Experimentelle Physik IV, 44221 Dortmund, Germany
[43]Technical University Dresden, Institut für Kern- und Teilchenphysik, Zellescher Weg 19, 01069 Dresden, Germany
[44]Duke University, Department of Physics, Durham, NC 27708, USA
[45]University of Edinburgh, School of Physics & Astronomy, James Clerk Maxwell Building, The Kings Buildings, Mayfield Road, Edinburgh EH9 3JZ, UK
[46]Fachhochschule Wiener Neustadt; Johannes Gutenbergstrasse 3, 2700 Wiener Neustadt, Austria
[47]INFN Laboratori Nazionali di Frascati, via Enrico Fermi 40, 00044 Frascati, Italy
[48]Albert-Ludwigs-Universität, Fakultät für Mathematik und Physik, Hermann-Herder Str. 3, 79104 Freiburg i.Br., Germany
[49]Université de Genève, Section de Physique, 24 rue Ernest Ansermet, 1211 Geneve 4, Switzerland
[50]INFN Sezione di Genova[(a)]; Università di Genova, Dipartimento di Fisica[(b)], via Dodecaneso 33, 16146 Genova, Italy
[51]Institute of Physics of the Georgian Academy of Sciences, 6 Tamarashvili St., 380077 Tbilisi; Tbilisi State University, HEP Institute, University St. 9, 380086 Tbilisi, Georgia
[52]Justus-Liebig-Universität Giessen, II Physikalisches Institut, Heinrich-Buff Ring 16, 35392 Giessen, Germany
[53]University of Glasgow, Department of Physics and Astronomy, Glasgow G12 8QQ, UK
[54]Georg-August-Universität, II. Physikalisches Institut, Friedrich-Hund Platz 1, 37077 Göttingen, Germany
[55]Laboratoire de Physique Subatomique et de Cosmologie, CNRS/IN2P3, Université Joseph Fourier, INPG, 53 avenue des Martyrs, 38026 Grenoble Cedex, France
[56]Hampton University, Department of Physics, Hampton, VA 23668, USA
[57]Harvard University, Laboratory for Particle Physics and Cosmology, 18 Hammond Street, Cambridge, MA 02138, USA
[58]Ruprecht-Karls-Universität Heidelberg: Kirchhoff-Institut für Physik[(a)], Im Neuenheimer Feld 227, 69120 Heidelberg; Physikalisches Institut[(b)], Philosophenweg 12, 69120 Heidelberg; ZITI Ruprecht-Karls-University Heidelberg[(c)], Lehrstuhl für Informatik V, B6, 23-29, 68131 Mannheim, Germany
[59]Hiroshima University, Faculty of Science, 1-3-1 Kagamiyama, Higashihiroshima-shi, Hiroshima 739-8526, Japan
[60]Hiroshima Institute of Technology, Faculty of Applied Information Science, 2-1-1 Miyake Saeki-ku, Hiroshima-shi, Hiroshima 731-5193, Japan
[61]Indiana University, Department of Physics, Swain Hall West 117, Bloomington, IN 47405-7105, USA
[62]Institut für Astro- und Teilchenphysik, Technikerstrasse 25, 6020 Innsbruck, Austria
[63]University of Iowa, 203 Van Allen Hall, Iowa City, IA 52242-1479, USA
[64]Iowa State University, Department of Physics and Astronomy, Ames High Energy Physics Group, Ames, IA 50011-3160, USA
[65]Joint Institute for Nuclear Research, JINR, Dubna 141 980, Moscow Region, Russia
[66]KEK, High Energy Accelerator Research Organization, 1-1 Oho, Tsukuba-shi, Ibaraki-ken 305-0801, Japan
[67]Kobe University, Graduate School of Science, 1-1 Rokkodai-cho, Nada-ku, Kobe 657-8501, Japan
[68]Kyoto University, Faculty of Science, Oiwake-cho, Kitashirakawa, Sakyou-ku, Kyoto-shi, Kyoto 606-8502, Japan
[69]Kyoto University of Education, 1 Fukakusa, Fujimori, fushimi-ku, Kyoto-shi, Kyoto 612-8522, Japan
[70]Universidad Nacional de La Plata, FCE, Departamento de Física, IFLP (CONICET-UNLP), C.C. 67, 1900 La Plata, Argentina
[71]Lancaster University, Physics Department, Lancaster LA1 4YB, UK
[72]INFN Sezione di Lecce[(a)]; Università del Salento, Dipartimento di Fisica[(b)]Via Arnesano, 73100 Lecce, Italy
[73]University of Liverpool, Oliver Lodge Laboratory, P.O. Box 147, Oxford Street, Liverpool L69 3BX, UK
[74]Jožef Stefan Institute and University of Ljubljana, Department of Physics, 1000 Ljubljana, Slovenia
[75]Queen Mary University of London, Department of Physics, Mile End Road, London E1 4NS, UK
[76]Royal Holloway, University of London, Department of Physics, Egham Hill, Egham, Surrey TW20 0EX, UK
[77]University College London, Department of Physics and Astronomy, Gower Street, London WC1E 6BT, UK
[78]Laboratoire de Physique Nucléaire et de Hautes Energies, Université Pierre et Marie Curie (Paris 6), Université Denis Diderot (Paris-7), CNRS/IN2P3, Tour 33, 4 place Jussieu, 75252 Paris Cedex 05, France
[79]Lunds universitet, Naturvetenskapliga fakulteten, Fysiska institutionen, Box 118, 221 00 Lund, Sweden
[80]Universidad Autonoma de Madrid, Facultad de Ciencias, Departamento de Fisica Teorica, 28049 Madrid, Spain
[81]Universität Mainz, Institut für Physik, Staudinger Weg 7, 55099 Mainz, Germany
[82]University of Manchester, School of Physics and Astronomy, Manchester M13 9PL, UK
[83]CPPM, Aix-Marseille Université, CNRS/IN2P3, Marseille, France
[84]University of Massachusetts, Department of Physics, 710 North Pleasant Street, Amherst, MA 01003, USA
[85]McGill University, High Energy Physics Group, 3600 University Street, Montreal, Quebec H3A 2T8, Canada
[86]University of Melbourne, School of Physics, Parkville, Victoria 3010, Australia
[87]The University of Michigan, Department of Physics, 2477 Randall Laboratory, 500 East University, Ann Arbor, MI 48109-1120, USA
[88]Michigan State University, Department of Physics and Astronomy, High Energy Physics Group, East Lansing, MI 48824-2320, USA
[89]INFN Sezione di Milano[(a)]; Università di Milano, Dipartimento di Fisica[(b)], via Celoria 16, 20133 Milano, Italy
[90]B.I. Stepanov Institute of Physics, National Academy of Sciences of Belarus, Independence Avenue 68, Minsk 220072, Republic of Belarus
[91]National Scientific & Educational Centre for Particle & High Energy Physics, NC PHEP BSU, M. Bogdanovich St. 153, Minsk 220040, Republic of Belarus
[92]Massachusetts Institute of Technology, Department of Physics, Room 24-516, Cambridge, MA 02139, USA
[93]University of Montreal, Group of Particle Physics, C.P. 6128, Succursale Centre-Ville, Montreal, Quebec H3C 3J7, Canada
[94]P.N. Lebedev Institute of Physics, Academy of Sciences, Leninsky pr. 53, 117 924 Moscow, Russia

[95]Institute for Theoretical and Experimental Physics (ITEP), B. Cheremushkinskaya ul. 25, 117 218 Moscow, Russia
[96]Moscow Engineering & Physics Institute (MEPhI), Kashirskoe Shosse 31, 115409 Moscow, Russia
[97]Lomonosov Moscow State University Skobeltsyn Institute of Nuclear Physics (MSU SINP), 1(2), Leninskie gory, GSP-1, Moscow 119991, Russia
[98]Ludwig-Maximilians-Universität München, Fakultät für Physik, Am Coulombwall 1, 85748 Garching, Germany
[99]Max-Planck-Institut für Physik, (Werner-Heisenberg-Institut), Föhringer Ring 6, 80805 München, Germany
[100]Nagasaki Institute of Applied Science, 536 Aba-machi, Nagasaki 851-0193, Japan
[101]Nagoya University, Graduate School of Science, Furo-Cho, Chikusa-ku, Nagoya 464-8602, Japan
[102]INFN Sezione di Napoli[(a)]; Università di Napoli, Dipartimento di Scienze Fisiche[(b)], Complesso Universitario di Monte Sant'Angelo, via Cinthia, 80126 Napoli, Italy
[103]University of New Mexico, Department of Physics and Astronomy, MSC07 4220, Albuquerque, NM 87131, USA
[104]Radboud University Nijmegen/NIKHEF, Department of Experimental High Energy Physics, Heyendaalseweg 135, 6525 AJ, Nijmegen, Netherlands
[105]Nikhef National Institute for Subatomic Physics, and University of Amsterdam, Science Park 105, 1098 XG Amsterdam, Netherlands
[106(a)]DeKalb, Illinois 60115, USA
[107]Budker Institute of Nuclear Physics (BINP), Novosibirsk 630 090, Russia
[108]New York University, Department of Physics, 4 Washington Place, New York, NY 10003, USA
[109]Ohio State University, 191 West Woodruff Ave, Columbus, OH 43210-1117, USA
[110]Okayama University, Faculty of Science, Tsushimanaka 3-1-1, Okayama 700-8530, Japan
[111]University of Oklahoma, Homer L. Dodge Department of Physics and Astronomy, 440 West Brooks, Room 100, Norman, OK 73019-0225, USA
[112]Oklahoma State University, Department of Physics, 145 Physical Sciences Building, Stillwater, OK 74078-3072, USA
[113]Palacký University, 17.listopadu 50a, 772 07 Olomouc, Czech Republic
[114]University of Oregon, Center for High Energy Physics, Eugene, OR 97403-1274, USA
[115]LAL, Univ. Paris-Sud, IN2P3/CNRS, Orsay, France
[116]Osaka University, Graduate School of Science, Machikaneyama-machi 1-1, Toyonaka, Osaka 560-0043, Japan
[117]University of Oslo, Department of Physics, P.O. Box 1048, Blindern, 0316 Oslo 3, Norway
[118]Oxford University, Department of Physics, Denys Wilkinson Building, Keble Road, Oxford OX1 3RH, UK
[119]INFN Sezione di Pavia[(a)]; Università di Pavia, Dipartimento di Fisica Nucleare e Teorica[(b)], Via Bassi 6, 27100 Pavia, Italy
[120]University of Pennsylvania, Department of Physics, High Energy Physics Group, 209 S. 33rd Street, Philadelphia, PA 19104, USA
[121]Petersburg Nuclear Physics Institute, 188 300 Gatchina, Russia
[122]INFN Sezione di Pisa[(a)]; Università di Pisa, Dipartimento di Fisica E. Fermi[(b)], Largo B. Pontecorvo 3, 56127 Pisa, Italy
[123]University of Pittsburgh, Department of Physics and Astronomy, 3941 O'Hara Street, Pittsburgh, PA 15260, USA
[124]Laboratorio de Instrumentacao e Fisica Experimental de Particulas – LIP[(a)], Avenida Elias Garcia 14-1, 1000-149 Lisboa, Portugal; Universidad de Granada, Departamento de Fisica Teorica y del Cosmos and CAFPE[(b)], 18071 Granada, Spain
[125]Institute of Physics, Academy of Sciences of the Czech Republic, Na Slovance 2, 18221 Praha 8, Czech Republic
[126]Charles University in Prague, Faculty of Mathematics and Physics, Institute of Particle and Nuclear Physics, V Holesovickach 2, 18000 Praha 8, Czech Republic
[127]Czech Technical University in Prague, Zikova 4, 166 35 Praha 6, Czech Republic
[128]State Research Center Institute for High Energy Physics, Pobeda street, 1, Protvino 142281, Moscow Region, Russia
[129]Rutherford Appleton Laboratory, Science and Technology Facilities Council, Harwell Science and Innovation Campus, Didcot OX11 0QX, UK
[130]University of Regina, Physics Department, Canada
[131]Ritsumeikan University, Noji Higashi 1 chome 1-1, Kusatsu, Shiga 525-8577, Japan
[132]INFN Sezione di Roma I[(a)]; Università La Sapienza, Dipartimento di Fisica[(b)], Piazzale A. Moro 2, 00185 Roma, Italy
[133]INFN Sezione di Roma Tor Vergata[(a)]; Università di Roma Tor Vergata, Dipartimento di Fisica[(b)] , via della Ricerca Scientifica, 00133 Roma, Italy
[134]INFN Sezione di Roma Tre[(a)]; Università Roma Tre, Dipartimento di Fisica[(b)], via della Vasca Navale 84, 00146 Roma, Italy
[135]Réseau Universitaire de Physique des Hautes Energies (RUPHE): Université Hassan II, Faculté des Sciences Ain Chock[(a)], B.P. 5366, Casablanca; Centre National de l'Energie des Sciences Techniques Nucleaires (CNESTEN)[(b)], B.P. 1382 R.P. 10001 Rabat 10001; Université Mohamed Premier[(c)], LPTPM, Faculté des Sciences, B.P.717. Bd. Mohamed VI, 60000, Oujda; Université Mohammed V, Faculté des Sciences[(d)], 4 Avenue Ibn Battouta, BP 1014 RP, 10000 Rabat, Morocco
[136]CEA, DSM/IRFU, Centre d'Etudes de Saclay, 91191 Gif-sur-Yvette, France
[137]University of California Santa Cruz, Santa Cruz Institute for Particle Physics (SCIPP), Santa Cruz, CA 95064, USA
[138]University of Washington, Seattle, Department of Physics, Box 351560, Seattle, WA 98195-1560, USA
[139]University of Sheffield, Department of Physics & Astronomy, Hounsfield Road, Sheffield S3 7RH, UK
[140]Shinshu University, Department of Physics, Faculty of Science, 3-1-1 Asahi, Matsumoto-shi, Nagano 390-8621, Japan
[141]Universität Siegen, Fachbereich Physik, 57068 Siegen, Germany
[142]Simon Fraser University, Department of Physics, 8888 University Drive, Burnaby, BC V5A 1S6, Canada
[143]SLAC National Accelerator Laboratory, Stanford, California 94309, USA
[144]Comenius University, Faculty of Mathematics, Physics & Informatics[(a)], Mlynska dolina F2, 84248 Bratislava; Institute of Experimental Physics of the Slovak Academy of Sciences, Dept. of Subnuclear Physics[(b)], Watsonova 47, 04353 Kosice, Slovak Republic
[145]Stockholm University, Department of Physics[(a)]; The Oskar Klein Centre[(b)], AlbaNova, 106 91 Stockholm, Sweden
[146]Royal Institute of Technology (KTH), Physics Department, 106 91 Stockholm, Sweden
[147]Stony Brook University, Department of Physics and Astronomy, Nicolls Road, Stony Brook, NY 11794-3800, USA

[148]University of Sussex, Department of Physics and Astronomy Pevensey 2 Building, Falmer, Brighton BN1 9QH, UK
[149]University of Sydney, School of Physics, Sydney NSW 2006, Australia
[150]Insitute of Physics, Academia Sinica, Taipei 11529, Taiwan
[151]Technion, Israel Inst. of Technology, Department of Physics, Technion City, Haifa 32000, Israel
[152]Tel Aviv University, Raymond and Beverly Sackler School of Physics and Astronomy, Ramat Aviv, Tel Aviv 69978, Israel
[153]Aristotle University of Thessaloniki, Faculty of Science, Department of Physics, Division of Nuclear & Particle Physics, University Campus, 54124 Thessaloniki, Greece
[154]The University of Tokyo, International Center for Elementary Particle Physics and Department of Physics, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
[155]Tokyo Metropolitan University, Graduate School of Science and Technology, 1-1 Minami-Osawa, Hachioji, Tokyo 192-0397, Japan
[156]Tokyo Institute of Technology, 2-12-1-H-34 O-Okayama, Meguro, Tokyo 152-8551, Japan
[157]University of Toronto, Department of Physics, 60 Saint George Street, Toronto, Ontario M5S 1A7, Canada
[158]TRIUMF[(a)], 4004 Wesbrook Mall, Vancouver, BC V6T 2A3; [(b)]York University, Department of Physics and Astronomy, 4700 Keele St., Toronto, Ontario M3J 1P3, Canada
[159]University of Tsukuba, Institute of Pure and Applied Sciences, 1-1-1 Tennoudai, Tsukuba-shi, Ibaraki 305-8571, Japan
[160]Tufts University, Science & Technology Center, 4 Colby Street, Medford, MA 02155, USA
[161]Universidad Antonio Narino, Centro de Investigaciones, Cra 3 Este No. 47A-15, Bogota, Colombia
[162]University of California, Irvine, Department of Physics & Astronomy, CA 92697-4575, USA
[163]INFN Gruppo Collegato di Udine[(a)]; ICTP[(b)], Strada Costiera 11, 34014, Trieste; Università di Udine, Dipartimento di Fisica[(c)], via delle Scienze 208, 33100 Udine, Italy
[164]University of Illinois, Department of Physics, 1110 West Green Street, Urbana, IL 61801, USA
[165]University of Uppsala, Department of Physics and Astronomy, P.O. Box 516, 751 20 Uppsala, Sweden
[166]Instituto de Física Corpuscular (IFIC) Centro Mixto UVEG-CSIC, Apdo. 22085 46071 Valencia, Dept. Física At. Mol. y Nuclear; Univ. of Valencia, and Instituto de Microelectrónica de Barcelona (IMB-CNM-CSIC) 08193 Bellaterra Barcelona, Spain
[167]University of British Columbia, Department of Physics, 6224 Agricultural Road, Vancouver, BC V6T 1Z1, Canada
[168]University of Victoria, Department of Physics and Astronomy, P.O. Box 3055, Victoria, BC V8W 3P6, Canada
[169]Waseda University, WISE, 3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555, Japan
[170]The Weizmann Institute of Science, Department of Particle Physics, P.O. Box 26, 76100 Rehovot, Israel
[171]University of Wisconsin, Department of Physics, 1150 University Avenue, Madison WI 53706, USA
[172]Julius-Maximilians-University of Würzburg, Physikalisches Institute, Am Hubland, 97074 Würzburg, Germany
[173]Bergische Universität, Fachbereich C, Physik, Postfach 100127, Gauss-Strasse 20, 42097 Wuppertal, Germany
[174]Yale University, Department of Physics, PO Box 208121, New Haven CT, 06520-8121, USA
[175]Yerevan Physics Institute, Alikhanian Brothers Street 2, 375036 Yerevan, Armenia
[176]ATLAS-Canada Tier-1 Data Centre, TRIUMF, 4004 Wesbrook Mall, Vancouver, BC, V6T 2A3, Canada
[177]GridKA Tier-1 FZK, Forschungszentrum Karlsruhe GmbH, Steinbuch Centre for Computing (SCC), Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany
[178]Port d'Informacio Cientifica (PIC), Universitat Autonoma de Barcelona (UAB), Edifici D, 08193 Bellaterra, Spain
[179]Centre de Calcul CNRS/IN2P3, Domaine scientifique de la Doua, 27 bd du 11 Novembre 1918, 69622 Villeurbanne Cedex, France
[180]INFN-CNAF, Viale Berti Pichat 6/2, 40127 Bologna, Italy
[181]Nordic Data Grid Facility, NORDUnet A/S, Kastruplundgade 22, 1, 2770 Kastrup, Denmark
[182]SARA Reken- en Netwerkdiensten, Science Park 121, 1098 XG Amsterdam, Netherlands
[183]Academia Sinica Grid Computing, Institute of Physics, Academia Sinica, No.128, Sec. 2, Academia Rd., Nankang, Taipei, Taiwan 11529, Taiwan
[184]UK-T1-RAL Tier-1, Rutherford Appleton Laboratory, Science and Technology Facilities Council, Harwell Science and Innovation Campus, Didcot OX11 0QX, UK
[185]RHIC and ATLAS Computing Facility, Physics Department, Building 510, Brookhaven National Laboratory, Upton, NY 11973, USA
[a]Also at CPPM, Marseille, France.
[b]Also at TRIUMF, 4004 Wesbrook Mall, Vancouver, BC V6T 2A3, Canada.
[c]Also at Faculty of Physics and Applied Computer Science of the AGH-University of Science and Technology, (FPACS, AGH-UST), al. Mickiewicza 30, 30059 Cracow, Poland.
[d]Also at Università di Napoli Parthenope, via A. Acton 38, 80133 Napoli, Italy.
[e]Also at Institute of Particle Physics (IPP), Canada.
[f]Louisiana Tech University, 305 Wisteria Street, P.O. Box 3178, Ruston, LA 71272, USA.
[g]At Department of Physics, California State University, Fresno, 2345 E. San Ramon Avenue, Fresno, CA 93740-8031, USA.
[h]Currently at Istituto Universitario di Studi Superiori IUSS, V.le Lungo Ticino Sforza 56, 27100 Pavia, Italy.
[i]Also at California Institute of Technology, Physics Department, Pasadena, CA 91125, USA.
[j]Also at University of Montreal, Canada.
[k]Also at Institut für Experimentalphysik, Universität Hamburg, Luruper Chaussee 149, 22761 Hamburg, Germany.
[l]Also at Petersburg Nuclear Physics Institute, 188 300 Gatchina, Russia.
[m]Also at School of Physics and Engineering, Sun Yat-sen University, China.
[n]Also at School of Physics, Shandong University, Jinan, China.
[o]Also at Rutherford Appleton Laboratory, Science and Technology Facilities Council, Harwell Science and Innovation Campus, Didcot OX11, UK.
[p]Also at School of Physics, Shandong University, Jinan.
[q]Also at Rutherford Appleton Laboratory, Science and Technology Facilities Council, Harwell Science and Innovation Campus, Didcot OX11 0QX, UK.

[r] Now at KEK.
[s] University of South Carolina, Dept. of Physics and Astronomy, 700 S. Main St, Columbia, SC 29208, USA.
[t] Also at KFKI Research Institute for Particle and Nuclear Physics, Budapest, Hungary.
[u] Also at Institute of Physics, Jagiellonian University, Cracow, Poland.
[v] Also at School of Physics and Engineering, Sun Yat-sen University, Taiwan.
[w] Transfer to LHCb 31.01.2010.
[x] Also at Department of Physics, Nanjing University, China.
[*] Deceased.

Received: 20 May 2010 / Revised: 27 July 2010 / Published online: 25 September 2010
© CERN for the benefit of the ATLAS collaboration 2010. This article is published with open access at Springerlink.com

Abstract The simulation software for the ATLAS Experiment at the Large Hadron Collider is being used for large-scale production of events on the LHC Computing Grid. This simulation requires many components, from the generators that simulate particle collisions, through packages simulating the response of the various detectors and triggers. All of these components come together under the ATLAS simulation infrastructure. In this paper, that infrastructure is discussed, including that supporting the detector description, interfacing the event generation, and combining the GEANT4 simulation of the response of the individual detectors. Also described are the tools allowing the software validation, performance testing, and the validation of the simulated output against known physics processes.

1 Introduction

ATLAS [1], one of the general-purpose detectors at the Large Hadron Collider [2], began operation in 2008. The detector will collect data from proton–proton collisions with center-of-mass energies up to 14 TeV, as well as 5.5 TeV per nucleon pair in heavy ion (Pb–Pb) collisions. During proton–proton collisions at the design luminosity of 10^{34} cm^{-2} s^{-1}, beam bunches will cross every 25 ns (40 MHz) and provide on average 23 collisions per bunch crossing. ATLAS has been designed to record up to 200 bunch crossings per second, keeping only the most interesting interactions for physics analyses, including searches for new physics.

In order to study the detector response for a wide range of physics processes and scenarios, a detailed simulation has been implemented that carries events from the event generation through to output in a format which is identical to that of the true detector. The simulation program is integrated into the ATLAS software framework, Athena [3], and uses the GEANT4 simulation toolkit [4, 5]. The core software and large-scale production infrastructures are discussed further in Sect. 2.

The simulation software chain is generally divided into three steps, though they may be combined into a single job: generation of the event and immediate decays (see Sect. 3), simulation of the detector and physics interactions (see Sect. 5), and digitization of the energy deposited in the sensitive regions of the detector into voltages and currents for comparison to the readout of the ATLAS detector (see Sect. 6). The output of the simulation chain can be presented in either an object-based format or in a format identical to the output of the ATLAS data acquisition system (DAQ). Thus, both the simulated and real data from the detector can then be run through the same ATLAS trigger and reconstruction packages.

The ATLAS detector geometry used for simulation, digitization, and reconstruction is built from databases containing the information describing the physical construction and conditions data. The latter contains all the information needed to emulate a single data-taking run of the real detector (e.g. detector misalignments or temperatures). The same geometry and simulation infrastructure is able to reproduce the test stands and installation configurations of the ATLAS detector. The detector description is discussed in Sect. 4.

Large computing resources are required to accurately model the complex detector geometry and physics descriptions in the standard ATLAS detector simulation. This has led to the development of several varieties of fast simulation. Each is best suited to a particular use-case, and they are described in Sect. 7. Validation of the software, testing of the software performance, and validation of the physics performance and output of each piece of the simulation software chain is discussed in Sect. 8.

This paper reviews the status of the software and geometry used for large-scale production in 2008.

2 ATLAS offline software overview

The ATLAS software framework, Athena [3], uses PYTHON as an object-oriented scripting and interpreter language to configure and load C++ algorithms and objects. Rather than develop an entirely new high-energy physics data processing infrastructure, ATLAS adopted the Gaudi framework [6, 7],

** e-mail: atlas.secretariat@cern.ch

originally developed for LHCb and written in C++. Gaudi was created as a flexible framework to support a variety of applications through base classes and basic functionality. As much as possible, the infrastructure relies on the CLHEP common libraries [8], which include utility classes particularly designed for use in high-energy physics software (e.g. vectors and rotations).

Athena releases are divided into major projects by functionality [9], and all of the ATLAS simulation software (including event generation and digitization) resides in a single project. The dependencies of the "simulation" project are the "core" project, which includes the Athena framework, the "conditions" and "detector description" projects, which include all code necessary for the description of the ATLAS detector, and the "event" project, which includes descriptions of persistent objects. The number of lines of code by software language for the simulation project are summarized in Table 1, as calculated using cloc [10] in Athena release 14.4. Lines of code in the upstream Athena projects, excluding external dependencies like Gaudi and CLHEP, are summarized in Table 2.

Table 1 Numbers of files, lines of code, and lines of comments in the ATLAS simulation project, by programming language for major contributors. External dependencies are not included

Language	Files	Comment	Code
C++	930	24,000	120,000
FORTRAN	270	15,000	42,000
C/C++ Header	1,100	13,000	34,000
Python	430	16,000	27,000
HTML	62	130	15,000
Bourne Shell	390	1,000	7,300
C Shell	380	210	3,800
XML	52	1,200	3,400
Sum	3,600	70,000	250,000

Table 2 Numbers of lines of code in each of the projects upstream of the ATLAS simulation project, versus the programming language. Most projects are dominated by C++ and PYTHON code. The most significant exception is the detector project, which contains 70,000 lines of XML and Java code

Project	C/C++ code	C/C++ headers	PYTHON code	Total code
Core	390,000	43,000	240,000	860,000
Event	200,000	110,000	16,000	350,000
Conditions	280,000	90,000	21,000	620,000
Detector	38,000	6,100	8,400	140,000
Sum	910,000	250,000	280,000	2,000,000

All Athena jobs consist of three distinct steps. First, in the initialization step, services and algorithms are loaded on demand using dynamic library loading. Generally, algorithms include methods to be called once per event, whereas services may be accessed many times during a single event. The configuration and initialization is controlled within a common PYTHON infrastructure which allows introspection, particularly useful in debugging and providing help for the users. Also, by using a scripting language for loading and configuring objects, there is no need to recompile C++ code or a script for each job. Small modifications can be made in the scripts (also called "fragments" or "job options"), or even in the midst of the job, without having to stop and recompile the libraries. This scripting method also lightens the load on the user, since there is, under normal circumstances, no need to compile anything prior to running a job. Each algorithm and service can be configured differently for each step of the simulation software chain, allowing maximal sharing of infrastructure among the distinct steps of the chain. Algorithms can be added to a top list of methods to be run during the event loop.

Second, the event loop begins. All algorithms in the top list are run sequentially on each event. An external generator or algorithm controlling GEANT4 may be added to this list, for example. From these main methods, other services and algorithms can be called. A messaging service, called throughout the jobs, controls log file outputs with different levels of verbosity. The user may configure the total logging verbosity or configure the verbosity individually for a single algorithm, particularly useful for debugging.

During the finalization stage of the job, all algorithms are terminated and all objects are deleted. At this point, algorithms may output any statistics (e.g. memory or CPU usage) they track.

These three steps comprise each Athena job, but the infrastructure allows for the insertion of hooks at various places. Each step of the ATLAS simulation chain takes advantage of this infrastructure to provide maximal flexibility for the user. Only requested modules are loaded as plug-ins, keeping each step as light as possible in memory and as fast as possible during the event loop.

For storing data, ATLAS has adopted a scheme for separating transient from persistent objects. Most general C++ types, immediately prior to storage, are converted to a type that requires less space. Although, for example, energy is accumulated in the calorimeter by summing double-precision floating point numbers, at the end of each event and prior to storage, the total energy is converted into a single-precision floating point number (float). Summing with floats was found to alter the total energy because of truncation. For some types, more complicated storage schemes are implemented that rely on properties of the information to be stored (e.g. where it is possible to sacrifice some accuracy). Metadata, general property information for data collected in a file,

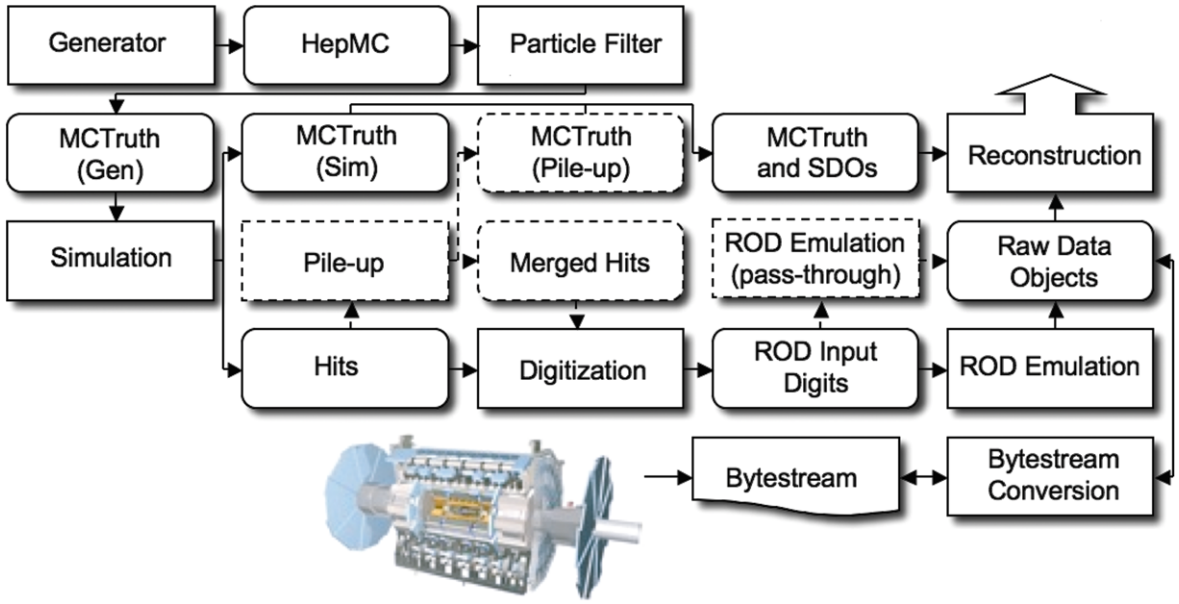


Fig. 1 The flow of the ATLAS simulation software, from event generators (*top left*) through reconstruction (*top right*). Algorithms are placed in *square-cornered boxes* and persistent data objects are placed in *rounded boxes*. The optional pile-up portion of the chain, used only when events are overlaid, is *dashed*. Generators are used to produce data in HepMC format. Monte Carlo truth is saved in addition to energy depositions in the detector (hits). This truth is merged into Simulated Data Objects (SDOs) during the digitization. Also, during the digitization stage, Read Out Driver (ROD) electronics are simulated

are included in the output files for all the stages of the event simulation. The metadata include all configuration information for the job. Athena has also adopted the POOL (Pool Of persistent Objects for LHC) file handling and persistency framework [11–13].

2.1 ATLAS simulation overview

An overview of the ATLAS simulation data flow can be seen in Fig. 1. Algorithms and applications to be run are placed in square-cornered boxes, and persistent data objects are placed in round-cornered boxes. The optional steps required for pile-up or event overlay (see Sect. 6.2) are shown with a dashed outline.

A generator produces events in standard HepMC format [14]. These events can be filtered at generation time so that only events with a certain property (e.g. leptonic decay or missing energy above a certain value) are kept. The generator is responsible for any prompt decays (e.g. Z or W bosons) but stores any "stable" particle expected to propagate through a part of the detector (see Sect. 3). Because it only considers immediate decays, there is no need to consider detector geometry during the generation step, except in controlling what particles are considered stable. During this step, the run number for the simulated data set and event numbers for each event are established. Event numbers are generally ordered in a single job, though events may be omitted because of filtering at each step. Run numbers for simulated data sets derive from the job options used to generate the sample and mimic real run numbers used during data taking.

These generated events are then read into the simulation. A record of all particles produced by the generator is retained in the simulation output file (see Sect. 3.6), but cuts can be applied to select only certain particles to process in the simulation. Each particle is propagated through the full ATLAS detector by GEANT4. The configuration of the detector, including misalignments and distortions, can be set at run time by the user. The energies deposited in the sensitive portions of the detector are recorded as "hits," containing the total energy deposition, position, and time, and are written to a simulation output file, called a hit file.

In both event generation and detector simulation, information called "truth" is recorded for each event. In the generation jobs, the truth is a history of the interactions from the generator, including incoming and outgoing particles. A record is kept for every particle, whether the particle is to be passed through the detector simulation or not. In the simulation jobs, truth tracks and decays for certain particles are stored. This truth contains, for example, the locations of the conversions of photons within the inner detector and the subsequent electron and positron tracks. In the digitization jobs, Simulated Data Objects (SDOs) are created from the truth. These SDOs are maps from the hits in the sensitive regions of the detector to the particles in the simulation truth record that deposited the hits' energy. The truth information

is further processed in the reconstruction jobs and can be used during the analysis of simulated data to quantify the success of the reconstruction software.

The digitization takes hit output from simulated events: hard scattering signal, minimum bias, beam halo, beam gas, and cavern background events. Each type of event can be overlaid at a user-specified rate before the detector signal (e.g. voltage or time) is generated. The overlay (called "pile-up") is done during digitization to save the CPU time required by the simulation. At this stage, detector noise is added to the event. The first level trigger, implemented with hardware on the real detector, is also simulated in a "pass" mode. Here no events are discarded but each trigger hypothesis is evaluated. The digitization first constructs "digits," inputs to the read out drivers (RODs) in the detector electronics. The ROD functionality is then emulated, and the output is a Raw Data Object (RDO) file. The output from the ATLAS detector itself is in "byte stream" format, which can be fairly easily converted to and from RDO file format. The two are similar, and in some subdetectors they are almost interchangeable. Truth information is the major exception. It is stripped in the conversion to bytestream.

The simulation software chain, divided in this way, uses resources more effectively than a single-step event simulation and simplifies software validation. Event generation jobs, typically quick and with small output files, can be run for several thousands of events at a time. By storing the output rather than regenerating it each time, it becomes possible to run identical events through different versions of the simulation software or with different detector configurations. The simulation step is particularly slow, and can take several minutes per event (see Sect. 8.2). Simulation jobs are therefore divided into groups of 50 or fewer events; only a few events may be completed in a single heavy ion simulation job. Digitization jobs are generally configured to run ∼1000 events. This configuration eases file handling by producing a smaller number of RDO files. Each step is partially configured based on the input files. For example, the detector geometry used for a digitization job is selected based on the input hit file.

The ATLAS high level trigger[1] (HLT) [15] and reconstruction [16] run on these RDO files. The reconstruction is identical for the simulation and the data, with the exception that truth information can be treated and is available only in simulated data. During data taking, the HLT is run on bytestream files, however all hypotheses and additional test hypotheses may be evaluated by translating the RDOs into bytestream format.

[1]The ATLAS high level trigger comprises two stages: level 2, and the event filter. Both are software triggers run with the reconstruction, and may be treated as a single unit for the purposes of this discussion.

2.2 Large-scale production system

Because of the significant time consumption of the ATLAS simulation, only minimal jobs can be completed interactively on most computers. It is, therefore, desirable to distribute as much as possible production of the necessary simulated data for ATLAS. Complete software releases are built and distributed to production sites and users every few weeks, providing all Athena software and all external dependencies, including generators and GEANT4. These releases are patched several times with "production caches" before a new clean release is built and distributed. With each release or production cache, a small set of data files are packaged that include database replicas, any necessary external data files, and some sample output files. These sample files can be used to ensure that the locally installed release can be validated by processing events through the entire software chain, from generation through reconstruction.

Large-scale production is then done on the World-wide LHC Computing Grid ("WLCG" or "Grid") [17]. A single task on the Grid (e.g. simulation of 500,000 $t\bar{t}$ events) is separated into many jobs depending on the content and complexity of the task. A job can be completed by a single CPU within the maximum allowed time for a job on the Grid (typically 2–3 days). The output, including log files, of every Grid job is registered with the ATLAS Distributed Data Management system (DDM) [18]. The DDM uses DQ2 [19] for dataset bookkeeping, and allows users to search for datasets on the Grid, analyze them in place, and, if necessary, retrieve them. Separate Grid software controls the distribution of jobs to the various Grid sites. In a typical task, 10 jobs are queued and run as a test sample, and only once they finish successfully are the remainder of the jobs released to the Grid. In the case of a full chain of jobs being run (generation, simulation, and digitization), each subsequent step is automatically held in the queue until the required data are available from the previous step. Frequently, Grid jobs are configured to run two steps (e.g. simulation and digitization together, or digitization and reconstruction together). About one million events per day can be produced using GEANT4 on the Grid.

On the Grid, "job transforms" are run, which may only include well-defined, minor modifications to some standard job configuration after the input events have been specified. A task is given a random number seed, and each job increments the seed in sequence. The modifications to a generation job also may include a configuration file for the selected generator to be run. These configuration files are included with each release and may not be arbitrarily modified by the user during production. The modification to a simulation job may include detector geometry and conditions and specially designed job options fragments that are included with each release. These fragments are typically constructed for

a very specific purpose, for example a non-standard vertex smearing, simulation of cavern background, or propagation and late decay of long-lived exotic particles. Many of these modifications can be chained to provide maximal flexibility to the user, but if two fragments are sufficiently complex such chaining becomes impossible. The modifications to a digitization job may include geometry and conditions versions, calorimeter sampling fraction, trigger configuration, and noise control. These modifications are discussed further in the subsequent sections.

3 Event generation overview

Event generation consists of the production of a set of particles which is passed to either full or fast detector simulation. Event generation runs within the Athena framework, but most of the generators themselves are written and maintained by authors external to ATLAS. The ATLAS-specific implementation, therefore, consists mostly of a set of interface packages. These are designed to be as simple as practicable and wherever possible to be factorized from the external packages. This is essential to allow rapid feedback and bug reporting to the authors of the external packages. Most of the well-understood and thoroughly debugged generators are written in FORTRAN. Their interfaces transfer the event information, mostly contained in FORTRAN common blocks, into an object format that can be used by the ATLAS software. This ensures that any downstream algorithms are shielded from details specific to an individual generator. Events can either be stored as POOL files for later use or passed to simulation in the same Athena job.

Details of the framework and comments specific to each generator are listed below. Large-scale production has been run with PYTHIA [20] (including an ATLAS variant, PythiaB [21, 22], used for production of events with B-hadrons), HERWIG [23–25], Sherpa [26], Hijing [27], Alpgen [28], MC@NLO [29], and AcerMC [30]. Tauola [31] and Photos [32] are routinely used to handle tau decays and photon emission. EvtGen [33] is used for B-decays in cases where the physics is sensitive to details of the B hadron decays.[2] ISAJET [34] is used for generating supersymmetric particles in conjunction with HERWIG. The newer C++ generators PYTHIA 8 [35] and HERWIG++ [36] are being tested. Both produce events in the HepMC format, so no translation is needed. They can be passed directly to simulation. As these new generators evolve and undergo extensive testing and validation, they are expected to enter the production shortly and eventually supersede their FORTRAN predecessors. Some production was also done with MadGraph [37] (vector boson scattering), CHARYBDIS [38] (black hole event generation), and CompHep [39, 40] (specific exotic physics models). Discussion of the generation of cavern background, beam halo, and beam gas events follow in Sects. 6.2.1 and 6.2.2. Single particle generators are also used to generate cosmic ray events and single particle events for performance studies and calibration of the detector.

Each generated event contains the particles from a single interaction with a vertex located at the geometric origin. Modifications to account for the beam properties are applied to the event before it is passed to GEANT4 (see Sect. 5.1). Particles with a proper lifetime $c\tau > 10$ mm are considered stable by the event generator. They can propagate far enough to interact with detector material before decaying. Their decays are handled by the simulation. Any particles with $c\tau < 10$ mm are decayed by the event generator, and their interactions with material or curving in the magnetic field of ATLAS are ignored.

3.1 Generator framework

Many external generator packages assume that the parameters for a particular job are set via a main program. This would require recompilation to change parameters. The Athena generator interfaces allow for the passing of all relevant parameters at run time, permitting a fixed software release to be used to produce different physics configurations. During initialization, the relevant parameters are passed via PYTHON fragments. The combination of the fragments, random number seeds, and the software release uniquely identifies the resulting data.[3] The Athena event manager is run for each event, and a run number and an event number are created; then the event generator is asked to produce an event. This event is created in memory in the format specific to the generator itself. The event must then be mapped into a common format so that subsequent algorithms are independent of the generator used.

ATLAS uses the HepMC event record [14], initially developed by the ATLAS collaboration but now supported by WLCG [17]. This is a set of C++ classes which holds the full event as produced by the generator. Stable particles are used as input to simulation; unstable ones can be of use in physics studies and diagnostics. Each event generator produces a very large number of stable particles (e.g. muon, kaons, pions, electrons, photons), a much larger number of unstable particles (e.g. gluons, quarks, B mesons, heavy hyperons), and, possibly, other objects (e.g. "strings" or "color singlet clusters") specific to an individual generator. The

[2] PYTHIA remains the default for current inclusive production, but EvtGen is likely to be used by default for the long-term production.

[3] Since pseudo-random number generators are chip architecture dependent, jobs are exactly reproducible only when run on the same type of processor (e.g. Intel or AMD).

HepMC record consists of a connected tree, navigation inside of which retains information on the event history including the parents of unstable particles. There is an important caveat here: the event generators are modeling quantum processes, and the event record has the structure of a classical decay chain. It is inevitable that compromises must be made and difficulties can arise from an over-literal interpretation of the tree structure. A very simple example is provided by events containing an e^+e^- pair. The parent of the e^+e^- pair cannot be uniquely specified, as the pair may arise from an intermediate Z boson, photon, or quantum interference. The HepMC event record is also used to contain the particle information from secondaries produced by interactions in the detector. This is discussed below in the section on Monte Carlo truth (see Sect. 3.6). Information about all interacting partons (e.g. momentum fractions x_1 and x_2) is saved, so that parton distribution function reweighting can be done without rerunning the event generation.

The FORTRAN generators usually use the HEPEVT common block [41] to store the information. Unfortunately, the different generators use slightly different structures. A separate translation into HepMC is needed for each one. The C++ generators such as HERWIG++ produce output in the HepMC format. No translation is required and the integrity of the HepMC event record is the responsibility of the generator authors.

3.2 General purpose generators

General purpose generators produce complete events starting from a proton–proton, proton–nucleus or nucleus–nucleus initial state. They are used standalone or with specialized generators that improve the description of certain final states. They have many parameters, some of which are related to fundamental parameters such as the QCD coupling constant and electroweak parameters, and some of which describe the models used to parametrize long distance QCD, soft QCD, and electroweak processes.

3.2.1 PYTHIA *and PythiaB*

PYTHIA [20] and HERWIG (see below) in their FORTRAN versions have been tested, used, and validated over many years in e^+e^- and hadron colliders. They start with a hard scattering process calculated to lowest order in QCD. They then add addition al QCD and QED radiation in a shower approximation which is most accurate when the radiation is emitted at small angle. The approximation is poorest in those cases with a large number of widely separated emissions of comparable energy. In addition, PYTHIA use a model for hard and soft scattering processes in a single event in order to simulate underlying activity. This model is used in the simulation of minimum bias events. While other generators may be used for specific final states, PYTHIA and HERWIG are the benchmarks.

ATLAS uses PYTHIA 6.4. There are two models of QCD radiation in PYTHIA. By default, ATLAS uses the showering model introduced in PYTHIA 6.3. This showering model is believed to better match the theoretical description of QCD showers. It produces somewhat more jet activity [42, 43], resulting in "busier" events than the older model which was used, for example, for detailed simulations at the Tevatron (see, for example, [44, 45]). In this model, the multiple scatters which make up the underlying event are interleaved with the parton shower according to the hard scale of the scatter or the emission. At the end of the shower, a phenomenological model is used to combine the quarks and gluons into hadrons. This hadronization model, which has many parameters, has been tuned by comparison with data in e^+e^-, ep, and hadron colliders [46, 47]. The underlying event model was retuned within ATLAS [48] to recover an acceptable description of the Tevatron data [49, 50]. PYTHIA contains a very large number of built-in processes, and new ones can be added by modifying the code. Hard scattering events can also be generated in a separate program in a standard format and fed into PYTHIA for the addition of a parton shower and hadronization. PYTHIA is the default generator in ATLAS: many hundreds of millions of events have been generated using it. Its ease of use, speed, and robustness make it an ideal choice for the default. It is supplemented by other generators, either to obtain some estimate of the uncertainties, or when specialized generators are expected to give a better physical description in certain final states.

PythiaB [21, 22] is an ATLAS-specific modification of PYTHIA aimed at the efficient generation of events related to B-physics. In PYTHIA, most high p_T bottom quarks are produced in the QCD shower of a high p_T light quark or gluon from a hard scattering process. Most showers do not produce such a $b\bar{b}$ pair, so using PYTHIA to generate B-physics events is inefficient. PythiaB reuses those QCD showers that contain a b- or c-quark, hadronizing them several times to increase the probability of producing a b-hadron. Since the probability producing a b- or c-quark in a parton shower is low, this procedure results in more efficient procedure of making b-hadron events without introducing any bias in the distribution of b-hadrons within the event. If a specific decay mode is then required the b-hadron decay can be forced using a modified b-hadron decay table, either in PYTHIA itself or via EvtGen.

3.2.2 HERWIG

ATLAS uses HERWIG 6.5 [23–25], the last release of the FORTRAN HERWIG package which is now superseded by HERWIG++ (see below). It is a flexible generator with a

large number of built in processes and has been tuned to agree with the Tevatron data [49, 50]. In particular, most of the generation of supersymmetric processes is done with HERWIG using the ISAWIG package [23–25] with the particle spectra and decay modes generated by ISAJET. ATLAS uses HERWIG with the Jimmy [51] implementation of the underlying event.

3.2.3 Sherpa

Sherpa [26] is a generator written in C++ which implements the CKKW duplicate removal prescription [52] to match fixed-order QCD matrix elements to QCD showers. It uses an interface to PYTHIA's hadronization model and produces complete events. It is expected to give better approximations for final states with large numbers of isolated jets than generators such as PYTHIA and HERWIG based on pure QCD showering. Sherpa generates underlying events using a simple multi-parton interaction model based on that of PYTHIA. For each new process to be generated, Sherpa must be recompiled to incorporate the specific libraries for the process of interest. On the Grid, this implies either recompiling Sherpa at the production site or deploying updated libraries for new production jobs. Instead, Sherpa is run locally to produce event files in Sherpa's native format. These files are then translated into the HepMC format with an additional Athena Grid job. It is also possible to run Sherpa entirely within an Athena job.

3.2.4 Hijing

Hijing [27] is a dedicated generator for the production of heavy ion events at all impact parameters. In a dense nuclear environment, such as appears in central collisions, a particle produced in a primary collision can re-interact several times as it propagates. Hijing models the propagation. It is also the only generator that can be used for proton–nucleus collisions occurring in beam–gas interactions. Hijing uses the PYTHIA hadronization model.

3.2.5 Single particle generators

A single particle event generator is frequently used for calibrating the detector, testing, and evaluating the reconstruction efficiencies. Although unphysical, these generators produce events with a single primary particle, for example a muon, electron, or charged pion, at a specified energy, position, and momentum direction. A range may also be specified for either the energy or direction. No underlying event, proton remnants, or other primary interactions are included when these events are generated.

A specialized single particle generator is used to produce cosmic ray events. Single muons are generated at the earth's surface in a square region (typically 600 m by 600 m) above the ATLAS detector and with the standard cosmic ray p_T spectrum [53, 54]. The upper and lower energy cutoffs of the spectrum are configurable. Those muons pointing to a sphere of configurable size (typically 20 m) centered at the geometric origin are propagated through the bedrock and the ATLAS cavern during simulation.

3.3 Specialized generators

Specialized generators do not produce complete events which can be passed directly to simulation. Rather, they are run in conjunction with one of the general purpose generators to improve the accuracy for specific decays or specific final states. Several of these specialized generators are "Les Houches" type generators. That is, they are run standalone using unmodified code from the generator author and produce an ASCII file containing partonic four-vectors in the "Les Houches" format [55, 56]. Athena uses a common interface that reads in these files and prepares them for processing in PYTHIA or HERWIG [55].

3.3.1 ISAJET

The FORTRAN generator ISAJET [34] is not used in large-scale production. However, it is used in conjunction with HERWIG for generation of supersymmetric events. Here, the ISASUGRA component of ISAJET is used to generate consistent sets of masses and decay modes for supersymmetric models. These are then loaded into HERWIG using the ISAWIG translation package, and HERWIG then generates complete final states.

3.3.2 Photos and Tauola

ATLAS uses the dedicated tau decay package Tauola to handle tau decays [31]. General purpose generators are set to treat tau leptons as stable. The events are passed to Tauola for decay. Because Tauola is a FORTRAN package, the events are extracted from the HEPEVT record. The Tauola interface is dependent on the generator that produced the tau, because helicities and helicity correlations are passed in generator-dependent formats. The original generator's results must be replaced, so both the input and output formats of Tauola are in fact generator-dependent. Special attention is paid to the polarization of the tau. In certain cases, for example the decay $W^{\pm} \to \tau^{\pm}\nu_{\tau}$, the polarization is known for the tau. In others, such as $Z \to \tau^{+}\tau^{-}$, there is a correlation between the polarization of the taus.

Photos handles electromagnetic radiation [32]. It is used by Tauola, and, therefore, Tauola cannot be used without Photos. Photos is also used to improve the description of electromagnetic radiation in, for example, the decay $W^{\pm} \to$

$e^{\pm}\nu_e$, where radiation distorts the electron energy distribution. In these cases the final state electromagnetic radiation is switched off in the general purpose generator, usually HERWIG or PYTHIA, to avoid double counting.

3.3.3 EvtGen

EvtGen [33], originally developed by the CLEO collaboration, provides a more complete description of B meson and hadron decays than that provided by defaults in PYTHIA or HERWIG. Recent modifications have been made to handle B_S and b-baryon decays, incorporating measurements from the Tevatron, BaBar, and Belle. In particular, EvtGen incorporates the best measurements of branching ratios and has theoretical models for unmeasured decay modes. It includes angular correlations, which impact the acceptance for certain decay modes of B mesons and baryons. It has been used for ATLAS studies involving the prospects for measurements of exclusive B decays.

3.3.4 Alpgen

Alpgen [28] is a "Les Houches" type generator enabling more sophisticated generation of certain final states. HERWIG or PYTHIA is then used to perform the hadronization and produce final (and initial) state QCD radiation. Alpgen is targeted at final states with several well-separated hadronic jets where the fixed order QCD matrix element is expected to give a better approximation than the shower approximation of PYTHIA or HERWIG. Alpgen is used, for example, to generate final states containing a W or Z and many jets. Alpgen also provides an algorithm to prevent double counting by event rejection. The Athena interface package includes the methods needed to pass events through HERWIG or PYTHIA and veto those events that would contribute to double counting. This process can be very inefficient for final states with large numbers of jets, and generation time can be significant.

3.3.5 MC@NLO

MC@NLO [29], which is also a "Les Houches" type generator, runs standalone to produce ASCII files which are then processed by HERWIG running inside of Athena. MC@NLO uses fundamental (hard scattering) processes evaluated at next to leading order in QCD perturbation theory. It is used, for example, to generate top events as it gives a better representation of the transverse momentum (p_T) distribution of top quarks than PYTHIA or HERWIG. MC@NLO includes one loop corrections, with the consequence that events appear with negative and positive weight which must be taken into account when they are used. Any resulting distribution will contain entries from both types of event, and, given sufficient statistics, the result will by physical (i.e. positive).[4] MC@NLO has been used for large-scale production of top, W and Z events. Only the parts of MC@NLO needed to read these events and process them via HERWIG are included in Athena releases.

3.3.6 AcerMC

AcerMC [30] is a "Les Houches" type generator aimed primarily at the production of W or Z bosons with several jets, including jets originating from b-quarks. A partonic final state is obtained by running it standalone and making an external ASCII file. Only the parts needed to read these events and process them via PYTHIA are included in Athena releases.

3.4 New C++ generators

3.4.1 PYTHIA 8

PYTHIA 8 [35] is a rewrite of the FORTRAN PYTHIA in C++ with new and expanded physics models. It provides a new user interface, transverse-momentum-ordered showers, and interleaving with multiple interactions. The program is under intensive tests and it will require some further tunings before it can replace the Pythia6 code as a leading generator. It is, however, interfaced to Athena and used for generator studies in ATLAS. It includes support for both "Les Houches" and HepMC event formats.

3.4.2 HERWIG++

HERWIG++ [36] is the C++ based replacement for HERWIG. It contains only important processes from the Standard Model, the universal extra dimensions model, and supersymmetric models (whose details are specified via Supersymmetric Les Houches Accord model files [58, 59]). Additional hard scattering processes can be used via "Les Houches" input from specialized generators, and additional decay models can be added by users.

HERWIG++ will soon be used for generation of some Standard Model processes, notably W and Z production. It will also be used for supersymmetric processes, because it includes full spin correlations and QCD radiation in the supersymmetric decay chains. The current version of HERWIG++ also incorporates an underlying event model based on the extension of Jimmy [51] to include soft scatters [60] and can thus potentially generate minimum bias physics.

[4] An alternative tool, POWHEG [57], implements essentially the same physics and produces events with only positive weight. Once it includes all the processes that MC@NLO does and has been validated, it is expected to take the place of MC@NLO.

3.5 Parton distribution functions

Parton distribution functions (PDFs) are used to describe the substructure of the proton and are used by all the event generators as external inputs. ATLAS uses the Les Houches Accord PDF Interface (LHAPDF [61]) library which is a replacement for PDFLIB [62] which provides a large repository of PDFs. CTEQ [63] PDFs are used by default (MC@NLO uses NLO PDFs, and all other generators use LO PDFs). There is a correlation between the PDFs and the tuning of parameters connected to initial state radiation [64, 65]: inconsistent results can be obtained by varying the PDFs in isolation. Therefore, when a new set of PDFs is used, the parameters of the event generator are retuned to produce consistent results [42].

3.6 Monte Carlo truth

The entire connected tree of the HepMC event record is stored as the Monte Carlo truth. Only the stable particles are propagated by the simulation. The various status codes and event history provided by the individual generators are retained within the HepMC event record. Unfortunately, much of this information is specific to a particular generator. Only status codes 1 (stable) and 2 (unstable) have a general meaning: the remaining values are used differently by the individual generators. As remarked in Sect. 3.1, there can be ambiguities resulting from the attempt to represent a quantum process by a classical tree. Some filters have been provided to select HepMC particles that, for example, are stable at the generator level or are non-interacting (e.g. neutrinos).

When the simulation is run, the HepMC tree from the event generator is copied, and some particles resulting from decays within, or interactions simulated by, GEANT4 are added to the copy (see Sect. 5.3). In this way, a complete event including both the generator and simulation information is provided. In order to ensure consistency, a particle decayed by GEANT4 but considered stable by the generator (such as a K_S) has its status code changed when the copy is made. A particle that has status code 2 after simulation will be identified as stable at the generator level, if the decay took place in GEANT4. GEANT4 secondaries are distinguished from those from the generators by an offset applied to their numerical identifier. The resulting Monte Carlo truth record can be large and account for a significant fraction (∼30%) of the disk space used by a simulated event after reconstruction.

3.7 Default parameters, tuning and bug fixing

The generator authors define default parameters. In some cases, however, these parameters are not tuned for use at the Large Hadron Collider and are superseded by parameters obtained by comparisons to data. The criteria for a particle to be considered stable are modified for use in ATLAS, for example. Once high-energy data appear, it is expected that retuning of the parameters will occur. These tunings can be made by varying parameters at run time. Once a new tuning is available, it can be loaded as a PYTHON fragment at run time or hard coding the values into the generator interfaces. In either case, the tuning becomes available as part of the next Athena software release and will be enabled by default. The settings can be overridden if needed or the previous defaults re-established. It is important to note that the parameters are often not independent and a complete set must be used. Arbitrary adjustments of a few parameters may result in inconsistent results. One of the most important sets of tunings is concerned with structure of minimum bias events and spectator processes in a hard scattering event: the underlying event. At present, these tunings are obtained for both PYTHIA and HERWIG [42] by first tuning to the Tevatron data and then extrapolating. The extrapolation from the Tevatron relies on the models used by PYTHIA and HERWIG. This extrapolation has had testing from comparisons of the Tevatron data at 630 and 1800 GeV [46, 66]. A high priority task for the ATLAS simulation as data accumulates is the testing of these tunings and changing of the parameters as needed.

4 ATLAS detector description

The ATLAS detector is described in detail in Ref. [1], but its main features will be summarized here. We discuss the geometry used in the simulation, which as much as possible matches the as-built detector. A cut-away view of the entire detector is shown in Fig. 2. ATLAS comprises several concentric components. The subdetectors are:

- A Beam Conditions Monitor (BCM) and Beam Loss Monitor used for detecting dangerous conditions and triggering an abort in the detector system. The BCM is located 1.84 m from the interaction point (IP) at $|\eta| \sim 4.2$.[5]
- A tracking detector composed of a fine granularity pixel detector with three layers covering $|\eta| < 2.5$, a silicon strip tracker (SCT) with eight layers determining four space points covering $|\eta| < 2.5$, and a transition radiation tracker (TRT) which has 32 space points on a typical track, covering $|\eta| < 2.0$.

[5]Pseudorapidity, $\eta \equiv -\ln\tan(\theta/2)$, where θ is the polar angle measured from the beam pipe. The other coordinate variables used are typically r, z and ϕ, where the x-axis points towards the center of the LHC ring, the y-axis points up, the z-axis defines a right-handed coordinate system, $r \equiv \sqrt{x^2+y^2}$, and ϕ is the azimuthal angle defined such that $\phi = 0$ along the x-axis.

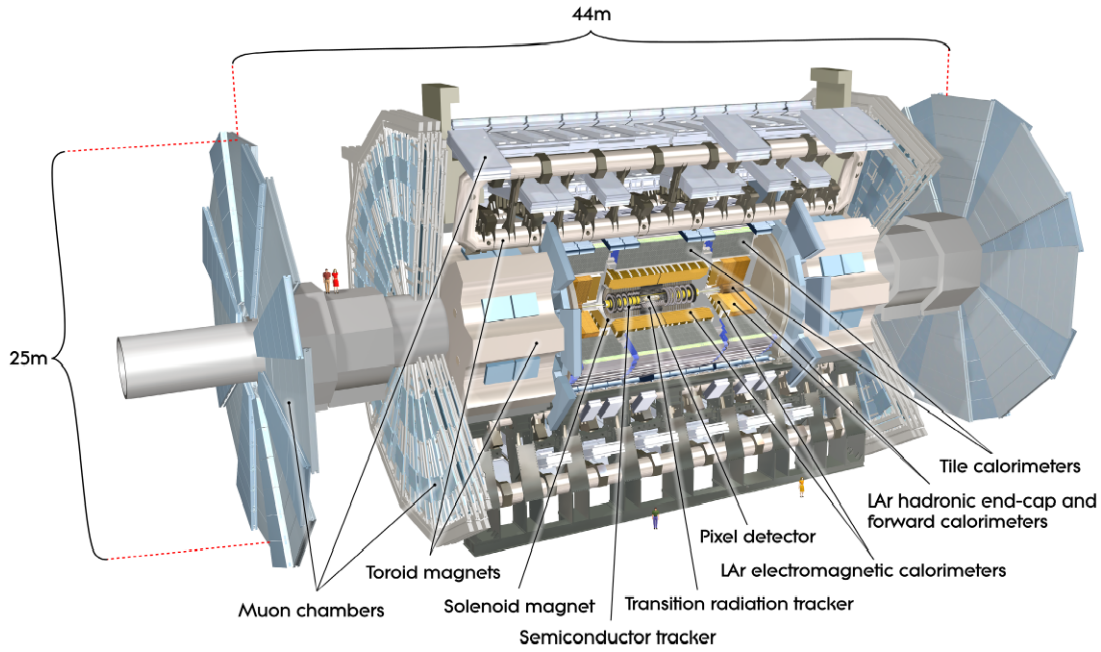


Fig. 2 ATLAS detector view

- Hermetic calorimetry composed of liquid argon (LAr) electromagnetic calorimetry covering $|\eta| < 3.2$, scintillating tile hadronic calorimetry in the barrel ($|\eta| < 1.7$), sampling LAr hadronic calorimetry in the endcap ($1.5 < |\eta| < 3.2$), and LAr electromagnetic and hadronic forward calorimetry covering $3.2 < |\eta| < 4.9$.
- Four different types of muon chambers, two of which are high precision (monitored drift tubes, and cathode strip chambers) and two of which have a rapid response for the muon trigger (thin gap chambers, and resistive plate chambers), covering $|\eta| < 2.7$.
- Luminosity detectors, including a zero-degree calorimeter that sits 140 m from the interaction point, a detector that performs a luminosity measurement using Cherenkov integration (LUCID), and an absolute luminosity detector for ATLAS.

The ATLAS magnetic field is formed by a solenoid, providing a 2.0 T uniform magnetic field in the tracking subdetectors, and a toroidal magnet system, composed of a barrel and two endcap toroid magnets. In the inner detector, the field has small ϕ- and z-asymmetries due to the toroid field and perturbations from the iron nearby. The field in the toroidal system has approximate z- and eight-fold ϕ-symmetry and provides on average 2.5 Tm of bending power in the barrel and 5 Tm in the endcap. During a simulation run, the field map required about 30 MB of memory.

In the standard production simulation (see Sect. 2.2) the luminosity detectors are not included. They can be simulated in dedicated jobs, but keeping particles in such a high pseudorapidity region increases simulation time by approximately 50% per unit of pseudorapidity ($|\eta|$) per event (see Sect. 5.1).

Several layouts of the complete detector are available, including those that were used for recording cosmic ray events while the detector was being completed. Test stands are also supported with the same infrastructure. All these layouts are described in Sect. 4.4. As much as possible, the details of the detector geometry are preserved in the simulation layout. Some approximations are necessary for describing dead materials, for example bundles of cables and cooling pipes in the service areas of the detector. In these cases, the description only aims to match the general distribution of the material, including inhomogeneities in ϕ.

4.1 Simulated detector geometry

The geometry structure can be viewed in terms of solids, basic shapes without a position in the detector; logical volumes, solids with ad ditional properties (e.g. name or material); and physical volumes, individual placements of logical volumes. Table 3 shows the number of materials, solids, logical volumes, physical volumes, and total volumes created when constructing various pieces of the ATLAS detector. Not all volumes are equivalent, however: in the case of repeating structures, as in the sampling portion of the LAr calorimetry in particular, it is possible to define a single logical volume that is repeated in hundreds of physical volumes (known as volume parameterization). Because of nesting, one can also define dependencies that create many total vol-

Table 3 Numbers of materials, solids, logical volumes, physical volumes, and total volumes required to construct various pieces of the ATLAS detector. "Inner Detector" here includes the beampipe, BCM, pixel tracker, SCT, and TRT

Subsystem	Materials	Solids	Logical vol.	Physical vol.	Total vol.
Beampipe	43	195	152	514	514
BCM	40	131	91	453	453
Pixel	121	7,290	8,133	8,825	16,158
SCT	130	1,297	9,403	44,156	52,414
TRT	68	300	357	4,034	1,756,219
LAr calorimetry	68	674	639	106,519	506,484
Tile calorimetry	8	51,694	35,227	75,745	1,050,977
Inner detector	243	12,501	18,440	56,838	1,824,614
Calorimetry	73	52,366	35,864	182,262	1,557,459
Muon system	22	33,594	9,467	76,945	1,424,768
ATLAS total	327	98,459	63,769	316,043	4,806,839

Table 4 Numbers of physical volumes and memory required to build various pieces of the ATLAS detector in GeoModel. Here "calorimetry" is simply the sum of the liquid argon and tile calorimetry

Subsystem	Phys. volumes	Memory [kB]
Inner detector	56,838	22,268
Calorimetry	182,262	44,116
Muon system	76,945	31,524
ATLAS total	316,043	97,908

umes from the physical volumes used. In other cases, a single volume can correspond to a piece of shielding or support with a complex shape. One can see in this table the complexity of the ATLAS detector, with hundreds of materials and hundreds of thousands of physical volumes. Such a detailed detector description is crucial for accurately modeling, for example, missing transverse energy, track reconstruction efficiencies, and calori meter response.

Table 4 shows the number of physical volumes contained in each detector subsystem and the memory required to build each using the GeoModel library (see Sect. 4.3) [67]. As expected, the two are correlated, although differences in volume complexity invalidate a direct correspondence. The entire geometry must be translated into a GEANT4 equivalent, so the total memory required for the geometry of the entire ATLAS detector is almost 300 MB (see Sect. 8).

In creating such a complex, dense geometry, removing volume overlaps and touching surfaces provides a particular challenge. Any overlap of more than 1 picometer and any place in which two volume faces touch can lead to stuck tracks during the simulation, a situation in which a track in GEANT4 may not know in which volume it belongs. These stuck tracks result in a loss of the event, but they can be overcome by introducing small gaps between volumes, at the cost of an extra step for each particle moving through the transition region.

Many layouts are available corresponding to the various revisions of material. The material budget is constantly updated, so that the geometry description is as realistic as possible. During any major updates of detector geometry, the subdetectors are generally required to make all changes backwards-compatible so that all older geometries can be configured and run as normal. This requirement allows for a fair comparisons between software releases with consistent geometries. During any job, the user may choose to enable or disable portions of the detector. Each subdetector is responsible for including any necessary materials and elements for its own construction. In this way, only required elements and materials are used during simulation, and memory loads are reduced when not using the entire ATLAS detector. The switches for disabling portions of the detector generally correspond to the highest level of the tree-structure in the detector geometry (i.e. entire subdetectors, not pieces).

It is possible to apply detector "conditions" modifications to each chosen geometry layout. The detector misalignment can be configured by selecting misaligned layouts either for each subdetector or for the full detector at once. Each ATLAS subdetector sits within a well defined envelope, allowing each to shift and distort without colliding with any other. One set of misalignments is provided specifically for the detector simulation, which includes appropriate levels of misalignment without any overlapping volumes. One may alternately use a misalignment directly derived from the data, but these misalignments may include overlapping volumes, which can cause missed hits. The misalignments are generally applied to more than simply the active volume. That is, if a silicon module in the inner detector is moved, any volumes associated with it (e.g. physically attached electronics) are moved as well.

In digitization and reconstruction jobs, conditions may include detector information beyond misalignments (e.g. dead channels). The infrastructure is in place to record detector conditions in a database and, at run time, allow the

user to select conditions from a specific data taking run. Conditions and geometry versions selected by the user can be transferred from the simulation jobs to the digitization and reconstruction jobs so that no additional user interaction is required. These default versions may at any time be overridden by job options.

In order to study the penalties of a poor material description on jet resolution and missing transverse energy bias, a special geometry layout with material distortions was created [68]. Material distortions correspond to additional material added to half of the detector ($y > 0$) to approximate a poor material description.

4.2 Databases and configuration

Two databases are used to construct the detector geometry chosen by the user: one to store basic constants (the ATLAS Geometry database), and one to store various conditions data (e.g. calibrations, dead channel, misalignments) for the specific run chosen (ATLAS Conditions database) [69]. At CERN, large (terabytes) Oracle data bases are used, primarily because they are well supported and straightforward to update. With any stable software release, a small subset of data needed for Athena jobs is replicated from Oracle into SQLite [70–72]—file based databases—and is distributed to the production centers. The large I/O requirements of production jobs can overwhelm a central Oracle server and are better handled by relatively small SQLite files. These files can also be replicated to individual production nodes for local and rapid access. The database replica version to be used can be chosen at run time for each Grid job.

Both the geometry and conditions databases support versioning of the data. The data are organized in a tree consisting of branch and leaf nodes. The nodes in this tree can be "tagged," and one can create a hierarchy of the tags. Such tag hierarchies are uniquely identified by the tag of the root node, which is usually referred to as top level geometry or conditions tag.

A geometry database stores all fundamental constants for detector construction. Volume dimensions, rotations, and positions, as well as element and material properties including density, are all stored as database entries. New detector-specific tags may be created for inclusion in a global ATLAS geometry tag, where different tags generally correspond to different detector geometry revisions. At run time, the user can select a global geometry tag as well as detector-specific geometry tags to create the desired geometry. In addition to constants for detector construction, the geometry database contains links to external data files that may store, for example, magnetic field maps. These files are shipped with software distribution kits to production sites. By using links through the database, it is possible to select a magnetic field map based on the chosen geometry layout. The selection of field map based upon the name provided in the database, for example, can be overridden with job options.

A separate conditions database stores detector conditions data which are indexed by intervals of validity and tags. The entire detector may be optionally misaligned with a global misalignment tag, and the user may configure the job to use specific misalignment versions for each subdetector. The global misalignment is used frequently to study the performance of the entire ATLAS detector with misalignments of the expected as-built magnitude. The detector-specific misalignments allow studies of the effects of misalignment of a single subdetector assuming ideal alignment of the remainder of ATLAS. The inner detector, for example, has completed an alignment challenge, wherein simulated data was produced with misalignments, and the analysis group was challenged to align the detector as with data. The tile calorimeter, on the other hand, does not use any misalignments in its geometry. A variety of misalignments have been used in the lead-up to data taking in order to speed the process of global detector alignment and improve early physics studies.

During data collection, the alignment constants of the detector are recorded periodically in the central conditions database. The user is able to recreate the misalignment conditions for a specific run by selecting an alignment version, again by subdetector if desired, at run time.

4.3 GeoModel and translations

The ATLAS simulation, digitization, and reconstruction each run in distinct jobs, but they must be able to use the same detector geometry. Therefore, a complete geometry description is maintained that can be used by each step and is not specific to any. By using the geometry databases, it is already possible to read identical detector constants and run conditions.

For these reasons, ATLAS uses GeoModel [67], a library of basic geometrical shapes, to describe and construct the detector. GeoModel contains geometry features similar to those of GEANT4: basic volumes can be constructed, rotated, and shifted in space; subvolumes can be placed inside a volume; boolean volumes can be made by adding or subtracting primitives; volumes can be parameterized and repeated. For the digitization and reconstruction, this detector description is entirely sufficient to place hits, reconstruct tracks and objects, and complete all necessary calculations.

The GeoModel descriptions of most ATLAS subsystems are built using constants in the geometry database. However, a translator has been constructed that parses an XML description of a detector's geometry and builds a transient representation from GeoModel primitives at run time. This generic package can translate any valid XML description of

detector geometry into GeoModel format. It has been used for describing the geometry of the muon system's rather complicated dead material.

For the simulation, the geometry is translated entirely from the GeoModel to the GEANT4 format. All volumes and subvolumes are translated, constructed, and properly placed within the "world volume" (the volume allocated for the detector, at the edge of which particles cease to be simulated). All information tied to GeoModel, including position, rotation, and dimensions, are also translated into a GEANT4 equivalent. Once the geometry has been translated, all subsystems rely solely on their GEANT4 description. The GeoModel geometry is currently maintained in memory for the entirety of the job, though it may be released to ease memory pressure in the future. As shown in Table 4, this release can be expected to save 100 MB of memory. Sensitive detectors and particle range cuts (see Sect. 5.5), for example, are tied to the GEANT4 geometry by volume name and can be added at any time after the geometry has been constructed. Each change in detector description is particularly weighty in simulation, because any additional volumes must be built both in GeoModel and in the GEANT4 geometry.

4.4 Alternate layouts

In addition to the standard detector layouts, several commissioning layouts are available to the user for simulation of cosmic ray data taking. During detector assembly, cosmic ray data were taken for several runs using as many subdetectors as were available. Some of these subdetector configurations included calorimeter endcaps shifted out of position while the inner detector was being accessed and were missing large portions of the beam pipe that had not yet been installed. One such commissioning layout is shown in Fig. 3. For studies of cosmic rays and cavern background, it is possible to simulate the ATLAS cavern surrounding the detector as well as the bedrock surrounding the two shafts leading down from the surface.

Several different magnetic field configurations are also available for some of the full detector layouts. Fields with the toroidal magnets on and solenoid off or solenoid on and toroidal magnets off are provided. These magnet configurations have already been used for some cosmic ray data taking runs and may be used for brief periods during high-energy collisions as well. Field maps have also been constructed that reflect the as-built misalignments of the mag-

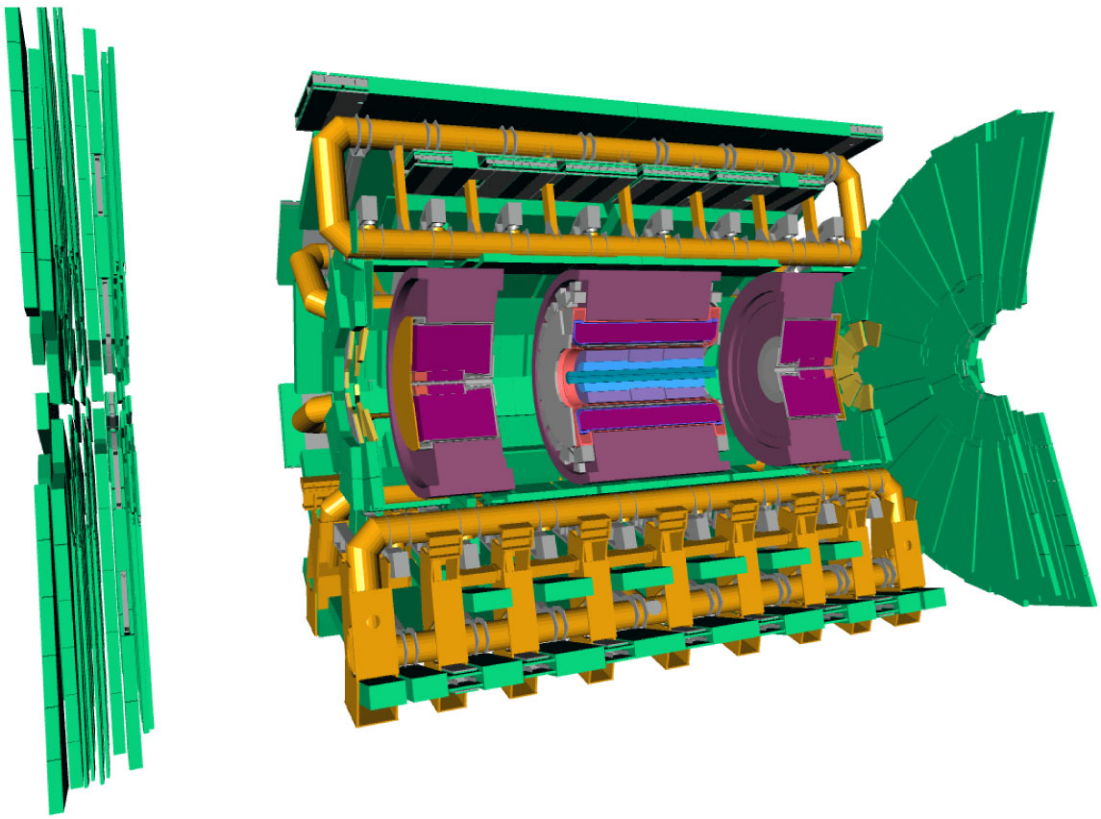


Fig. 3 Commissioning layout of the detector used for cosmic ray data taking during 2008. The endcap toroidal magnets and beampipe are not yet installed. The calorimeter endcaps (*purple*) are shifted by 3.1 m and the muon endcaps (*green*) are shifted to provide access to the inner detector during installation. The barrel toroid magnets are shown in *yellow*, and the inner detector is shown in *blue*

Table 5 Examples of test stands for ATLAS simulated using GEANT4

Subsystem	Incident particle	Energy
Hadronic Endcap calorimeter	$e^{+/-}$, $\pi^{+/-}$, $\mu^{+/-}$	6–245 GeV
Electromagnetic Barrel calorimeter	$e^{+/-}$	10–245 GeV
Electromagnetic Endcap calorimeter	$e^{+/-}$	10–200 GeV
Combined Endcap calorimeter	$e^{+/-}$, $\pi^{+/-}$, $\mu^{+/-}$	6–200 GeV
Hadronic Barrel calorimeter	$\pi^{+/-}$, p	5–350 GeV
Entire detector endcap wedge	$e^{+/-}$, $\pi^{+/-}$, $\mu^{+/-}$	1–350 GeV
Muon Detectors	$\mu^{+/-}$	20–350 GeV
Silicon Pixel Tracker Endcap	Cosmic Rays	0.5–200 GeV
Silicon Strip Tracker Barrel	Cosmic Rays	0.5–200 GeV

net system, for example a vertical shift of a 1.6 mm in the solenoid.

There are also several test stand layouts that were constructed to model test beam and standalone cosmic ray runs. A sample of these various test stands, including subsystems, incident particles, and energies, are listed in Table 5. A combined test beam run was taken with a wedge of the full detector during 2004 [73], and standalone test beams were constructed for the muon detectors [74, 75], tile calorimeter [76], and liquid argon calorimeter subsystems [77, 78]. The combined test beam setup is shown in Fig. 4. Cosmic ray data were also collected with various pieces of the inner detector [79, 80] and with the muon chambers both prior to and after installation.

All test stand and commissioning layouts are available as a part of the same geometry infrastructure and can be selected at run time for simulation. By maintaining all detector configurations as a part of a common infrastructure, it is possible to ensure consistency between, for example, the test beam and full detector simulation. Conclusions drawn from analysis of the test beam simulated data are generally still valid for the full detector simulation. The extensive tuning of the detector simulation and digitization on test beam data can be applied directly to the full detector. As many common elements as possible are kept between the two, including GEANT4 version and physics list (see Sect. 5).

5 Core simulation

The standard simulation of ATLAS relies on the GEANT4 particle simulation toolkit. GEANT4 provides models for physics and infrastructure for particle transportation through a geometry, but several ATLAS-specific pieces are provided as user-code. The detector geometry itself is constructed in the GEANT4 format, and all particle scoring (done in "sensitive detector" classes) are done on the Athena side. Each subsystem's scoring is optimized and tailored to store only what is necessary for accurately reproducing the performance of that particular subsystem [81–88]. Athena code is necessary to add to the Monte Carlo truth record. Physics models are chosen and parameters optimized for the ATLAS detector. The results shown in this paper used GEANT4 version 8.3 with official patch #2 and two modifications: updates for boundary represented volumes and a patch to the G4Tubs code. The software is continuously evolving, and ATLAS has moved to newer Geant4 versions since the writing of this paper.

The GEANT4 Collaboration and ATLAS Simulation Group have benefitted from 15 years of close collaboration. Frequently, new GEANT4 features have allowed faster or more realistic simulation of the ATLAS detector. Feature requests from the ATLAS collaboration have helped drive the development of GEANT4. The ATLAS simulation has also provided one of the more complicated test-beds for the GEANT4 toolkit, and GEANT4 has been extensively evaluated and validated during large-scale simulation production.

In order to provide PYTHON flexibility to the GEANT4 simulation, an additional layer of infrastructure is necessary. "Standard" GEANT4 simulation typically runs from compiled C++, and in order to modify any of the parameters or the geometry used in the simulation it is necessary to recompile. The Framework for ATLAS Detector Simulation (FADS) [89] wraps several GEANT4 classes in order to allow selection and configuration without recompilation of any libraries. Since a PYTHON interface is used for configuration, all the usual introspection capabilities of PYTHON may be employed. FADS wraps GEANT4 base-classes for volumes, materials, and sensitive detectors for hit processing as well as GEANT4 physics process definitions. These wrappers serve a dual purpose: first, they ease translations between the GEANT4 and Athena standards of geometry, hits, and particle storage. Second, FADS can catalogue the options available to the user, loading only those that will be needed for the desired simulation configuration while still providing all possibilities without any recompilation. Through FADS, a user is free to select a physics list (see Sect. 5.4) for use during the simulation. The user may also modify the physics list by adding particles or processes not included in the GEANT4 toolkit but included in the FADS catalogues, for instance in the simulation of long-lived exotic particles. Similarly, the detector description is configured with PYTHON dictionaries and FADS catalogues before it is built in GEANT4 and may be modified by the user. For example, sensitive detectors may be assigned to any volume in the detector. Range cuts (see Sect. 5.5) may also be added in the PYTHON and FADS layer prior to their being applied to any constructed GEANT4 geometry. Once the PYTHON configuration is complete, FADS objects are translated into their GEANT4 equivalents and loaded. Even after

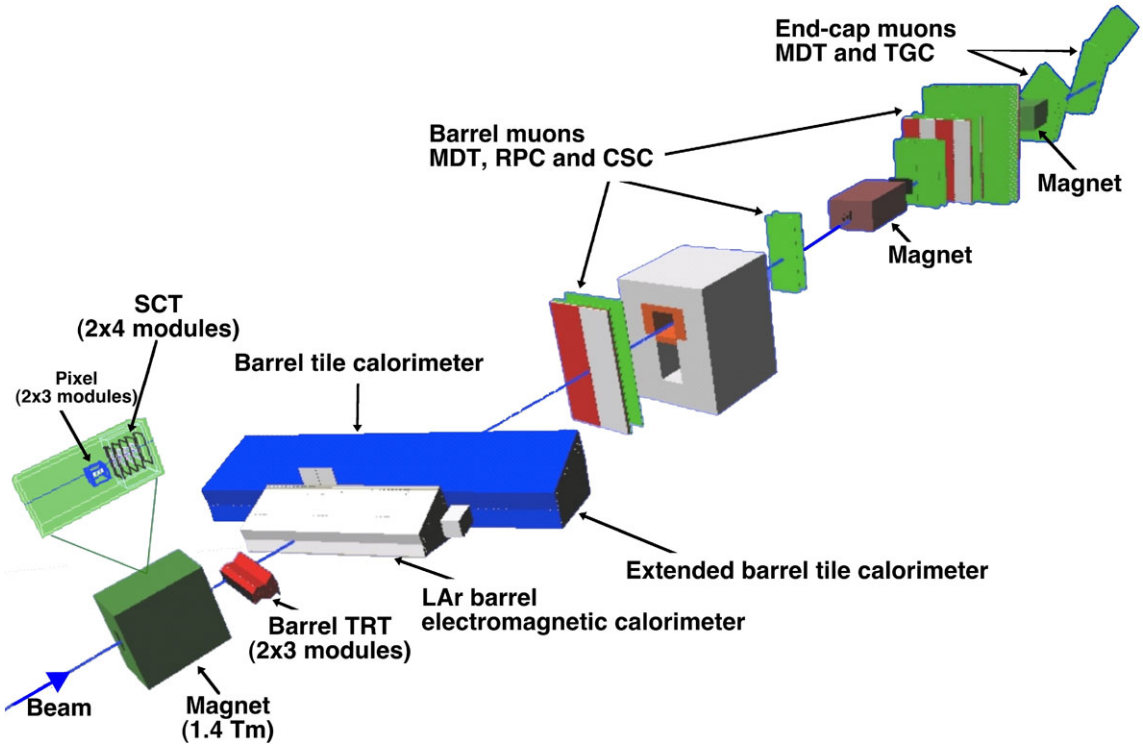


Fig. 4 Combined test beam setup from 2004

this translation, they can be modified through the standard GEANT4 user interface.

In order to fit into the Athena framework, a service for GEANT4 and an algorithm that calls the service during the event loop have been implemented [90]. The service wraps the event loop of GEANT4 and provides a few additional handles for user-configuration in the PYTHON layer. The service also takes care of initialization and finalization of each GEANT4 event. The generated events are translated from HepMC format into the standard GEANT4 event format prior to each event, and at the end of each event an analysis is done to ensure that the simulation finished without errors. Most of the functionality of the standard GEANT4 run manager is included in this service, so that any Athena-specific modifications (e.g. event translation from HepMC to GEANT4 format) to the usual GEANT4 event sequence can be made. The service also provides for interaction with GEANT4 through its standard user interface.

This section describes the possible user inputs, initialization, output, and various parameters of the simulation. Several useful features, including visualization, are also described.

5.1 Simulation input

The ATLAS simulation offers a choice for event generation. Events can be read from a file produced by any of the generators described in Sect. 3; one of the external generators can be configured and run concurrently; or commands can be provided for a single particle generator. The single particle generator can produce particles by the particle PDG identifier [91] with a configurable position and momentum. Neutral and charged geantinos (pseudo-particles without any interactions) are available for making material depth maps of the detectors and for debugging. The user may also choose to skip a certain number of events at the beginning of an input file, allowing 20 simulation jobs of 50 events each to a 1000 event input file without overlap or repetition.

Several cuts and transformations can then be made to the event. The vertex position is smeared to represent the luminous region within ATLAS.[6] It can be shifted if the user desires, but both the shift and smear are given initial default values that represent ideal collisions within the ATLAS detector. The generated event can be rotated in any direction, though only rotation in ϕ is physical. Primary particles are only passed through the detector simulation if they are within a specified range in η–ϕ. By default, primary particles with $|\eta| \geq 6.0$ are not simulated to save time. For example, typically beam-remnants are not simulated. This cut was chosen to ensure consistent response in the forward calorimeter: a cut at $|\eta| = 6.0$ allows a sufficient number of particles to scatter back into the forward detector from high-

[6] During early data taking, the beams will collide head-on. Therefore, no crossing angle is added to the simulated events for the time being.

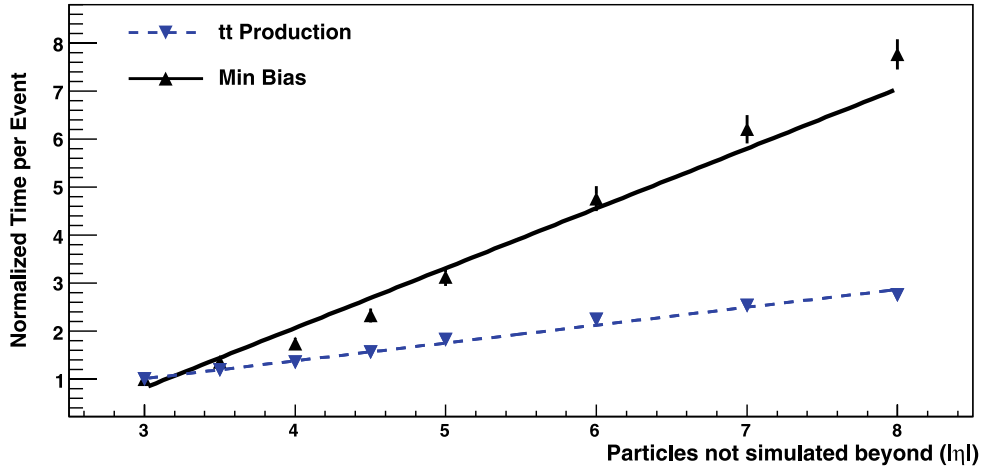


Fig. 5 CPU time per event as a function of the acceptance in $|\eta|$ of primary (generator) particles. If only central response is important, the CPU time required to simulate a single event is significantly different than if forward detectors must be included in the simulation. These two cases approximately represent the extremes of the plot. The time is normalized to CPU time for simulation of all primaries inside of $|\eta| < 3.0$ and increases between three-fold and eight-fold for simulation of all primaries in $|\eta| < 8.0$. The average of 200 simulated $t\bar{t}$ and minimum bias events was taken. Linear fits are overlaid

η without requiring an unacceptable amount of CPU time. Generally, an increase in one unit of pseudorapidity corresponds to an increase of 40–120% in CPU time, so that it is not possible to simulate the very forward detectors like LUCID during a standard simulation job. Figure 5 shows the η-dependence of the CPU time per event. The increase is approximately three-fold in $t\bar{t}$ events and eight-fold in minimum bias events from simulating particles in $|\eta| < 3.0$ to simulating particles in $|\eta| < 8.0$. The difference between the two types of events is primarily because the majority of activity in the minimum bias events is forward, and there is considerable central ($\eta < 3.0$) activity in the $t\bar{t}$ events.

At run time, either through job options or in the production system described in Sect. 2.2, seeds for pseudo-random number generators to be used by GEANT4, Athena, and any particle generators can be set. Different pseudo-random number generators may be configured for each. Since all random number seeds can be controlled, a single job is entirely reproducible. The seeds can also be written to a file or read from a file, providing an additional level of reproducibility.[7]

The user must also select a layout for the detector. As described in Sect. 4, several layouts of the full detector and various test stands are available. Combined test beam simulation requires such a different configuration that an independent but similar core drives the PYTHON configuration and loading of user job options. The distinction is made at run time based on the detector or test stand configuration selected. The layout of the detector determines what other options are available to the user at run time. During simulation of the entire ATLAS detector, several addition al options are available. For example, a neutron time cut (see Sect. 5.5) may be enabled. The magnetic field may be enabled and the field map may be selected.

The user may optionally select a set of run conditions for the simulation job, through which all options and flags are set and a pre-defined job is run. This option is particularly useful for testing and debugging.

5.2 Simulation initialization

Although the initialization in a standard Athena job occurs in a single step, for an ATLAS simulation job the initialization is broken into three steps to allow additional user intervention. Table 6 summarizes the processes that occur at each one of the three steps. The division of the initialization is such that most modifications to the simulation conditions can be accomplished in job options alone (i.e. without code modification). Normally, the user provides job options and allows the initialization to progress unhindered. Some parts of the job, for example the detector layout, are only loaded after the initialization has begun. In order to modify volumes after the layout is loaded, the user must intervene during the initialization. Only certain commands are effective at each stage of the initialization, since some parts of the GEANT4 simulation have been loaded and created while others have yet to be translated from dictionaries.

Stage one of the initialization occurs as soon as Athena is started. Several external PYTHON modules are loaded that provide basic functionality for any Athena job. The job properties provided by the user are read during this phase and

[7]Because of caching in GEANT4, it is not possible to reproduce an individual event or track without starting from the beginning of the job.

Table 6 Initialization sequence for the ATLAS simulation. Dividing the necessary configuration into several distinct steps allows user intervention at critical points

Init. stage	Processes
1	External modules loaded, job properties locked, metadata written, event generation configured, hit file initialized, GEANT4 service created
2	Detector, physics regions, range cuts created, GeoModel geometry translated, truth strategies initialized, magnetic field loaded, physics list selected, user actions initialized
3	Fast simulation models assigned, physics regions constructed, sensitive detectors assigned, GEANT4 run manager and physics models initialized, recording envelopes and visualization initialized

are locked. Once the job properties are locked, any significant modification to the running of a simulation job must be done by directly accessing the affected services and algorithms. This saves propagation of changes in the case of a late modification to a job property. Metadata that will be stored with the hit output file are gathered. External dependencies that require early initialization are loaded, providing a service for GeoModel, a service for database interaction, and a service for frozen showers (see Sect. 7.1). The event generation mode (reading external events, generating events from an external generator, or generating single particles on-the-fly) is determined, and any necessary configuration is included for the generator. A stream is opened for the output hit file, if necessary, and hit containers for each enabled subdetector are added to the new file. Finally, a service is created to interface with and control GEANT4, although at this point GEANT4 is not fully initialized.

Stage two of the initialization begins with the construction of the detector in PYTHON dictionaries. Dictionaries of physics regions, range cuts, and volumes in which to apply step limitation (see Sect. 5.5) are constructed, and all key properties are assigned to the detector facilities or built in dictionaries for later addition to the geometry. Each enabled piece of the ATLAS detector or test setup is then recursively constructed in GeoModel according to the parameters specified in the geometry and conditions databases. After each subdetector has been constructed in GeoModel, it is translated recursively into an equivalent GEANT4 geometry. After this point in the initialization all volumes and regions are available to the user for modification. Next, the Monte Carlo truth strategies (see Sect. 5.3) are added to the simulation. The magnetic field is then loaded. Under normal circumstances, the field is a map loaded from an external data file, the name of which is specified in the geometry database according to the geometry layout selected. The user may optionally choose to load data from one of the magnetic field test configurations, rather than the standard ATLAS magnetic field, or to create a new, basic magnetic field. The physics list (see Sect. 5.4) to be used for simulation is also set at this point.

User actions are then initialized. GEANT4 allows a user to insert pieces of code in various places throughout the simulation event loop, including after each step, when each track is queued ("stacked"), before and after each event, before and after each track is simulated, and before and after each run.[8] By default, ATLAS includes user actions that monitor simulation time, memory, and the number of tracks generated during each event. A neutrino cut (see Sect. 5.5) is also implemented as a track-stacking action. Whenever a new track is queued, its type is checked. The LAr calorimeters also use end-of-event actions for merging hits to save space prior to storage. All truth storage strategies (see Sect. 5.3) are implemented as stepping actions that store interesting interactions based on the type of process, detector region, and energies of the particles involved. Users may also configure their own actions and add them to the simulation in the same way. Examples have been constructed for integrating interaction lengths or radiation lengths through the detector when making geantino maps, for stopping or killing particles if certain conditions are met, and for turning on additional output only under specific conditions in order to study a bug or issue without having to sift through enormous log files.

Stage three of the initialization completes the job preparation. During this stage, the fast simulation models are built and added to the volumes to which they have been assigned. Any physics regions that will be used are constructed. Sensitive detectors are built and assigned to the regions of the detector that are to be made sensitive (i.e. in which hits will be stored). GEANT4's run manager and physics models are initialized. Recording envelopes are added (see Sect. 5.3), and any visualization that has been enabled by the user is initialized (see Sect. 5.7).

Once the initialization is complete and all the necessary elements have been loaded into memory, the event loop begins.

5.3 Monte Carlo truth information

The GEANT4 simulation adds to the Monte Carlo truth record already defined during generation (see Sect. 3.6).

[8] Here, run is used in the GEANT4 sense to refer to a finite set of events within a simulation job. Several runs may comprise a job, and each run may include an arbitrary number of events.

Far too many secondary tracks are produced during detector simulation to store information for every interaction. Only those interactions which are of greatest relevance to physics analyses are saved, according to several saving rules ("strategies"). Most are applicable only to the inner detector. For each interaction that satisfies any of the storage criteria, the incoming particle, step information, vertex, and outgoing particles are included in the truth record. Later in the software chain, individual track segments are recombined so that, for example, a single electron that undergoes several bremsstrahlung events along its path is counted as only one "true" particle.

The strategies include (with all cuts on kinetic energy)[9]:

- In the inner detector, bremsstrahlung vertices are stored if the primary electron or muon has an energy above 500 MeV and the photon produced has an energy above 100 MeV.
- In the inner detector, ionization vertices are stored if the primary particle has an energy above 500 MeV and the electron generated has an energy above 100 MeV.
- In the inner detector, hadronic interaction vertices are stored if the primary particle has an energy above 500 MeV.
- In the inner detector, decay vertices are stored if the decaying particle has an energy above 500 MeV.
- In the inner detector, the conversions of photons above 500 MeV are stored.
- In the calorimeter, muon bremsstrahlung vertices are stored if the primary muon's energy was above 1 GeVand the photon generated is above 500 MeV.

All cuts and regions of applicability are made configurable, so that any energy cut-offs can be modified and a strategy can be assigned to any volume in the simulation. additional rules could be constructed, for example, for tracking shower development within the calorimeter, but many would consume too much CPU time and disk space for use in standard simulation jobs.

Standard simulation jobs also define several volumes that are used to record all particles escaping part of the detector. All tracks above 1 GeV are typically recorded at the end of the inner detector, the end of the calorimeter, and the end of the muon system (and the end of the ATLAS world volume). It is possible for the user to configure the simulation at run time to add additional volumes to the list of these recording volume. In each case, tracks are saved as they exit each volume.

[9]For the most recent production, cuts are applied on transverse momentum, $p_T > 100$ MeV, rather than on kinetic energy. The lower cut allows for a study of tracking performance in minimum bias events where it may be possible to reconstruct tracks down to only a few hundred MeV.

5.4 Physics list

Physics lists include all numerical models that describe the particles' interactions in the GEANT4 simulation. Models are generally good for a single type of interaction and over a limited energy range. The GEANT4 collaboration provides several combinations of these models that have been tailored to various scenarios as standard physics lists that ship with each distribution. In order to enhance reproducibility and ensure that validated combinations of models are used, only those physics lists provided by the GEANT4 collaboration are used by the ATLAS simulation. One exception is allowed, namely transition radiation. Transition radiation is crucial for the tracking portion of the inner detector and is added to each physics list.

There are several physics lists that are used by ATLAS:

QGSP_BERT—the physics list used for all simulation production after 2008. The list includes the Quark–Gluon String Precompound model (QGSP) and the Bertini intranuclear cascade model (BERT) [4] as part of the hadronic physics package. The electromagnetic physics package includes step-limiting Multiple Coulomb Scattering (MSC).

QGSP_EMV—the physics list used for simulation production before 2008. This list included the QGSP model, but without the Bertini cascade. The MSC of this list was not allowed to limit the step, so it is labeled an electromagnetic variant (EMV).

QGSP_BERT_HP—the physics list used for neutron fluence studies and comparisons with the FLUKA simulation package [92, 93]. This list includes the QGSP and Bertini models, step-limiting MSC, and additional "high-precision" low-energy neutron physics models.

A step limitation process that controls the maximum allowed step length of a charged particle was added in the inner detector when using the QGSP_EMV physics list. It helped the simulation to better reproduced test beam and cosmic ray data. The step-limiting MSC that is a part of QGSP_BERT was found to agree equally well with data, and therefore the step limitation was removed from simulation with QGSP_BERT.

These physics lists were studied in detail for each subdetector [94]. Table 7 shows the number of steps, number of hits in sensitive detector regions, and number of secondary particles with kinetic energy above 50 MeV and 1 GeV within several regions of the detector and for the whole of ATLAS using both the QGSP_EMV and QGSP_BERT physics lists. Sensitive detectors to record calibration hits (described in Sect. 5.6) are included in the cryostat of the LAr calorimeter. The average was taken of 50 $t\bar{t}$ events, where there were on average 482 primary (generator-level) particles per event. The calorimeter clearly

Table 7 Number of steps, number of hits in sensitive detector (SD) regions, and number of secondary particles with kinetic energy above 50 MeVand 1 GeVwithin several regions of the detector and for the whole of ATLAS, using both the QGSP_EMV and QGSP_BERT physics lists. The average was taken of 50 $t\bar{t}$ events, where the average number of primary tracks per event was 482. Sensitive detectors to record calibration hits are included in the cryostat of the LAr calo rimeter

QGSP_EMV	Steps	Hits in SD	Sec. above 50 MeV	Sec. above 1 GeV
Inner detector	1.80×10^6	3.10×10^5	1,570	260
Calorimetry	1.87×10^7	6.87×10^6	39,900	2,040
Muon system	1.90×10^6	1,030	7,820	332
Total ATLAS	2.24×10^7	7.18×10^6	49,300	2,630
QGSP_BERT	Steps	Hits in SD	Sec. above 50 MeV	Sec. above 1 GeV
Inner detector	2.13×10^6	1.98×10^5	1,450	269
Calorimetry	3.93×10^7	1.36×10^7	40,100	2,170
Muon system	2.69×10^6	1,285	8,210	385
Total ATLAS	4.41×10^7	1.38×10^7	49,700	2,820

Table 8 Number of steps for various processes and detector regions during simulation with the QGSP_EMV physics list. The average was taken of 50 $t\bar{t}$ events, where the average number of primary tracks per event was 482. The "other processes" in the inner detector are primarily step limitation processes

Process	Inner detector	Calorimeter	Muon system
Transportation	1.50×10^6	9.33×10^6	2.15×10^5
MSC	4,910	1.09×10^5	5,200
Photoelectric effect	6,060	1.32×10^6	2.03×10^5
Compton scattering	12,800	1.43×10^6	4.26×10^5
Ionization	1.08×10^5	4.97×10^6	8.10×10^5
Bremsstrahlung	6,310	1.28×10^6	1.87×10^5
Conversion	434	82,400	17,300
Annihilation	291	82,800	17,800
Decay	254	2,320	538
Other hadronic interaction	1,710	1.23×10^5	21,600
Other process	1.56×10^5	4,800	831
Total	1.80×10^6	1.87×10^7	1.90×10^6

dominates the total number of steps and hits in sensitive detector for both physics lists. The muon system, though it has a comparable number of hits, consists mostly of shielding and therefore has far fewer hits in sensitive detector regions. The numbers of steps divided into different process types for QGSP_EMV and QGSP_BERT are listed in Tables 8 and 9. In both cases, transportation processes dominate the inner detector simulation, while electromagnetic physics and transportation dominate the calorimeter and the muon system.

Simulation time was also examined for each physics list. Simulation using the QGSP_BERT physics list consumes ∼2.5 times more CPU time than does simulation with the QGSP_EMV physics list. However, applying a neutron time cut (see Sect. 5.5) with the QGSP_BERT list reduces simulation time by more than 30%. Simulation with QGSP_BERT_HP requires approximately five times more CPU time than QGSP_EMV. Therefore, the QGSP_BERT_HP physics list cannot be used for standard simulation.

5.5 Simulation optimizations

In order to optimize use of both disk space and CPU time, several other modifications are made to the standard GEANT4 simulation [94, 95].

Comparing the QGSP_BERT physics list to the QGSP_EMV physics list, approximately three times as many neutrons are generated in typical hard scattering events, and they travel approximately three times further. These neutrons cause an increase in the output hit file size of approximately 75% as well as an increase in CPU time per event for hard scattering events. A GEANT4 neutron time cut is, therefore, applied which removes all neutrons

Table 9 Number of steps for various processes and detector regions during simulation with the QGSP_BERT physics list. The average was taken of 50 $t\bar{t}$ events, where the average number of primary tracks per event was 482. The "other processes" in the calorimeter and muon system are primarily neutron killer processes

Process	Inner detector	Calorimeter	Muon system
Transportation	1.76×10^6	1.46×10^7	2.31×10^5
MSC	2.31×10^5	1.48×10^7	5,200
Photoelectric effect	6,760	1.37×10^6	2.32×10^5
Compton scattering	14,800	1.66×10^6	5.03×10^5
Ionization	1.03×10^5	4.81×10^6	9.71×10^5
Bremsstrahlung	6,060	1.22×10^6	1.92×10^5
Conversion	416	86,800	18,100
Annihilation	271	87,000	18,500
Decay	212	1,670	402
Other hadronic interaction	2,190	6.66×10^5	1.23×10^5
Other process	426	25,400	5,720
Total	2.13×10^6	3.93×10^7	2.69×10^6

150 ns after the primary interaction. This cut was found to be sufficient time for the hadronic shower development and did not degrade the energy scale or energy resolution of the calorimeters. The total energy deposition of the shower is changed by ∼1%. Output files are the same size when using the QGSP_BERT physics list with this cut enabled as they are when using the QGSP_EMV physics list without a neutron time cut. The simulation time required for QGSP_BERT is reduced by 10–15% when the neutron cut is enabled.

Neutrinos are also removed as soon as they are created in the simulation. No particle is allowed by GEANT4 to step through more than one volume at a time. Therefore, neutrinos may require several thousand steps to exit the entire ATLAS detector. They may therefore consume a noticeable fraction of simulation time, even though their interaction probability is practically null. The removal is done when the particles are stacked.

Range cuts are GEANT4 parameters that control the creation of secondary electrons or photons during bremsstrahlung and ionization processes. If the expected range of the secondary is less than some minimum value, the energy of that secondary particle is deposited at the end of the primary particle's step and no separate secondary is produced. Effectively, this parameter defines an energy scale at which particle propagation may be ignored. By increasing the range cuts throughout the detector one can decrease the CPU time required per event. Particularly near boundaries and thin materials, the detector's sampling fraction may be affected if the range cuts are too large. Range cuts can be specified separately for electrons, positrons, and photons, but in ATLAS the same distance is used for all three. Range cuts are specified as a distance, and for each material the distance is translated into an energy based on the average energy loss of a particle in that material. For the majority of the ATLAS detector, range cuts take a default value of 1 mm. Exceptions are listed in Table 10. Deviations usually occur in sensitive volumes that are very thin, where it is important to correctly calculate the sampling fraction of the detector or model the energy deposition. Reduced range cuts are also applied to very thin volumes that are adjacent to sensitive volumes for the same reason. In the monitored drift tube muon chambers, for example, range cuts are only reduced in the thin aluminum tubes surrounding the sensitive detector (gas)—the gas itself takes the standard 1 mm cuts. In some shielding volumes it may be possible to relax range cuts considerably without degrading physics performance.

GEANT4 uses a set of parameters to control errors and accumulated biases on charged particles transportation through a magnetic field. Because the equation of motion is solved numerically, the user must select the numerical integration method to be used, including the order of integration, and the tolerances on the errors of the step. ATLAS has chosen to use the GEANT4 standard fourth-order Runge–Kutta method with the default error parameters for the majority of the detector. These parameters are generally satisfactory and result in errors and biases that are less than the position resolution of the detector. In the inner detector, however, tracks were found to be shifted sufficiently that detector residuals were affected. Here, stepping parameters were tightened by an order of magnitude. Further optimization of the stepping algorithms of GEANT4 has been undertaken, including the configuration of the choice of stepper and stepping parameters as a function of the initial particle type, energy, and position within the detector. Such a configuration can allow more careful stepping of muons in the calorimetry without degrading the total performance of the simulation significantly. Muons in particular can accumulate a significant bias after passing through all the sampling layers of the calorimeter, making more accurate tracking necessary. As a fourth-order stepper requires four values of the magnetic

Table 10 Range cuts for detectors that do not take the ATLAS default of 1 mm

Subdetector	Range cut value
Silicon pixels and strips in the inner detector	0.05 mm
Gas in the transition radiation tracker	0.05 mm
Electromagnetic Barrel and Endcap calorimeters	0.1 mm
Forward calorimeter (all compartments)	0.03 mm
Aluminum tubing of monitored drift tube muon chambers	0.05 mm

Table 11 Hit collection size, in kB per event, by subdetector. The average was taken of 50 simulated $t\bar{t}$ events. Calorimeter calibration hits are hits in the dead material of the calorimeters stored for studying simulation-based calorimeter calibration schemes

Collection name	Size [kB/event]	Percentage of File
Silicon pixel tracker	82	4%
Silicon strip tracker	356	16%
Transition radiation tracker	921	46%
Electromagnetic Barrel calorimeter	89	4%
Electromagnetic Endcap calorimeter	104	5%
Hadronic Barrel calorimeter	29	1%
Hadronic Endcap calorimeter	22	1%
Forward calorimeter	42	2%
Calorimeter calibration hits	243	12%
Muon system (all collections)	3	<1%
Truth (all collections)	134	7%
Total	1987	100%

field to be calculated, optimization of magnetic field map access will also be key to improving the performance of the simulation's tracking.

5.6 Hit storage format

The output from the simulation is a hit file, containing some metadata describing the configuration of the simulation during the run, all requested truth information, and a collection of hits for each subdetector. The hits are records of energy deposition, with position and time, during the simulation. Each subdetector is responsible for implementing their own sensitive detector for the selection, processing, and recording of these hits. In most subsystems, including the inner detector and muon system, this consists simply of recording all hits that occur in sensitive regions of the detector for subsequent storage. Some additional manipulation is done at the end of each event to compress the output as much as possible; still, the files are typically 2 MB per event for hard scattering events (e.g. $t\bar{t}$ production).

The file size is large, mostly due to the inner detector, for which the majority of hits are independently stored. Merging hits there is difficult, since they tend to be isolated and cannot normally be merged across readout channels. These consume typically 60% of the disk space in a hit file (e.g. 65% of the hit file for $t\bar{t}$ events). In the calorimetry, there are far too many hits created by electromagnetic and hadronic showers for the individual storage of a four-vector for each (see Sect. 5.4). Instead, hit merging occurs at the end of each event. By optimizing time binning, hits can be compressed to a large extent. About 10% of the hit file is consumed by optional "calibration hits" for the calorimeters, hits in dead material, stored to improve the detector calibration and missing energy calculation and to study simulation-based calorimeter calibration schemes. Under normal circumstances, the muon systems contribute a negligible portion of the hit file. The contributions by subdetector can be found in Table 11 for the average of 50 simulated $t\bar{t}$ events. Here and elsewhere in this paper, file sizes are without compression and are taken from ROOT [96]. In practice, compression reduces the actual disk space required for the files, but file-level metadata adds several hundred kilobytes.

By comparing Tables 7 and 11, one can understand these numbers in terms of hits in the sensitive detector region. Although the muon system is large, the majority of it is shielding. Therefore, it collects far fewer hits than the other subsystems and requires less disk space for the hit records. The calorimetry produces 95% of the hits in sensitive regions during simulation. Because of the compression applied prior to storage, the calorimetry comprises only 25% of the hit file.

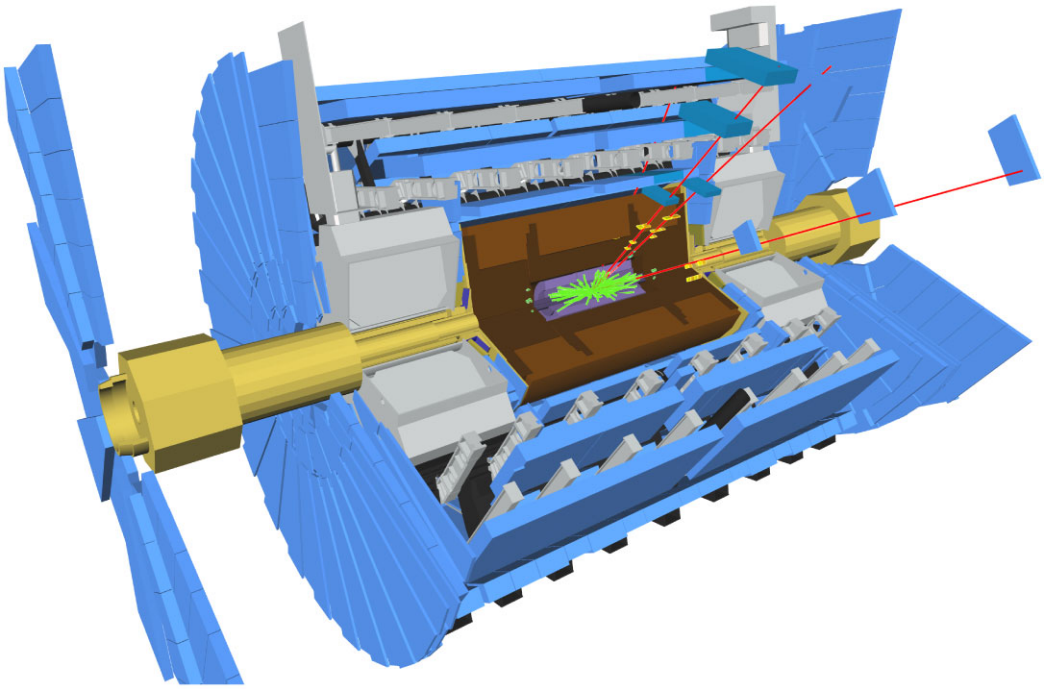


Fig. 6 An event display made with VP1. A Higgs boson decays into four muons (shown in *red*). Inner detector tracks are in *green*, and energy deposited in the calorimeter by the muons is shown in *yellow*

5.7 Visualization

Visualization is used to understand anomalies or features in odd events, occasionally to debug errors due to geometry, and to check for overlaps and touching volumes in the geometry that can be spotted by eye. Although GEANT4 contains viewing software of its own, because the geometry of ATLAS must be translated from GeoModel into GEANT4 format it is useful to use a viewer that can construct geometry directly from its GeoModel description. A general purpose three-dimensional event display program, VP1[10] [97], has been developed specifically for ATLAS. It is optimized specifically for the visualization of the ATLAS geometry and is arguably the most useful tool for understanding and debugging of detector description across all ATLAS subsystems. Two examples of VP1 event displays are in Figs. 6 and 7. Ray Tracer, the GEANT4 visualization utility, has also been used to visualize portions of the detector containing some exotic shapes.

VP1, as well as other event display programs used in ATLAS (e.g. Atlantis [98] and Persint [99]), are mostly used for visualizing real and simulated events after they have run through the common reconstruction software. The VP1 viewer can be injected directly into the simulation job in order to visualize events immediately after the simulation step.

[10]ATLAS is at Point 1 of the LHC ring. The name VP1 is short for Virtual Point 1.

6 Digitization

The ATLAS digitization software converts the hits produced by the core simulation into detector responses: "digits." Typically, a digit is produced when the voltage or current on a particular readout channel rises above a pre-configured threshold within a particular time window. Some subdetector's digit formats include the signal shape in detail over this time, while others simply record that the threshold has been exceeded within the relevant time window.

The peculiarities of each subdetector's charge collection, including cross-talk, electronic noise and channel-dependent variations in detector response are modelled in subdetector-specific digitization software [79, 82, 100–102]. The various subdetector digitization packages are steered by a top-level PYTHON digitization package which ensures uniform and consistent configuration across the subdetectors. The properties of the digitization algorithms were tuned to reproduce the detector response seen in lab tests, test beam data, and cosmic ray running. Dead channels and noise rates are read from database tables to reproduce conditions seen in a particular run. In some cases, dead channels are removed during the reconstruction step.

The digits of each subdetector are written out as Raw Data Objects (RDOs). For some subdetectors this requires the digits produced to be converted to RDOs by a second algorithm during the digitization process. For others there is no intermediate digit object and RDOs are produced di-

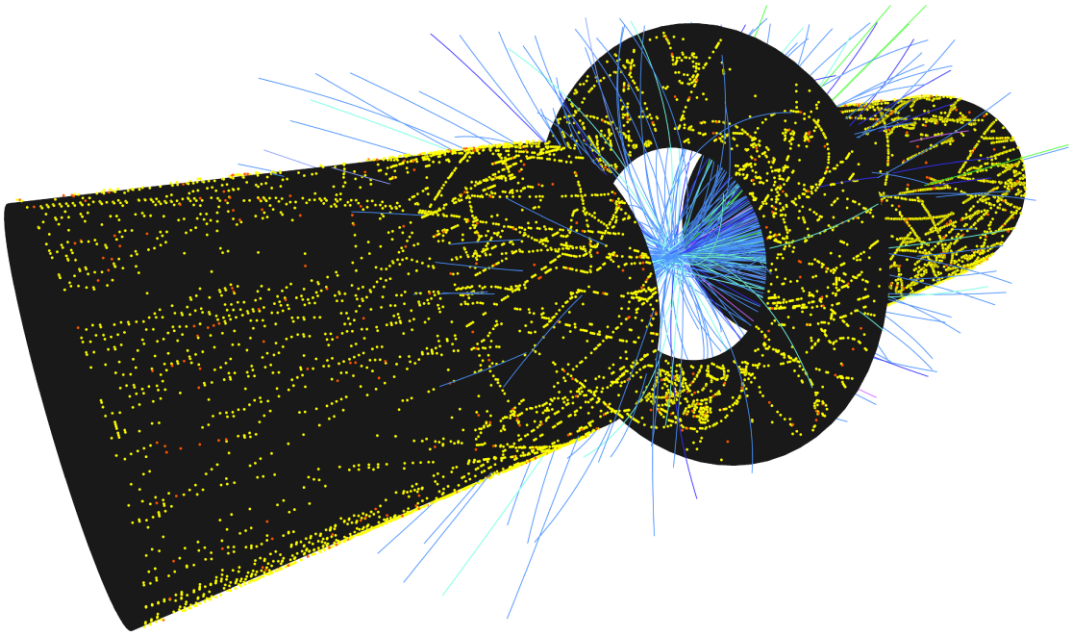


Fig. 7 A Higgs boson decaying into four muons, with only the inner detector tracks and hits in the TRT being displayed by VP1

rectly from the hits. In addition to RDOs, the digitization algorithms can also produce Simulated Data Objects (SDOs). These SDOs contain information about all the particles and noise that contributed to the signal produced in the given sensor and the amount of energy contributed to the signal by each. The relationship between RDOs and SDOs depends on the particular subdetector. For example, in the SCT each RDO represents a group of consecutive strips which recorded a hit, whereas one SDO is produced for each strip where energy was deposited by a particle in the Monte Carlo truth tree. No SDOs are created in the calorimeter. SDOs are mainly used for determining tracking efficiency and fake track rates.

Simulating the detector readout in response to a single interesting hard scattering interaction is unrealistic. In reality, for any given bunch crossing there may be multiple proton–proton interactions. In addition to the hard scattering which triggers the detector readout, many inelastic, non-diffractive proton–proton interactions may appear. These interactions must be included in a realistic model of detector response. The effects of beam gas and beam halo interactions, as well as detector response to long-lived particles, must be incorporated. These interactions are treated separately at the event generation and simulation stages. Within a digitization job, hits from the hard scattering are overlaid with those from the requested number of these additional interactions before the detector response is calculated. Because of long signal integration times, most subdetector responses are affected by interactions from neighboring bunch crossings as well. Therefore, additional interactions offset in time are overlaid as necessary. The overlaying off these various types of events, known collectively as "pile-up," is described in Sect. 6.2.

Before reconstruction can be run, bytestream data from the real detector must be converted into RDO format. As mentioned above, the digitization usually avoids this step by writing out RDOs directly. However, in order to do simulation studies with the High Level Trigger it is necessary to translate the RDO files into bytestream format. There is some loss due to truncation in the first conversion from RDO to bytestream, but the inverse operation is basically lossless. Having the ability to convert output in both directions also allows evaluation of the conversions themselves.

6.1 Digitization configuration

The ATLAS digitization takes as input hit files produced by the ATLAS simulation. For pile-up simulation, there are also input hit files for each type of background interaction to be overlaid. In such cases it is the main hard scattering event which sets the run number and event number. Run and event numbers from overlaid events are ignored.

The digitization steering package exists entirely in the PYTHON layer and configures how the digitization will be performed before the event loop starts. This configuration is highly flexible, but also ensures that sensible default values are given for each configurable property of the job. In the configuration of digitization jobs, the user may specify the number of events to digitize, the number of leading events to skip in the input file, the input hit file(s), and the output file. Digitization and writing out of RDOs may be enabled or disabled by subdetector. In order to ensure consistency,

the detector layout version is, by default, read from the hard scattering events' hit file metadata.

Digitization options also include the following:

Detector Noise Simulation: Detector noise simulation can be turned off in the inner detector, calorimeter or muon spectrometer or any combination thereof. This is useful for data overlay jobs where noise is taken from real data events and for studies using a noise-free detector.

Random Number Services: The type of random number engine to be used in all digitization algorithms can be specified (Ranlux64, the default, or Ranecu [103]). Each algorithm has one or more random number streams. Random number seeds can be initialized from a text file or set in job options. The user may alternately specify an offset from the default values of the seeds, to be used in all streams.

Metadata: In the default configuration, metadata from the simulation stage are used to configure the physics list (for setting the sampling fraction of the calorimeters) and the detector layout. The metadata can be overridden.

Pile-up Background Events: The overlay of minimum bias, cavern background, beam gas and beam halo events can all be configured separately. In each case the mean number of events (if any) per bunch crossing to be overlaid and a collection of files containing the events to be overlaid onto the signal events can be specified.

Beam Properties: The LHC beam bunch spacing can be configured, as can the number of bunch crossings to overlay before and after the hard scattering event.

Detector Conditions: Default detector conditions (including, e.g., dead electronics and noisy channels) are associated with each detector layout. Non-default conditions may be specified globally or by subdetector for use in digitization.

After a check to make sure that at least one subdetector has been left switched on, the input and output streams are initialized. GeoModel is initialized using the detector layout and conditions versions read from the hits file metadata or specified by the user. Setting the detector layout version to be different from that used in the simulation is possible, but considered to be an expert action. The magnetic field service is initialized at this point. It is necessary because the magnetic field affects charge propagation from the active regions of the detector to the readout surfaces.

At this point, caches for pile-up events are created and configured with the appropriate collection of hits files as well as the number of events to be overlaid per bunch crossing. These caches are controlled by an overall pile-up manager service. A second pile-up service is created to hold information about the time window within which interactions can affect the response recorded by each subdetector. During the initialization stage, this information can be combined with the bunch spacing to calculate the number of bunch crossings which should be simulated for each subdetector for each event.

Subsequently, the subdetector digitization algorithms are configured and added to the sequence of algorithms to be run in the job. The collections of RDOs, hits, and truth information which are to be recorded are added to the output stream. Digitization algorithms exist for the following subdetectors:

Inner Detector: BCM, silicon pixel tracker, SCT, and TRT.

Calorimeter: LAr and tile calorimeters. Separate algorithms also exist to simulate the formation of trigger towers in the calorimeters, which serve as inputs to the level one trigger.

Muon Spectrometer: Cathode Strip Chambers, Monitored Drift Tubes, Resistive Plate Chambers and Thin Gap Chambers.

If requested, the level one trigger simulators are added to the algorithm sequence, provided that the digitization of the relevant parts of ATLAS have been turned on. The default mode of simulation production is to run the level one trigger simulation during the reconstruction step rather than as part of the digitization step.

As the digitization algorithm for each subdetector is configured, the names and seeds for the random number streams it requires are added to a list. In the case where seeds are to be read in from a file, the default list of stream names and their seeds are replaced by the file contents. Once all algorithms have been configured, the list is used to configure the random number service. Separate random number streams are used for each subdetector digitization algorithm and give the same result independent of what is used for the other subdetectors.[11]

Much of the job configuration information, along with the detector layout version, is written to the output file as digitization metadata. The run number provided in the simulation metadata is used to establish a validity range for the digitization metadata corresponding to the current run only. At this point the digitization job is fully configured and the event loop begins.

6.2 Pile-up

To simulate pile-up, various types of events are read in, and hits from each are overlaid. The different types considered can be configured at run time, and normally comprise signal, minimum bias, cavern background, beam gas, and beam halo events. The number of events to overlay of each type

[11] Here "digitization algorithm" does not include the calorimeter trigger tower simulation algorithms, which require the corresponding calorimeter digitization to be performed. Similarly, the level one trigger simulation requires the simulation and digitization of the expected trigger inputs to give meaningful results.

Table 12 The time window (relative to the current bunch crossing) during which interactions in each subdetector are simulated during pile-up jobs, along with the corresponding numbers of bunch crossing simulated in the case of 25 ns bunch spacing and 75 ns bunch spacing

Subdetector	Simulation window [ns]	No. Bunch crossings (25 ns bunch spacing)	No. Bunch crossings (75 ns bunch spacing)
BCM	−50, +25	4	1
Pixel trackers	−50, +25	4	1
SCT	−50, +25	4	1
TRT	−50, +50	5	1
LAr calorimeter	−801, +126	38	12
Tile calorimeter	−200, +200	17	5
Muon chambers	−1000, +700	69	23

per bunch crossing may also be set at run time and is a function of the luminosity to be simulated. The mean number of interactions per bunch crossing (BC), for example 23 at the design luminosity of 10^{34} cm^{-2} s^{-1} with 25 ns bunch spacing, depends linearly on luminosity and bunch spacing. However, this number is Poisson-distributed, with a long tail beyond the most probable value. Thus, a substantial fraction of the bunch crossings will have more than the average number of interactions. In addition, the ATLAS subdetectors are sensitive to hits several bunch crossings before and after the BC that contains the hard scattering event (which triggers the readout). Table 12 shows the simulation window for each detector along with the corresponding number of bunch crossings for 25 ns and 75 ns bunch spacing. All of these detector and electronic effects are taken into account during the pile-up event merging.

6.2.1 *Cavern background*

Neutrons may propagate through the ATLAS cavern for a few seconds before they are thermalized, thus producing a neutron–photon gas. This gas produces a constant background, called "cavern background," of low-energy electrons and protons from spallation. The cavern background consists mainly of thermalized slow neutrons, long-lived neutral kaons and low-energy photons escaping the calorimeter and the forward shielding elements. Muon detectors are most affected by high cavern-background rates. The radiation levels to be expected in the ATLAS cavern scale with luminosity, and they have been simulated as a function of r and z [104] for the design luminosity of 10^{34} cm^{-2} s^{-1}. Depending on the type of radiation, exact composition of the equipment, and sensitivity of the study, the rates sometimes have to be increased by a safety factor. Cavern background is produced in the following way:

- A standalone dedicated GEANT3/GCALOR-based [105] detector simulation program with improved neutron propagation and a simplified ATLAS detector geometry is run on proton–proton collisions. The cavern walls are not included in the detector description. The output of this program includes particle fluxes in the envelopes surrounding muon spectrometer chambers. The fluxes are provided as list of particles with all related parameters per proton–proton interaction at the entrance of each envelope.
- The kinematic information of all particles generated by GEANT3/GCALOR is converted to HepMC format, and the flux is modified to be uniform in the time interval of the required bunch spacing (typically [0, 25 ns]).
- The simulation is then carried out using the full detector geometry and GEANT4, and hits are stored.
- The cavern events are mixed, with a safety factor of up to 10, at the digitization level with the minimum bias and signal events.

There are a number of issues with the current simulation of the cavern background. The original primary cavern events were generated in an older version of PYTHIA where the generated particle density is a factor of two lower than in the newer versions of PYTHIA [42, 43]. The statistics for the available cavern events are limited: 40,000 events are available with a safety factor of 1; 10,000 events are available with a safety factor of 2 or 5; and 5,000 events are available with a safety factor of 10. Because of the limited statistics, a number of monitored drift tubes in the muon detector fire more often than expected (i.e. there are spikes in the hit response of the detector). Additionally, neutral particles are tracked through the entire detector during simulation, thus producing additional hits from particles that should have been removed at the edges of the muon chamber envelopes (multiple counting).

In the short term, the problem of limited statistics of the cavern events has been alleviated by taking advantage of the ϕ-symmetry of the muon spectrometer: the cavern events are rotated and re-simulated eight times or more (in multiples of eight). Further improvement in the available cavern statistics can be achieved by repeating the simulation of the cavern events many times with different random number seeds, since the probability of a neutral interaction is very low, of the order 1%.

Table 13 Container size on disk in RDO files. Columns two and three show the average of 50 $t\bar{t}$ events digitized in the absence of pile-up. These events were chosen because they produce large energy deposits throughout the detector. Columns four and five show the average of 50 $t\bar{t}$ events digitized in the presence of pile-up with 10^{33} cm^{-2} luminosity and 25 ns bunch spacing

Category	No Pile-up space on disk [kB/event]	No Pile-up percentage of file	Pile-up space on disk [kB/event]	Pile-up Percentage of file
Inner detector RDOs	187	7.6	322	11.5
Inner detector SDOs	247	10.0	333	11.9
Calorimeter raw channels	995	40.3	1006	35.9
Calorimeter calibration hits	601	24.3	601	21.4
Muon spectrometer RDOs	1	0.04	27	1.0
Muon spectrometer SDOs	1	0.05	59	2.1
Level one trigger	289	11.7	300	10.7
Truth	147	6.0	151	5.4
Headers	>1	0.01	2	0.1
Total	2469	100.00	2801	100.00

6.2.2 *Beam halo and beam gas*

Beam halo is the background resulting from interactions between the beam and upstream accelerator elements. The flux from upstream (in the tunnel and collimators) is provided by the LHC Machine Division [106, 107]. Beam halo events are generated as discrete particle losses against the upstream collimators. The LHC machine division has estimated the proton loss rate in design conditions as being on the order of 1 MHz. FLUKA simulation of the last 150 m of the beamline indicates that daughter particles from these proton losses will reach the cavern wall (23 m from the interaction point) at a rate of ∼400 kHz. This flux is input to the normal GEANT4 simulation to produce hit files.

Beam gas includes the residual hydrogen, oxygen, and carbon gasses in the ATLAS beam pipe. Beam gas interaction events are generated with Hijing (see Sect. 3.2.4) with appropriate time offsets. The interactions are allowed to take place anywhere in the beam pipe of ATLAS, 23 m in either direction from the interaction point.

6.2.3 *Pile-up with real data*

The pile-up mechanism described above will not work with real data, because it begins at the hit level. One must instead overlay events beginning from detector electronics output (RDOs). One may collect minimum bias, cavern background, beam halo, and beam gas backgrounds from the same "zero bias" trigger used to understand detector electronic noise. Then, one would overlay hits from simulated hard scattering events onto the zero bias trigger data to simulate the pile-up. The zero bias trigger data needed for this type of event overlay can be selected at random from the filled-bunch crossings.[12] The subdetectors should be read out with as little zero-suppression as is possible and with the HLT in "pass-through" mode (i.e. without further filtering). One can use bunch-by-bunch luminosity information to correctly weight the event sample for pile-up studies.

In principle one needs as many zero bias events as generated events, but in practice zero bias events can be reused with independent simulated data sets without introducing any bias. During data taking, zero bias events are sampled at all times, because detector and cavern conditions are likely to vary with time. The zero-bias events could, for example, be collected exactly one orbit after a high-p_T trigger has fired. Appropriately pre-scaled to the output rate needed for simulations (on the order of 1–2 Hz, or about 1% of the recorded events), this means that the rate follows the luminosity and that the bunch structure is guaranteed to be right.

6.3 RDO storage format

The ATLAS detector electronics produce data in bytestream format. The RDO format can be thought of as a POOL-compatible version of the byte stream.[13] The file size on disk is typically around 2.5 MB per event for hard scattering events (e.g. $t\bar{t}$ production) and increases in the presence of pile-up. Table 13 shows an example of the disk consumption by container for 50 $t\bar{t}$ events without pile-up and with pile-up at 10^{33} cm^{-2} s^{-1}. In the absence of pile-up, one of the main consumers of disk space are the calibration hit collections, as described in Sect. 5.6, which are copied directly from the hit file to the RDO. As pile-up luminosity increases however the inner detector containers become increasingly significant.

[12] The zero bias trigger is not a minimum bias trigger.

[13] POOL compatibility requires separate transient and persistent object representation.

7 Fast simulations

Because of the complicated detector geometry and detailed physics description used by the ATLAS GEANT4 simulation, it is impossible to achieve the required simulated statistics for many physics studies without faster simulation. To that end, several varieties of fast simulation programs have been developed to complement the GEANT4 simulation. In this section, the standard GEANT4 simulation will be referred to as "full simulation."

Almost 80% of the full simulation time is spent simulating particles traversing the calorimetry, and about 75% of the full simulation time is spent simulating electromagnetic particles. The Fast G4 Simulation aims to speed up this slowest part of the full simulation [108, 109]. The approach taken, therefore, is to remove low energy electromagnetic particles from the calorimeter and replace them with pre-simulated showers stored in memory. Using this approach, CPU time is reduced by a factor of three in hard scattering events (e.g. $t\bar{t}$ production) with little physics penalty. This simulation may eventually become the default simulation for all processes that do not require extremely accurate modeling of calorimeter response or electromagnetic physics.

ATLFAST-I has been developed for physics parameter space scans and studies that require very large statistics but do not require the level of detail contained in the full simulation [110, 111]. Truth objects are smeared by detector resolutions to provide physics objects similar to those of the reconstruction. Object four-vectors are output, without any detailed simulation of efficiencies and fakes. A factor of 1000 speed increase over full simulation is achieved with sufficient detail for many general studies.

ATLFAST-II is a fast simulation meant to provide large statistics to supplement full simulation studies. The aim is to try to simulate events as fast as possible while still being able to run the standard ATLAS reconstruction. ATLFAST-II is made up from two components: the Fast ATLAS Tracking Simulation (Fatras) for the inner detector and muon system simulation [112] and the Fast calorimeter Simulation (FastCaloSim) for the calorimeter simulation. Optionally, any subdetector can be simulated with GEANT4 to provide the higher level of accuracy without the same CPU time consumption as full simulation of the entire detector. An improvement over full simulation time of a factor of 10 is achieved with full GEANT4 inner detector and muon simulation and FastCaloSim, and a factor of 100 is achieved with Fatras and FastCaloSim.

7.1 Fast G4 simulation

The Fast G4 Simulation reduces CPU time consumption without sacrificing accuracy by speeding up the slowest parts of the full simulation. By treating (as described below) electromagnetic showers in the sampling portions of the calorimeters, a reduction in CPU time of a factor of three can be achieved even in hadronic events. Although the calorimetry dominates simulation time for the full simulation, after treatment of the electromagnetic showers the simulation time is evenly distributed throughout all subdetectors. One particular advantage of this fast simulation over the other varieties is that its output file matches identically the format of the output of the full GEANT4 simulation. The data can therefore be run through the identical tests and digitization software following simulation, and the standard ATLAS trigger and reconstruction can be run.

There are three treatments applied to electromagnetic showers. For very high energy (>10 GeV) electrons and positrons, a tuned shower parameterization is available. For medium energy (10 MeV to 1 GeV) electrons, positrons, and photons, libraries of pre-simulated showers can be applied during the event. For very low energy (<10 MeV) electrons and positrons, a single hit can be deposited to recreate detector response. Each one of these treatments can be turned on by the user in each compartment of the electromagnetic calori meter and the forward hadronic calorimeter. The energy ranges can be set for each method, compartment, and particle.

For electrons and positrons above a sufficiently high energy, around 10 GeV in the central calorimeters, the sampling calorimeter is sufficiently homogeneous to apply a shower parameterization. Small steps are taken in the direction of the original particle, depositing energy according to several tuned functions as it traverses the detector. The longitudinal profiles of showers are parameterized and normalized with an energy scale to approximate the sampling fraction in each subdetector. The radial profile changes as a function of depth in the shower and is normalized by the longitudinal profile. Energy is deposited in hits in order to mimic the full simulation. Fluctuations are introduced in three separate places, representing the random characteristics of shower length and shape, the sampling resolution of the calorimeter, and the geometric fluctuations in the energy collected.

Particles captured by the fast simulation in the appropriate energy range (typically <1 GeV) are replaced by a shower from a pre-simulated library, rotated and scaled to match the primary particle. Shower libraries are generated in bins of pseudorapidity and energy for electrons and photons. Only hits in the sensitive detectors are stored in order to save space on disk and in memory. The binning reproduces the fine structure in the calorimeters. The libraries are read into memory the first time they are requested by the simulation, ensuring minimal memory overhead. They consume about 200 MB when all are in use. Showers are randomly selected with linear weighting from the energy bin above or below the primary particle and from the pseudorapidity bin above or below the particle. The shower is then rotated to

Table 14 The default combination of strategies used in the Fast G4 Simulation for each calorimeter compartment

Calorimeter	Parameterization	Shower libraries	Killing
Electromagnetic Barrel	Not used	10–1000 MeV e^+e^- <10 MeV photons	<10 MeV e^+e^-
Electromagnetic Endcap	Not used	10–1000 MeV e^+e^- <10 MeV photons	<10 MeV e^+e^-
Electromagnetic Forward	>9 GeV e^+e^-	10–1000 MeV e^+e^- <10 MeV photons	<10 MeV e^+e^-
Hadronic Forward	>1.5 GeV e^+e^-	Not used	<10 MeV e^+e^-

match the primary particle's original direction, and the energy is scaled to match the primary particle's energy. For example, an electron at pseudorapidity of 2.37 and with an energy of 12 MeV might use a shower from the electron and positron library's 2.4 bin in pseudorapidity and 10 MeV bin in energy. The shower would then be moved and rotated to match the position and direction of the original particle, and its energy would be scaled up by 20%.

Electrons and positrons with energies below about 10 MeV typically deposit only one hit in the sensitive region of the calorimeter. These particles are removed when inside the regular sampling region of any of the calorimeters ("killed"), and a single hit is placed in the calorimeter. The position of the hit is determined by a random exponential number times the radiation length in the detector in order to approximate the particle's range. The energy of the hit is scaled to the response of the detector and smeared by its resolution.

The standard combination of strategies is shown in Table 14. The strategies used in a particular subdetector is optimized for maximum CPU time improvement with minimal complexity. The upper energy bound for shower libraries, 1 GeV, balances memory use with speed. Libraries at higher energies also may not correctly reproduce the tails of electromagnetic shower shape distributions as well as low-energy libraries do. The minimum energy for application of the parameterization model is based purely on CPU time. In most subdetectors, it is faster to allow GEANT4 to produce 1 GeV secondaries and apply shower libraries to those secondaries than it is to apply the shower parameterization. The same argument applies to high energy photons. They pair produce sufficiently quickly that treating them separately only adds complexity to the models. The speed of the parameterization is limited by random number generation and locating hits within the detector geometry.

7.2 ATLFAST-I

ATLFAST-I performs a fast simulation of the ATLAS detector, including object reconstruction, in order to produce high statistics samples of signal and background events. The lowest possible CPU time per event is achieved by replacing detailed detector simulation with parameterizations of the desired detector and reconstruction effects. The high speed of simulation in ATLFAST-I makes it possible to study channels where the statistics involved would otherwise be prohibitive. For example, the background to a $Z \to \tau^+\tau^-$ study from fake taus in di-jet events is expected to require $O(10^9)$ events in 100 fb^{-1} of data. Some searches also require many datasets to be simulated in order to scan across parameter space for the model being tested, such as SUSY.

ATLFAST-I is the least detailed simulation method. There is no realistic detector description, so studies of detector-based quantities, such as calorimeter sampling energies and track hit positions, are not possible. There is no simulation of reconstruction efficiency or misidentification rates, discussed later on, which means the presence of genuine physics objects are overestimated while fake objects are not modeled, with two exceptions. Because jet-flavor tagging efficiencies are applied, fake b-jets and taus are simulated. However, ATLFAST-I provides a useful method of making quick estimates of systematic uncertainties in early data analyses due to the simple process of re-parameterizing the detector and modeling reconstruction effects. The speed of operation enables datasets to be reproduced with different generator configurations, allowing quick estimates of systematic uncertainties arising from generators.

Common to the reconstruction of all objects in ATLFAST-I is that by default no reconstruction efficiencies are applied. These efficiencies can be taken from full simulation and accounted for by the user in the analysis. This applies to electrons, photons and jets as well as to ATLFAST-I tracks. It should be noted that tagging efficiency factors are implicitly taken into account in the tau- and b-tagging procedures. A system to apply a common set of efficiencies and misidentification rates at the analysis stage is in development. The misidentification rates will allow the modeling of fake objects as well.

ATLFAST-I takes input in HepMC format, enabling it to read the output of all ATLAS generators. Generator input is filtered to choose only particles that are useful in the current step. For example, only charged particles are considered in the tracking stage, and all particles are required to be a part of the final state.

The following sections describe steps taken in ATLFAST-I.

7.2.1 Tracks

Charged particle tracks from the generator with $p_T > 500$ MeV and with $|\eta| < 2.5$ are considered as reconstructed ATLFAST-I tracks, and five track parameters[14] are associated to them. These parameters are calculated from the true particle properties by applying parametrized resolution functions which account for the measurement precision, energy loss, and multiple scattering as well as for hadronic interactions in the inner detector material. The resolution functions are taken from fully simulated events. The non-Gaussian tails resulting from hadronic interactions are taken into account by applying a double-Gaussian correlated smearing to the track parameters of hadrons [110, 111]. No vertex smearing is applied. In ATLFAST-I, three types of charged particles are distinguished: hadrons, electrons and muons. Due to the relatively large energy loss from bremsstrahlung, high-p_T electrons are treated separately, and an additional energy loss correction is applied. It should be noted that while these tracks are used for specific studies in B physics, they are not used for lepton identification or b-tagging.

7.2.2 Track-based tau identification

Track-based tau identification is split into two distinct parts, namely reconstruction and identification of tau candidates. The reconstruction part applies a parameterized efficiency to the tracks to calculate the charged component in a tau candidate, while the neutral component is calculated directly from neutral particles in the generated event.

Once a sample of tau candidates has been reconstructed, the identification part is carried out by separating the sample into true and fake taus. True taus are defined as those matched to a hadronic decay in the truth record with $\Delta R \equiv \sqrt{\Delta\eta^2 + \Delta\phi^2} \leq 0.2$, whereas the remainder are considered fakes. Subsequently, a parametrization of the identification efficiency is applied based on the number of tracks.

7.2.3 Calorimetry

Stable charged particles from the event generators are propagated through the magnetic field along a simple helix. The primary vertex is assumed to be at the geometric origin. Using a helix model and assuming a perfectly homogeneous magnetic field inside the central tracking volume, the impact point on the calorimeter surface is calculated. To calculate this point, no interactions of the particle with the detector material (i.e. no multiple scattering, energy loss, or nuclear interactions) are taken into account. In particular, this implies that no energy loss due to bremsstrahlung for electrons and no pair production from photons in the inner detector media are simulated for the energy depositions in the calorimeters. The effects of these interactions are, however, implicitly taken into account by the application of appropriate resolution functions. For the calculation of track parameters, the four-momenta and the starting point of the particles (e.g. for stable decay products of long-lived particles) are taken from the generator information.

The energies of the electrons, photons, and hadrons are deposited in a calorimeter cell map. The response of the calorimeter is assumed to be unity and uniform over the full detector. No smearing (i.e. no resolution function) is applied. The energy of the particle is entirely deposited in the hit calorimeter cell, assuming a granularity of the calorimeter cell map of $\eta \times \phi = 0.1 \times 0.1$ up to $|\eta| < 3.2$ and $\eta \times \phi = 0.2 \times 0.2$ for $3.2 < |\eta| < 5.0$. Neither lateral nor longitudinal shower development is simulated. Therefore, the longitudinal fine structure of the calorimeters is not taken into account. There is also no separation between the electromagnetic and the hadronic calorimeter compartments.

Based on the map of deposited cell energies, cluster reconstruction is carried out using either SISCone [113] or FastKt algorithms via the FastJet libraries [114]. The default clustering routine is SISCone with a cone size of 0.4. The cluster transverse energy must pass a threshold, typically 5 GeV. The clusters may get re-classified as electrons, photons, taus or jets in one of the following steps. If they are associated to one of these objects, they are removed from the list of clusters.

7.2.4 Electrons and photons

For each true electron or photon, the reconstructed energy is obtained by smearing the true energy according to a resolution calculated by interpolating between resolutions measured in fully simulated events at precise values of η and energy. If, after smearing, the candidate electron or photon has transverse energy exceeding a threshold value, typically 5 GeV, and has $|\eta| < 2.5$, then it is recorded with the η and ϕ directions of the true particle.

Electrons and photons are matched to calorimeter clusters in (η, ϕ) space, with a maximum allowed separation of $\Delta R = 0.15$. If there is a matching cluster then it is removed from the list of clusters to be considered as jet candidates later on.

[14] The five parameters are: the azimuthal angle ϕ; longitudinal impact parameter, z_0, transverse impact parameter, $d_0 \equiv \sqrt{x^2 + y^2}$; polar angle in θ; and charge divided by momentum amplitude.

7.2.5 Muons

For each true muon with $p_T > 0.5$ GeV, the reconstructed momentum is calculated from the true muon momentum. A Gaussian resolution function which depends on p_T, η, and ϕ is applied. After smearing, muons with $p_T > 5$ GeV and with $|\eta| < 2.5$ are kept.

7.2.6 Isolation

In order to define isolated electrons, photons, and muons, the following criteria are applied: the difference in the unsmeared energy in a cone of $\Delta R = 0.2$ around the object direction and its smeared energy needs to be below 10 GeV. In addition, there should be no further clusters reconstructed with $\Delta R < 0.4$ around the object direction.

7.2.7 Jets

All clusters that have not been assigned to a true electron or photon are considered jets if their transverse energy exceeds 10 GeV. The jet energy is taken to be the cluster energy, after adding non-isolated muons within $\Delta R = 0.4$, and is smeared according to the jet energy resolution.

These functions do not account for pile-up, although there is a "high luminosity" mode available which adds a pile-up term to the resolution. The pile-up correction is constant with respect to jet transverse energy and is dependent on the size of the jet.

The jet direction is taken to be the cluster direction. Since the response function of the calorimeter is set to one, no jet calibration is needed to correct for the non-compensation of the calorimeter. However, an out-of-cone energy correction is needed. This correction is applied in a separate jet calibration step [110, 111].

7.2.8 Tagging

For each jet found, a label is attached to indicate whether the true jet originated from a light quark, b-quark, c-quark, or tau. This label is based on matching b or c partons or the visible decay products of hadronically-decaying taus at truth-level with $\Delta R < 0.3$ to a reconstructed jet. In the case of hadronically-decaying taus, the ratio between the true visible energy and the jet energy is also required to be larger than unity minus 2σ, where σ is the jet energy resolution as above.

The results of b- and tau-tagging are then simulated by applying identification efficiencies and fake rates to the labels. These efficiencies are determined from full simulation studies and are parameterized as a function of p_T and η.

7.2.9 Missing E_T

The missing transverse energy is calculated from all reconstructed objects: isolated electrons, photons, muons, taus, jets and non-isolated muons, and remaining calorimeter clusters not associated to jets. In addition, cells not associated to clusters are included in the missing E_T calculation. The cell energies are smeared by applying the jet resolution functions.

7.3 ATLFAST-II

ATLFAST-II directly simulates the input to the standard Athena reconstruction algorithms to mimic the full simulation. Unlike ATLFAST-I, which provides only momenta for the reconstructed objects, reconstructed ATLFAST-II output includes all the properties associated with a reconstructed object. In the case of Fatras these include the hits in the inner detector and muon system, and for FastCaloSim these include the energies in the calorimeter cells. Because the standard reconstruction is run, it is possible to work with a combination of full and ATLFAST-II simulated events without modifying any analysis code. Both Fatras and FastCaloSim run together with the event reconstruction. The simulation time is reduced by making use of the simplified detector description used for reconstruction [115]. By default, ATLFAST-II uses full simulation for the inner detector and muon system and FastCaloSim in the calorimetry. ATLFAST-IIF uses FastCaloSim in the calorimetry and Fatras in the inner detector and muon system.

As input, Fatras uses input in HepMC format, performs a smearing of the primary vertex position to represent the luminous region within ATLAS, and records truth information in a way similar to the full simulation. FastCaloSim uses the truth information of all interacting particles at the end of the inner detector volume as input to the calorimeter simulation. In order to simulate pile-up, generated events must be overlaid prior to detector simulation.

7.3.1 Fatras

ATLFAST-II with the fast track simulation engine Fatras (ATLFAST-IIF) reduces simulation time in the inner detector and muon system. Fatras is an ATLAS specific development and establishes a complete simulation within the track reconstruction framework. The reconstruction geometry is a simplified description of the full detector geometry, which keeps the same descriptive accuracy for sensitive detector parts, while approximating all other detector components as simplified layers that carry a high-granularity density map. This detector material description can be sufficient. A factor of 100 reduction in CPU time is obtained with only small physics performance degradation. The propagation of the

particles through the tracking detectors is carried out by the extrapolation engine [116] used in the offline track reconstruction applications.

The interactions of the particles with the simplified detector layers are simulated using several methods. Multiple Coulomb scattering is implemented as a Gaussian mixture model to account for tail effects from single large-angle scattering processes; ionization and radiative energy loss are simulated according to the Bethe-Bloch and Bethe-Heitler models; conversion of a photon into an electron and positron is carried out depending on the thickness of the traversed material; hadronic interactions of particles with the detector layers are simulated from a parametric model that has been obtained from GEANT4 simulation results. The decay of unstable particles is enhanced by a dedicated wrapper of the associated GEANT4 modules [112]. The calorimeter simulation of ATLFAST-IIF is typically FastCaloSim, and Fatras provides the input particle collection. Energy deposition for muons in the calorimeter layers is also recorded according to the material description of the reconstruction geometry and is further used for cluster simulation in the FastCaloSim application.

Fatras was first established as a validation tool for the newly deployed inner detector reconstruction sequence. It has already been used for noise studies in the Transition Radiation Tracker and first simulations for a potential future upgrade of the ATLAS inner detector. The validation of Fatras against the full simulation results to be used for first collisions data from LHC is ongoing. An extension of the fast track simulation within the reconstruction geometry has taken place that also allows the use of Fatras in the muon spectrometer. The particles being simulated at the end of the inner detector volume are filtered. Muons are transported through the calorimeter, and their deposited energy is stored as an input to the FastCaloSim module. The trajectories of the muons are then simulated in the muon spectrometer, and the hits within sensitive detector elements are recorded. Standard digitization is applied on top of the simulated hits to account for the detailed calibration that must be included for a comparison to data.

7.3.2 FastCaloSim

Instead of simulating the particle interactions with the detector material, the energy of single particle showers is deposited by FastCaloSim directly using parametrizations of the longitudinal and lateral energy profile. The distribution of active and inactive material in the calorimeter needs to be respected by the parametrization, so a fine binning of the parametrization in the particle energy and pseudorapidity is needed. Furthermore, the energy deposition depends strongly on the origin of the shower in the calorimeter, so all parametrizations are also binned versus the longitudinal depth of the shower center.

The parametrizations are based on a 30 million event sample of fully-simulated (i.e. simulated with GEANT4) single photons and charged pions in an energy range between 200 MeV and 500 GeV, evenly distributed in $|\eta| < 5.0$ and $-\pi < \phi < \pi$. All electron and photon showers are approximated by the photon parametrization and all hadronic showers are approximated by the charged pion parametrization. The simplified reconstruction geometry of the calorimeter is used with details at the level of the readout cells.

The parameterization of the longitudinal energy distribution is constructed from histograms of the total energy in all calorimeter layers, the longitudinal depth of the shower center, and the energy fraction in each layer for the fully-simulated single-particle events. The dominant correlations between fractional energy deposits in each calorimeter layer (i.e. those related to the longitudinal depth of the shower's origin) are accounted for in the parameterization binning. Gaussian correlations between fractional energy deposits in each calorimeter layer (i.e. those describing shower development) are stored in a correlation matrix and are applied to improve the parameterized energy distribution. During fast simulation, the parametrization closest in energy and pseudorapidity to the particle is taken, and then the total shower energy and the shower depth are chosen randomly from the stored histograms and rescaled to match the true particle energy. It was found that after rescaling no interpolation between parametrizations is necessary. Afterwards, the energy fractions in all calorimeter layers are generated randomly, taking into account the correlation matrix. The lateral energy distribution inside each calorimeter layer is simulated using a symmetric average radial shape function. The shape functions are extracted from fits to fully-simulated single-particle events and are constructed for bins of particle type, primary particle energy, position in η, and shower depth in the calorimeter. The asymmetry of shower shapes for particles entering the calorimeter at large incident angles is absorbed in a shape function describing a pseudorapidity-dependent asymmetry term. During simulation, the energy of a calorimeter cell is determined by the integral of the shape function over the cell surface area. Fluctuations derived from the intrinsic resolutions of each calorimeter are applied to the cell energy. The total energy of all cells in one calorimeter layer is normalized to the total energy in the layer making use of the longitudinal shower shape.

The histograms and shape functions needed as input for the parametrizations use about 200 MB of memory. Since no simulation of particle interactions is done, the dominant part of the simulation time is spent on the numerical integration of the lateral shape functions. Overall, the calorimeter simulation time for a single particle is a few microseconds, and a typical (e.g. $t\bar{t}$) event needs a few seconds.

The parameterization of FastCaloSim differs in several important ways from that of the Fast G4 Simulation.

Table 15 Simulation times per event, in kSI2K seconds, for single particles generated with $|\eta| < 3.0$ and with the same transverse momentum. All times are averaged over 500 events. ATLFAST-II uses full simulation for the inner detector and muon system and FastCaloSim in the calorimetry. ATLFAST-IIF uses FastCaloSim in the calorimetry and Fatras in the inner detector and muon system

Sample	Full Sim	Fast G4 Sim	ATLFAST-II	ATLFAST-IIF	ATLFAST-I
5 GeV $\mu^{\pm}$	0.879	0.899	1.28	0.633	0.011
50 GeV $\mu^{\pm}$	1.63	1.15	2.71	0.606	0.011
500 GeV $\mu^{\pm}$	12.0	10.4	11.8	0.615	0.011
1 GeV $e^{\pm}$	3.62	0.734	0.825	0.513	0.011
5 GeV $e^{\pm}$	17.8	1.64	1.00	0.542	0.011
50 GeV $e^{\pm}$	179	4.86	1.25	0.588	0.013
1 GeV $\pi^{\pm}$	2.40	1.48	0.701	0.515	0.011
5 GeV $\pi^{\pm}$	10.4	4.27	0.811	0.540	0.011
50 GeV $\pi^{\pm}$	94.7	30.3	1.04	0.569	0.011

Table 16 Simulation times per event, in kSI2K seconds, for the full simulation, Fast G4 simulation, ATLFAST-II, ATLFAST-IIF, and ATLFAST-I. ATLFAST-II uses full simulation for the inner detector and muon system and FastCaloSim in the calorimetry. ATLFAST-IIF uses FastCaloSim in the calorimetry and Fatras in the inner detector and muon system. All times are averaged over 250 events, except heavy ion times which were averaged over only 50 events. Because the memory required to reconstruct heavy ion events exceeds 3 GB and because FastCaloSim runs during the reconstruction step, the amount of time taken by FastCaloSim could not be measured in that sample. It was estimated as 10% of the full inner detector simulation time, consistent with the other hard scattering events

Sample	Full Sim	Fast G4 Sim	ATLFAST-II	ATLFAST-IIF	ATLFAST-I
Minimum bias	551	246.	31.2	2.13	0.029
$t\bar{t}$	1990	757	101.	7.41	0.097
Jets	2640	832	93.6	7.68	0.084
Photon and jets	2850	639	71.4	5.67	0.063
$W^{\pm} \to e^{\pm}\nu_e$	1150	447.	57.0	4.09	0.050
$W^{\pm} \to \mu^{\pm}\nu_{\mu}$	1030	438	55.1	4.13	0.047
Heavy ion	56,000	21,700	∼3050	203	5.56

FastCaloSim fills the readout geometry of ATLAS and applies a parameterization from the edge of the inner detector, whereas the Fast G4 Simulation places hits like those of GEANT4 into the full ATLAS detector geometry and is only applied in the sampling portion of the calorimeter (e.g. excluding the cryostats surrounding the calorimetry). As a result, the Fast G4 Simulation output can be run through the standard digitization software, whereas the FastCaloSim output is fed directly into the reconstruction.

7.4 Computing performance

Examples of simulation times in kSI2K seconds [117] for various types of events in the full and fast simulations are provided in Tables 15 and 16.[15] In single central ($|\eta| < 3$) electron events the simulation time is decreased by a factor of ten or more by the fast G4 simulation, and in hard scattering events the simulation time is decreased by a factor of 2–5. ATLFAST-II without Fatras decreases simulation time by a factor of 20–40, and ATLFAST-IIF decreases simulation time by a factor of 100. FastCaloSim accounts for about 10% of the total simulation time in ATLFAST-II and 60–70% of the total simulation time in ATLFAST-IIF. ATLFAST-I requires a relatively negligible amount of CPU time even for hard scattering events. ATLFAST-I, FastCaloSim, and Fatras run during the reconstruction step (see Sect. 2.1), but for these purposes the time consumed by their methods is included in "simulation time." Figure 8 shows the distribution of simulation times per event for full, Fast G4, and ATLFAST-II simulation of 250 $t\bar{t}$ events. The distributions are similar in shape.

In evaluating these CPU times, it is necessary to keep in mind the ad ditional steps required before analysis of the data can be performed. For both full and fast G4 simulation,

[15] Measurements were performed on Sun Fire X2200 M2 units with dual dual-core 2.6 GHz AMD Opteron 2218 processors. All times assume the jobs run on a single core, with the other cores loaded. Normalization was done using the peak specmark int 2000 rating, 1794. For the same system, the peak specmark floating point 2000 rating was 3338. The normalization follows the published results, rather than the WLCG formula in [118]. Details of cross-platform benchmarking can be found in [119].

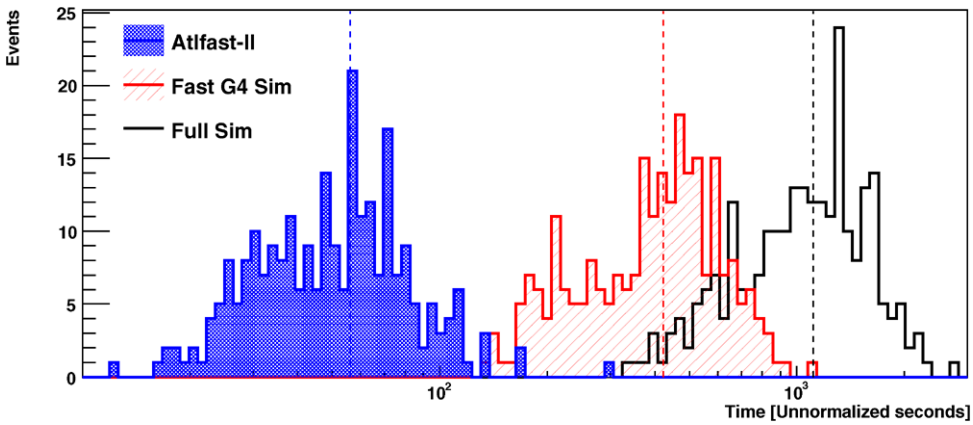


Fig. 8 Distributions of CPU time for 250 $t\bar{t}$ events in full, Fast G4, and ATLFAST-II simulations. *Vertical dotted lines* denote the averages of the distributions

the data must be digitized and reconstructed. For ATLFAST-II, the inner detector and muon system must be digitized[16] and reconstructed, but the calorimeter requires only reconstruction. For ATLFAST-IIF, only the muon system must be digitized before reconstruction is performed. The output of ATLFAST-I is in a format similar to that of the reconstruction and needs no further processing. The CPU time required for these additional steps is discussed in Sect. 8.2.

7.5 Physics performance

The fast simulations have been compared to full simulation in both low-level analyses with single particles entering the calorime ter and high-level analyses of detector observables with jets and active hard scattering events. The Fast G4 Simulation agrees to about 1–2% in jet energy scale after the standard calibration procedure and agrees to within 5% percent in electron identification efficiencies. Due to the simplifications in the calorimeter simulation, FastCaloSim differs at the 5% level from full simulation after reconstruction, especially in properties that are sensitive to the shape of hadronic showers. The jet energy scale differs by 1–2% after recalibration, and electron identification efficiency differs by about 5%. Since all particles are simulated using an average lateral shape function, visible effects like electromagnetic subshowers in charged pion showers are not described. These differences can be reduced by applying additional object-dependent correction functions after reconstruction. Fakes and calorimeter punch-through are not well modeled in ATLFAST-II and ATLFAST-IIF.

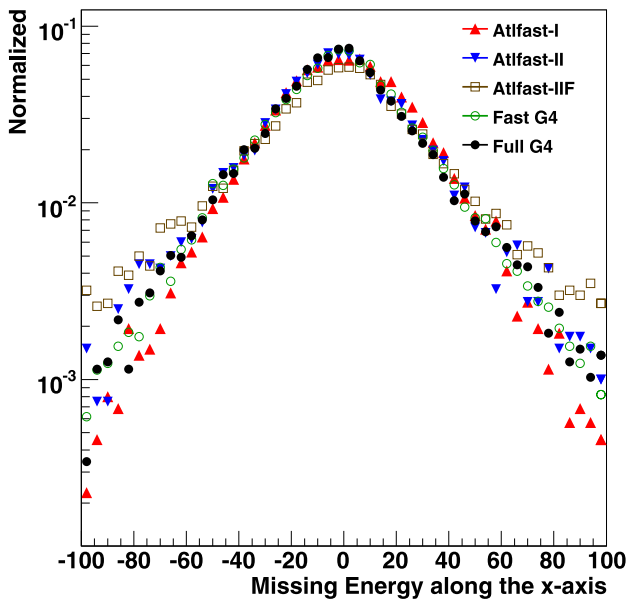
Figure 9 shows missing transverse energy along the x-axis for the full and fast simulations in di-jet events with a leading parton p_T between 560 and 1120 GeV, as well as jet p_T resolution as a function of η in $t\bar{t}$ events for jets with $20 < p_T^{\text{True}} < 40$ GeV. ATLFAST-II and the Fast G4 Simulation agree well with full simulation in missing transverse energy spectrum, even in the tails of the distribution. ATLFAST-I does not sufficiently populate the tails of the missing transverse energy distribution, and ATLFAST-IIF has too wide a distribution. ATLFAST-I, ATLFAST-IIF, and ATLFAST-II show 10–20% deviations from full simulation in jet transverse momentum resolution. These fast simulations mostly deal with the response of individual hadrons, which is not always sufficient to model the threshold effects common in jet and missing transverse energy measurements. Fast G4 simulation is consistent with full simulation through the entire range in pseudorapidity.

Figure 10 shows reconstructed muon p_T resolution as a function of muon p_T in $Z \to \mu^+\mu^-$ events. Muons reconstructed using the muon spectrometer alone and those reconstructed using both the muon spectrometer ("standalone") and inner detector ("combined") are shown. Only one type of muon is provided by ATLFAST-I, so it is only included in the combined reconstruction plot. The difference between full simulation and ATLFAST-I is due to an older parameterization. In the cases of ATLFAST-II and the Fast G4 simulation, muon spectrometer simulation is done by GEANT4 and should, therefore, be identical to full simulation. The fast simulations show generally good agreement over the entire range of p_T. ATLFAST-IIF has standalone muon resolution that is 10% better than full simulation in some bins of p_T, but since the muon system simulation of ATLFAST-IIF is still under development, the agreement is expected to improve. It is generally left to the physics groups to evaluate the fast simulations with their analyses and determine which is acceptable.

8 Validation

Validation of the ATLAS simulation chain is done in two distinct phases. First, the software performance must be assessed. Then, the physics performance must be tested and

[16] The inner detector and muon system together require about 2/3 of the total digitization time.



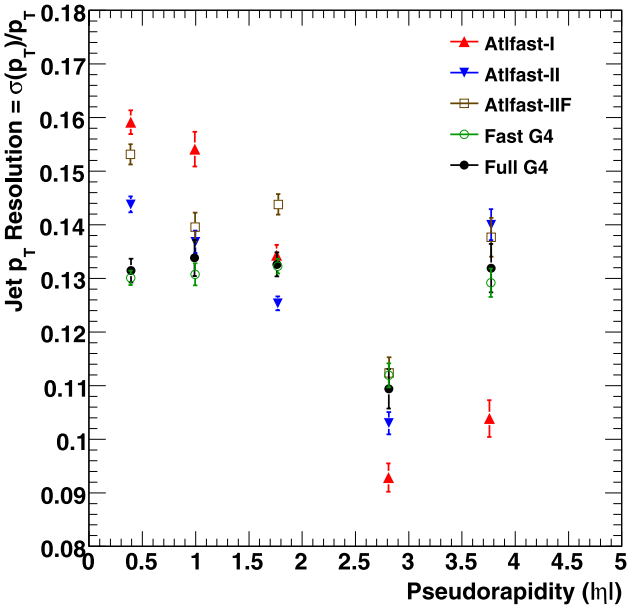
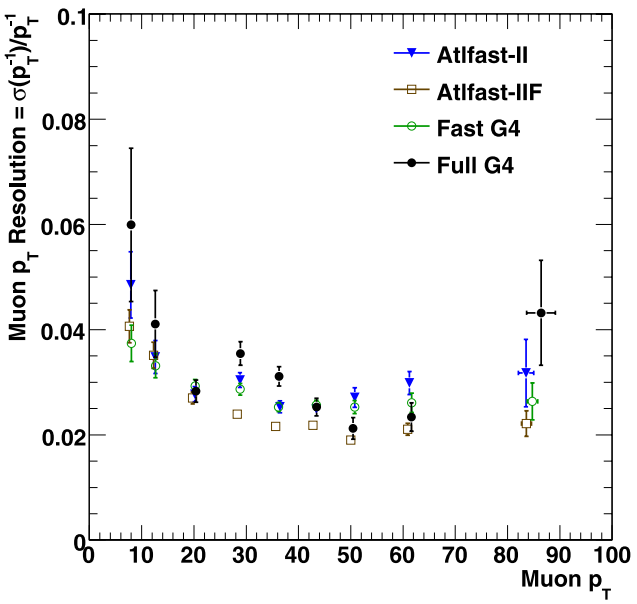


Fig. 9 *Left*, fast simulations (*color*) and full simulation (*black*) comparison of missing transverse energy along the x-axis in di-jet events with a leading parton p_T between 560 and 1120 GeV. *Right*, a comparison of jet p_T resolution as a function of pseudo rapidity in $t\bar{t}$ events for jets with $20 < p_T^{\text{True}} < 40$ GeV



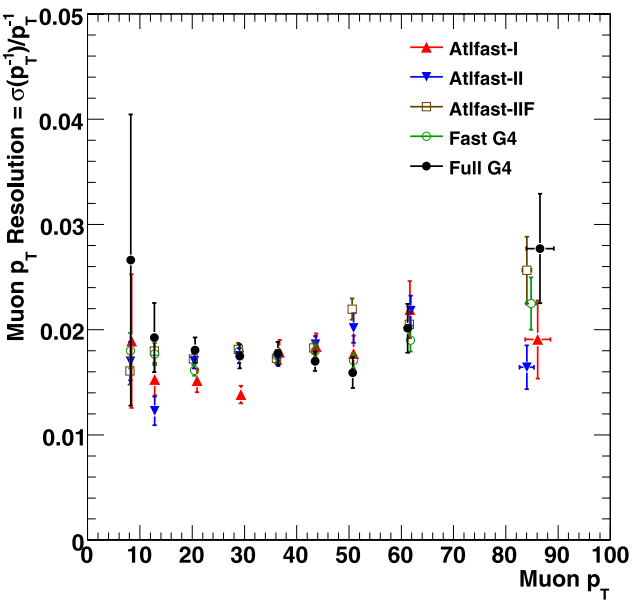


Fig. 10 Fast simulations (*color*) and full simulation (*black*) comparison of reconstructed muon p_T resolution as a function of muon p_T for central ($|\eta| < 1.2$) muons in $Z \to \mu^+\mu^-$ events for muons reconstructed using only the muon spectrometer (*left*) and using both the inner detector and spectrometer (*right*). ATLFAST-I only provides one type of muon, which is included in the right plot

compared to available data. The first step includes testing robustness, testing software performance, and testing basic functionality. The second step includes comparison to test beam, cosmic data, and physics results obtained from previous simulation productions. In this section the infrastructure for each stage of validation is described.

ATLAS has a fresh software build every night for coordinating software development. Each nightly build is run through a rigorous test cycle, and as a release deadline approaches the test results are increasingly scrutinized to evaluate stability and performance. Thanks to the evaluation prior to release, generally only rare bugs appear in pro-

duction for the first time. The automatic testing infrastructure also allows evaluation of many different versions of the Athena software. Separate bug-fixing and development branches are employed, for example, and significant interface changes or low-level code migrations take place in separate branches until they are sufficiently stable to be merged into the main branch. Each version of the software comes in several flavors for different system architectures, operating systems, compilers, and so on. The simulation production in 2008 used 32-bit builds with gcc 3.4.6 [120] on CERN's Scientific Linux 4 [121]. External dependencies include CLHEP 1.9.3.1 and WLCG 54G.

A web portal has been constructed, using the Savannah bug tracking software [122], for monitoring problems with the various facets of the simulation software. Bugs can be reported and tracked as they are diagnosed and solutions are found, and new features can be requested.

8.1 Automated testing

The software performance of the simulation is monitored in three types of automated tests: ATLAS Testing Nightly (ATN) tests, Run Time Tester (RTT) tests, and Full Chain Tests (FCT). ATN tests are run every night on every software build and are basic functionality tests. RTT tests are run on a subset of builds and include 50 simulation tests to ensure functionality and, in some cases, consistent results. FCT tests are run on only a few builds each night and test the entire software chain in a production-like environment. Releases are required to pass a minimal number of milestones before being declared ready for production. For details about the ATN and RTT, refer to [9].

FCT tests are run daily on a small set of jobs. The aim of the FCT is to verify the readiness of a production cache release candidate for Grid production. The FCT runs jobs that test the functionalities of generation, the different flavors of fast and full simulation, digitization, bytestream conversion, and reconstruction of Standard Model processes, black hole production, and heavy ion collisions. In the case of Standard Model processes, a full chain[17] of jobs are run per release, with hard scattering events that stress the software. If successful, the output from each day's run is saved for use in the next step of the test the following day (e.g. Monday's generation provides input for Tuesday's simulation, which provides input for Wednesday's digitization). The typical number of events processed (50) is limited by the CPU requirements for the full simulation.

As part of the FCT, 1000 events that were simulated with an old validated release are reconstructed. This long test allows better evaluation of the reconstruction's stability. Moreover, the relatively large sample is used to make a preliminary check on the quality of the reconstruction for final state objects (jets, electrons, muons, etc.). All the other tests only check for the success or failure of the job, the number of events in the output file, and unknown error messages in the log file. If any of these checks fail, the release candidate is rejected, and an additional iteration of bug fixing is undertaken. Only once a release succeeds in all FCT tests is it distributed to the Grid.

8.2 Computing performance benchmarking

Event generation jobs are typically fast enough that not a large effort is made to test their software performance. In general, the jobs take a tens of milliseconds per event. Generation with PYTHIA or HERWIG requires about 450 MB of memory, and generation with Hijing requires about 170 MB of memory. The files produced in generation jobs are tens of kB per event (e.g. 40 kB/event for $t\bar{t}$ events).

CPU time, memory consumption, and output file size for the simulation are tested in each stable release using a variety of physics processes. Single muons, electrons, and charged pions are used, as well as di-jets in bins of leading parton p_T, a supersymmetric benchmark point,[18] minimum bias, Higgs boson decaying to four leptons, $Z \to e^+e^-$, $Z \to \mu^+\mu^-$, and $Z \to \tau^+\tau^-$ events. The same input events are used every time to ensure fair comparison of simulation results independent of changes to the event generation.

Simulation of the full detector requires typically ∼750 MB of memory (i.e. VSIZE) and includes loading of almost 400 libraries into memory. Memory consumed during simulation is also broken down into its three key components by taking snapshots of memory use during initialization: GeoModel, the ATLAS-side detector geometry, typically about ∼100 MB; G4Atlas, the purely GEANT4 component of the memory, typically about ∼300 MB; and load modules, the remaining algorithms and services loaded during the job, typically about ∼300 MB. Significant changes in any of these can indicate the proper source of a change in memory. The memory requirement is independent of the number of events in the job and varies by only a few percent for different physics processes. Although up to 2 GB of memory may be reserved for a Grid job, by keeping the memory requirements of a typical simulation job under 1 GB, more machines can be used. The memory required to build each piece of the detector can be found in Table 4. Of major concern is any increase in memory ("leaks") during the event loop once all libraries have been loaded and setup is complete. Some increase due to caching is expected

[17] A single chain of jobs runs all steps from event generation through reconstruction, sequentially, using the output of one step as the input of the next.

[18] ATLAS mSUGRA benchmark point SU3: $m_0 = 100$ GeV, $m_{1/2} = 300$ GeV, $A_0 = -300$, $\tan\beta = 6$, and $\mu > 0$.

Table 17 Generation, simulation, digitization, and reconstruction times per event, in kSI2K seconds. Generation times are averaged over 5000 events, except generation of heavy ion events, which were averaged over 250 events. Simulation, digitization, and reconstruction times are averaged over 250 events, except simulation of heavy ion events, which were averaged over 50 events. The heavy ion event simulation time is for events with a random impact parameter. Central collisions require on average 3.4 times longer to simulate. Reconstruction times should be taken as indicative of the order of magnitude, rather than as a precise measurement. Based on a previous release, heavy ion collision reconstruction is estimated to take ∼ 10 times longer than $t\bar{t}$ event reconstruction

Sample	Generation	Simulation	Digitization	Reconstruction
Minimum bias	0.0267	551	19.6	8.06
$t\bar{t}$ Production	0.226	1990	29.1	47.4
Jets	0.0457	2640	29.2	78.4
Photon and jets	0.0431	2850	25.3	44.7
$W^{\pm} \to e^{\pm}\nu_e$	0.0788	1150	23.5	8.07
$W^{\pm} \to \mu^{\pm}\nu_{\mu}$	0.0768	1030	23.1	13.6
Heavy ion	2.08	56,000	267	–

Table 18 Digitization computing resources for 50 $t\bar{t}$ events as they scale with luminosity. CPU times are normalized to the time required by a no pile-up job. Cavern background was overlaid during these jobs with a safety factor of one. Beam gas and beam halo were ignored

Resource	No Pile-up	10^{33} cm^{-2} s^{-1}	3.5×10^{33} cm^{-2} s^{-1}	10^{34} cm^{-2} s^{-1}
CPU time factor	1.0	2.3	5.8	160
Memory leak [kB/event]	10	270	800	2,100
Virtual memory [MB]	770	1,000	1,300	2,000
Allocated memory [MB/event]	12	21	40	985

during the processing of the first few events. However, if the memory required by the application continues to grow beyond the system limits, memory corruption and memory pressure can result in serious problems. The memory required by the ATLAS simulation has been found to increase by less than 0.25 MB per event under normal circumstances. The increases are not steady, but come in large (∼10 MB) and sporadic jumps. The source of these increases is not fully understood, but a 50 event simulation job still consumes well under 1 GB of memory.

The CPU time consumed by generation, full simulation, digitization, and reconstruction for various types of events is shown in Tables 15 and 17 and is typically several minutes per hard scattering event. All times are normalized to kSI2K seconds [117]. For the purposes of testing, logfile output was suppressed and no output files (e.g. hit or RDO files) were created. CPU time is also measured as a function of other simulation input parameters prior to significant changes, for example using different physics lists. For these runs, output files were disabled; in simulating $t\bar{t}$ events the time per event is increased by ∼0.5% when file writing is enabled. The hard scattering events shown in Table 17 were generated with a 14 TeV center of mass energy; for 10 TeV center of mass energy the simulation time is reduced by 17% for $t\bar{t}$ events. The distribution of CPU time for simulation of 250 $t\bar{t}$ events is shown in Fig. 8.

For the samples in Tables 16 and 17, event generation of W production, minimum bias interactions, di-jet events, and photon and jet events was done using PYTHIA. $t\bar{t}$ production was done using MC@NLO for the hard scattering and Herwig for hadronization and showering. In this case, the generation time includes only the time required by Herwig. Heavy ion event production was done using Hijing.

Digitization jobs are generally fast, but memory consumption can be a serious concern during jobs with many overlaid events. Table 18 shows how resource consumption during digitization of 50 $t\bar{t}$ events scales with pile-up luminosity.[19] The memory required for digitizing with a luminosity of 10^{34} cm^{-2} s^{-1} is sufficiently large that the memory limit of the testing machine was reached, and, therefore, swapping resulted in a significant increase in CPU time. The allocated memory per event is provided for some benchmark of the change in memory over the course of a single event.

8.3 Physics validation

Once a new release is distributed to the Grid sites, a set of several physics samples is produced. Typically, a "valida-

[19]The ATLAS software is under a continuous process of improvement, with improving performance in terms of calculation speed and memory profile, and problems such as memory leaks being identified and eliminated.

tion sample" includes 10,000 events for each process, a total of 110,000 single particle events and 250,000 hard scattering events. This standard validation sample includes single muons, pions, and electrons, Standard Model processes ($t\bar{t}$ production, vector boson production, B-physics), and exotic processes (e.g. supersymmetric events and black hole production). The composition of the validation sample has been chosen to test all aspects of the event reconstruction.

The running of the validation sample on the Grid usually exposes rare software problems in the release. It is unlikely that software bugs that appear with a frequency much lower than 1/1000 events are caught by the automatic validation procedure (1000 events is the size of the "long" jobs of the FCT, described in Sect. 8.1). This first round of production provides a feedback mechanism for the developers, who produce bug fixes before the next production cycle.

The last step before using a release for production is physics validation. A dedicated group of experts, including representatives from every detector performance (e.g. tracking, b-tagging, and jet reconstruction groups) and physics group (e.g. Standard Model, supersymmetry, and exotics search groups) in ATLAS, runs physics analyses on the validation samples. Their task is to verify the quality of the single object reconstruction (e.g. jets, electrons, and muons) and the results of more complex physics analyses (e.g. mass reconstruction in $Z \to \mu^+\mu^-$, $Z \to e^+e^-$, and $t\bar{t}$ events). The relatively large validation samples may expose minor problems that could not be found with lower statistics, for example a shift of a few percent in the reconstructed energy. In order to properly validate each version of the software, the results from each release are typically compared to those of previous validated releases. The software must, therefore, maintain backwards-compatibility in order to allow fair comparisons. Shifts in file format are carefully coordinated, and maintenance of the old format is continued for as long as necessary to ensure result consistency. The physics validation procedure is also used for checking major changes in the fast and full simulation (detector description, change in the simulation parameters, etc.).

The GEANT4 simulation has also been validated in a physics sense with all available detector data. Combined test beam studies have proven invaluable in understanding the performance of each of the subsystems, and the standalone test beam analyses have provided crucial input towards the optimization of the simulation and choice of parameters [123–126]. In 2008, a significant sample of cosmic ray data was collected with multiple subdetectors. The data have provided an important test of the simulation [127].

Although the detector simulation relies heavily on GEANT4, a significant effort was put into comparing tile calorimeter test stand response with the FLUKA simulation package [128]. For this comparison, the test stand geometry was translated into the FLUKA geometry format, and the output from the FLUKA simulation was translated back into a format comparable to that of GEANT4. It was eventually concluded that little would be gained by attempting a transition to FLUKA that could not already by gained by modifications to parameters and a different choice of physics models within GEANT4. FLUKA has also been used to study neutron flux and radiation levels throughout the detector [104], but many of these studies are being updated in GEANT4 with the high-precision neutron physics list (see Sect. 5.4).

Extensive efforts are underway to compare simulated data to real data and validate the output from each detector. Thanks to the multiple detector descriptions, several analyses have already been prepared and tested to find discrepancies between the detector description of the simulation and that of the as-built detector. For example, subdetectors can be "weighed" in the simulation to ensure that the amount of material is within a few percent of the constructed detector. Although the agreement with first high-energy collision data is not expected to be perfect, a great deal of experience has been gained. The effects of modifications to GEANT4 parameters have also been studied in some detail, so that differences between data and simulation might be remedied rapidly.

Digitization algorithms have been tuned against laboratory test results, test beam data, and, where possible, cosmic ray data taken during the detector commissioning. The studies continue with the data.

9 Summary and conclusions

We have presented the status of the ATLAS simulation project, including all steps from event generation to digitization. A robust and flexible framework is required to cope with the demands of complex detector descriptions and physics models. The software project has been prepared for data since late 2008 and is ready for data.

A variety of event generators are available to provide the user with a complete set of tools for testing new physics models. The simulation is highly configurable to ensure maximal flexibility in the face of the uncertain challenges approaching. The detector description itself, conditions of the detector, and many parameters used in the simulation can be modified at run time. The digitization is also made configurable to cope with uncertainty in machine performance, detector conditions, and cavern conditions. Three varieties of fast simulation have been made available to ease the difficulties caused by the time consumption of the full detector simulation. They each complement the full simulation.

Generation, simulation, and digitization tasks are running continually on the Grid. The validation program has produced a high quality simulation sample for the ATLAS experiment data.

Acknowledgements We are greatly indebted to all CERN's departments and to the LHC project for their immense efforts not only in building the LHC, but also for their direct contributions to the construction and installation of the ATLAS detector and its infrastructure. We acknowledge equally warmly all our technical colleagues in the collaborating Institutions without whom the ATLAS detector could not have been built. Furthermore we are grateful to all the funding agencies which supported generously the construction and the commissioning of the ATLAS detector and also provided the computing infrastructure.

The ATLAS detector design and construction has taken about fifteen years, and our thoughts are with all our colleagues who sadly could not see its final realisation.

We acknowledge the support of ANPCyT, Argentina; Yerevan Physics Institute, Armenia; ARC and DEST, Australia; Bundesministerium für Wissenschaft und Forschung, Austria; National Academy of Sciences of Azerbaijan; State Committee on Science & Technologies of the Republic of Belarus; CNPq and FINEP, Brazil; NSERC, NRC, and CFI, Canada; CERN; CONICYT, Chile; NSFC, China; COLCIENCIAS, Colombia; Ministry of Education, Youth and Sports of the Czech Republic, Ministry of Industry and Trade of the Czech Republic, and Committee for Collaboration of the Czech Republic with CERN; Danish Natural Science Research Council and the Lundbeck Foundation; European Commission, through the ARTEMIS Research Training Network; IN2P3-CNRS and Dapnia-CEA, France; Georgian Academy of Sciences; BMBF, DFG, HGF and MPG, Germany; Ministry of Education and Religion, through the EPEAEK program PYTHAGORAS II and GSRT, Greece; ISF, MINERVA, GIF, DIP, and Benoziyo Center, Israel; INFN, Italy; MEXT, Japan; CNRST, Morocco; FOM and NWO, Netherlands; The Research Council of Norway; Ministry of Science and Higher Education, Poland; GRICES and FCT, Portugal; Ministry of Education and Research, Romania; Ministry of Education and Science of the Russian Federation, Russian Federal Agency of Science and Innovations, and Russian Federal Agency of Atomic Energy; JINR; Ministry of Science, Serbia; Department of International Science and Technology Cooperation, Ministry of Education of the Slovak Republic; Slovenian Research Agency, Ministry of Higher Education, Science and Technology, Slovenia; Ministerio de Educación y Ciencia, Spain; The Swedish Research Council, The Knut and Alice Wallenberg Foundation, Sweden; State Secretariat for Education and Science, Swiss National Science Foundation, and Cantons of Bern and Geneva, Switzerland; National Science Council, Taiwan; TAEK, Turkey; The Science and Technology Facilities Council and The Leverhulme Trust, UK; DOE and NSF, USA; Gordon and Betty Moore Foundation, USA.

The authors would like to thank the GEANT4 developers for their work and support.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. G. Aad et al., The ATLAS experiment at the CERN Large Hadron Collider. J. Instrum. **3**, S08003 (2008)
2. L. Evans, The Large Hadron Collider. New J. Phys. **9**, 335 (2007)
3. ATLAS Collaboration, ATLAS computing technical design report. ATLAS-TDR-017, CERN-LHCC-2005-022 (2005). See also http://atlas-computing.web.cern.ch/atlas-computing/packages/athenaCore/athenaCore.php
4. S. Agostinelli et al., Geant4—a simulation toolkit. Nucl. Instrum. Methods Phys. Res. A **506**, 250–303 (2003)
5. J. Allison et al., Geant4 developments and applications. IEEE Trans. Nucl. Sci. **53**, 270–278 (2006)
6. M. Cattaneo et al., Status of the GAUDI event-processing framework, in *Computing in High Energy and Nuclear Physics 2001 Conference (CHEP2001)*, IHEP, Beijing, China, September 3–7, 2001, ed. by H.S. Chen, (e-article, 2001)
7. G. Barrand et al., GAUDI—a software architecture and framework for building LHCb data processing applications, in *Computing in High Energy and Nuclear Physics 2000 Conference (CHEP2000)*, ed. by M. Mazzucato, Pavia, Italy, February 7–11, 2000 (Padua, INFN, 2000)
8. L. Lonnblad, CLHEP: a project for designing a C++ class library for high-energy physics. Comput. Phys. Commun. **84**, 307–316 (1994)
9. E. Obreshkov et al., Organization and management of ATLAS offline software releases. Nucl. Instrum. Methods A **584**, 244–251 (2008)
10. CLOC—Count Lines of Code, http://cloc.sourceforge.net, 2006
11. D. Dullmann et al., POOL development status and plans, in *Computing in High Energy and Nuclear Physics 2004 (CHEP2004)*, ed. by A. Aimar, J. Harvey, N. Knoors, Interlaken, Switzerland, September 27–October 1, 2004 (CERN, Geneva, 2004)
12. R. Chytracek et al., POOL development status and production experience. IEEE Trans. Nucl. Sci. **52**, 2827–2831 (2005)
13. D. Duellmann, The LCG POOL project, general overview and project structure, in *Computing in High Energy and Nuclear Physics 2003 (CHEP2003)*, La Jolla, CA, March 24–28, 2003. (CERN, Geneva, 2003). physics/0306129
14. M. Dobbs, J.B. Hansen, The HepMC C++ Monte Carlo event record for high energy physics. Comput. Phys. Commun. **134**, 41–46 (2001)
15. M. Grothe et al., Architecture of the ATLAS high level trigger event selection software. Nucl. Instrum. Methods Phys. Res. A **518**, 537–541 (2003). In Proceedings of the 2003 Conference for Computing in High Energy and Nuclear Physics, La Jolla, CA, USA, 24–28 March 2003
16. V. Boisvert et al., Final report of the ATLAS reconstruction task force. ATL-SOFT-2003-010 (2003)
17. I. Bird et al. (eds.), LHC computing Grid. Technical design report, CERN-LHCC-2005-024 (2005)
18. P. Nevski, Large scale data movement on the GRID, (2006)
19. M. Branco et al., Managing ATLAS data on a petabyte-scale with DQ2. J. Phys. Conf. Ser. **119**, 062017 (2007). In Proceedings of Computing in High Energy and Nuclear Physics 2007 Conference (CHEP2007), Victoria, BC, Canada, September 2–7, 2007, ed. R. Sobie, R. Tafirout, J. Thomson
20. T. Sjöstrand, S. Mrenna, P. Skands, PYTHIA 6.4 physics and manual. J. High Energy Phys. **05**, 026 (2006). hep-ph/0603175
21. M. Smizanska, S.P. Baranov, J. Hrivnac, E. Kneringer, Overview of Monte Carlo simulations for ATLAS B-physics in the period 1996–1999. ATL-PHYS-2000-025 (2000)
22. C. Anastopoulos et al., Physics analysis tools for beauty physics in ATLAS. J. Phys. Conf. Ser. **119**, 032003 (2007). In Proceedings of Computing in High Energy and Nuclear Physics 2007 Conference (CHEP2007), Victoria, BC, Canada, September 2–7, 2007, ed. R. Sobie, R. Tafirout, J. Thomson
23. G. Marchesini, B.R. Webber, G. Abbiendi, I.G. Knowles, M.H. Seymour, L. Stanco, HERWIG: a Monte Carlo event generator for simulating hadron emission reactions with interfering gluons. Version 5.1, April 1991. Comput. Phys. Commun. **67**, 465 (1992)
24. G. Corcella et al., HERWIG 6: an event generator for hadron emission reactions with interfering gluons (including supersymmetric processes). J. High Energy Phys. **01**, 010 (2001). hep-ph/0011363
25. G. Corcella et al., HERWIG 6.5 release note. CERN-TH/2002-270 (2005). hep-ph/0210213v2
26. T. Gleisberg et al., SHERPA 1.alpha, a proof-of-concept version. J. High Energy Phys. **02**, 056 (2004). hep-ph/0311263

27. M. Gyulassy, X.N. Wang, HIJING 1.0: a Monte Carlo program for parton and particle production in high-energy hadronic and nuclear collisions. Comput. Phys. Commun. **83**, 307 (1994). nucl-th/9502021
28. M.L. Mangano et al., ALPGEN, a generator for hard multiparton processes in hadronic collisions. J. High Energy Phys. **07**, 001 (2003). hep-ph/0206293
29. S. Frixione, P. Nason, B.R. Webber, Matching NLO QCD and parton showers in heavy flavour production. J. High Energy Phys. **08**, 007 (2003). hep-ph/0305252
30. B.P. Kersevan, E. Richter-Was, The Monte Carlo event generator AcerMC version 2.0 with interfaces to PYTHIA 6.2 and HERWIG 6.5 (2004). hep-ph/0405247
31. S. Jadach, J.H. Kuhn, Z. Was, TAUOLA: a Library of Monte Carlo programs to simulate decays of polarized tau leptons. Comput. Phys. Commun. **64**, 275–299 (1990)
32. E. Barberio, B. van Eijk, Z. Was, PHOTOS: a universal Monte Carlo for QED radiative corrections in decays. Comput. Phys. Commun. **66**, 115–128 (1991)
33. D.J. Lange, The EvtGen particle decay simulation package. Nucl. Instrum. Methods A **462**, 152–155 (2001)
34. F.E. Paige, S.D. Protopopescu, H. Baer, X. Tata, ISAJET 7.69: a Monte Carlo event generator for p p, anti-p p, and e+ e- reactions (2003). hep-ph/0312045
35. T. Sjöstrand, S. Mrenna, P. Skands, A brief introduction to PYTHIA 8.1. Comput. Phys. Commun. **178**, 852–867 (2008). 0710.3820 [hep-ph]
36. M. Bahr et al., Herwig++ physics and manual. Eur. Phys. J. C **58**, 639–707 (2008). 0803.0883 [hep-ph]
37. T. Stelzer, W.F. Long, Automatic generation of tree level helicity amplitudes. Comput. Phys. Commun. **81**, 357–371 (1994). hep-ph/9401258
38. C.M. Harris, P. Richardson, B.R. Webber, CHARYBDIS: a black hole event generator. J. High Energy Phys. **08**, 033 (2003). hep-ph/0307305
39. E. Boos et al. (CompHep Collaboration), CompHEP 4.4: Automatic computations from Lagrangians to events. Nucl. Instrum. Methods A **534**, 250 (2004). hep-ph/0403113
40. A. Pukhov et al., CompHEP—a package for evaluation of Feynman diagrams and integration over multi-particle phase space. User's manual for version 3.3. INP MSU report 98-41/542 (1999). hep-ph/9908288. See also http://comphep.sinp.msu.ru
41. T. Sjöstrand et al., in *Z Physics at LEP 1*, vol. 3 (Geneva, 1989), p. 143
42. A. Moraes, C. Buttar, I. Dawson, Prediction for minimum bias and the underlying event at LHC energies. Eur. Phys. J. C **50**, 435–466 (2007)
43. A. Moraes, C. Buttar, I. Dawson, Comparison of predictions for minimum bias event generators and consequences for ATLAS radiation background. ATL-PHYS-2003-020 (2002)
44. V.M. Abazov et al. (DZero Collaboration), Search for large extra dimensions in the monojet + MET channel with the DZero detector. Phys. Rev. Lett. **90**, 251802 (2003). hep-ex/0302014
45. T. Aaltonen et al. (CDF Collaboration), Measurement of the inclusive jet cross section at the Fermilab Tevatron p anti-p collider using a cone-based jet algorithm. Phys. Rev. D **78**, 052006 (2008)
46. R. Field, R.C. Group (CDF Collaboration), PYTHIA Tune A, HERWIG, and JIMMY in Run 2 at CDF, CDF-ANAL-CDF-PUBLIC-7822 (2005). hep-ph/0510198
47. E. Norrbin, T. Sjostrand, Production and hadronization of heavy quarks. Eur. Phys. J. C **17**, 137–161 (2000). hep-ph/0005110
48. A. Moraes, in *Proceedings of the First International Workshop on Multiple Partonic Interactions at the LHC (MPI@LHC08)*, Perugia, Italy, October 27–31, 2008, ed. by P. Bartalini, L. Fano (DESY-PROC, 2008), to appear
49. A.A. Affolder et al. (CDF Collaboration), Charged jet evolution and the underlying event in $p\bar{p}$ collisions at 1.8 TeV. Phys. Rev. D **65**, 092002 (2002)
50. D.E. Acosta et al. (CDF Collaboration), The underlying event in hard interactions at the Tevatron $\bar{p}p$ collider. Phys. Rev. D **70**, 072002 (2004) hep-ex/0404004
51. J.M. Butterworth, J.R. Forshaw, M.H. Seymour, Z. Phys. C **72**, 637–646 (1996)
52. S. Catani, F. Krauss, R. Kuhn, B.R. Webber, QCD matrix elements + parton showers. J. High Energy Phys. **11**, 63 (2001). hep-ph/0109231
53. M. Boonekamp et al. Cosmic ray, beam-halo and beam-gas rate studies for ATLAS commissioning. ATL-GEN-2004-001 (2004)
54. A. Dar, Atmospheric neutrinos, astrophysical neutrons, and proton-decay experiments. Phys. Rev. Lett. **51**, 227 (1983)
55. E. Boos et al., Generaic user process interface for event generators, in *Proceedings of Les Houches 2001: Physics at TeV Colliders*, Les Houches, France, May 21–June 1, 2001, ed. by W. Giele et al. (2001). hep-ph/0109068v1, The QCD/SM Working Group Report
56. J. Alwall et al., A standard format for Les Houches event files. Comput. Phys. Commun. **176**, 300–304 (2007). hep-ph/0609017
57. S. Frixione, P. Nason, C. Oleari, Matching NLO QCD computations with parton shower simulations: the POWHEG method. J. High Energy Phys. **11**, 070 (2007). 0709.2092 [hep-ph]
58. P. Skands et al., SUSY Les Houches accord: interfacing SUSY spectrum calculators, decay packages, and event generators. J. High Energy Phys. **0407**, 36 (2004). hep-ph/0311123
59. B.C. Allanach et al., SUSY Les Houches Accord 2. Comput. Phys. Commun. **180**, 8–25 (2009)
60. I. Borozan, M.H. Seymour, An eikonal model for multiparticle production in hadron hadron interactions. J. High Energy Phys. **09**, 015 (2002). hep-ph/0207283
61. D. Bourilkov, R.C. Group, M.R. Whalley, LHAPDF: PDF use from the Tevatron to the LHC, 2006. hep-ph/0605240
62. H. Plothow-Besch, PDFLIB: a library of all available parton density functions of the nucleon, the pion and the photon and the corresponding alpha-s calculations. Comput. Phys. Commun. **75**, 396–416 (1993)
63. W.K. Tung et al., Global QCD Analysis and Collider Phenomenology–CTEQ, in *Proceedings of the DIS2007 Workshop*, Munich, Germany, April 2007. 0707.0275 [hep-ph]
64. S. Chekanov et al. (ZEUS Collaboration), An NLO QCD analysis of inclusive cross-section and jet-production data from the ZEUS experiment. Eur. Phys. J. C **42**, 1–16 (2005). hep-ph/0503274
65. Albrow M. et al. (TeV4LHC QCD Working Group), Tevatron-for-LHC report of the QCD working group. FERMILAB-Conf-06-359 (2006). hep-ph/0610012
66. P. Skands, Color connections, multijets, in *Proceedings of the First International Workshop on Multiple Partonic Interactions at the LHC (MPI@LHC08)*, Perugia, Italy, October 27–31, 2008, ed. by P. Bartalini and L. Fano, (DESY-PROC, 2008), to appear
67. J. Boudreau, V. Tsulaia, The GeoModel toolkit for detector description, in *Computing in High Energy and Nuclear Physics 2004 (CHEP2004)*, ed. by A. Aimar, J. Harvey, N. Knoors, Interlaken, Switzerland, September 27–October 1, 2004 (CERN, Geneva, 2004)
68. G. Aad et al. (ATLAS Collaboration), Expected performance of the ATLAS experiment, detector, trigger, and physics. CERN-OPEN-2008-020 (2008)
69. F. Viegas, R. Hawkings, G. Dimitrov, Relational databases for conditions data and event selection in ATLAS. J. Phys. Conf. Ser. **119**, 001001 (2007). In Proceedings of Computing in High Energy and Nuclear Physics 2007 Conference (CHEP2007), Victoria, BC, Canada, September 2–7, 2007, ed. R. Sobie, R. Tafirout, J. Thomson

70. SQLite, http://www.sqlite.org, 2008
71. R.F. van der Lans, *The SQL Guide to SQLite*, 1st edn. http://lulu.com, 2009
72. C. Newman, *SQLite (Developer's Library)*, 1st edn. (Sams, 2004)
73. B. Di Girolamo et al., Beamline instrumentation in the 2004 combined ATLAS testbeam. ATL-TECH-PUB-2005-001 (2005)
74. C. Adorisio et al., Nucl. Instrum. Methods Phys. Res. A **593**, 232–254 (2008)
75. C. Adorisio et al., System test of the ATLAS muon spectrometer in the H8 beam at the CERN SPS. Nucl. Instrum. Methods A **593**, 232–254 (2008)
76. M. Hurwitz, Performance of ATLAS tile calorimeter production modules in calibration testbeams. Nucl. Instrum. Methods Phys. Res. A **572**, 80–81 (2007)
77. P. Strizenec, A. Minaenko (ATLAS LAr Endcap group), J. Phys. Conf. Ser. **160**, 012078 (2009)
78. H. Bartko, Performance of the combined ATLAS liquid argon end-cap calorimeters in beam tests at the CERN SPS. Diploma Thesis ATL-LARG-2004-007, 2004
79. ATLAS Pixel Collaboration, Pixel offline analysis for endcap A cosmic data. ATL-INDET-PUB-2008-003 (2008)
80. B. Demirkoz, Construction and performance of the ATLAS SCT barrels and cosmic tests. CERN-THESIS-2008-001 (2008)
81. Z. Broklova, Z. Dolezal, Simulations of ATLAS silicon strip detector modules in ATHENA framework. CERN-THESIS-2005-007 (2004)
82. T.H. Kittelmann, Slepton spin determination and simulation of the transition radiation tracker at the ATLAS experiment. PhD in physics, Niels Bohr Institute, University of Copenhagen, 2007
83. D. Salihagic, Comparison of beam test results of the combined ATLAS liquid argon endcap calorimeters with Geant3 and Geant4 simulations, in *11th International Conference on Calorimetry in High Energy Physics*, ed. by C. Cecci, P. Cenci, P. Lubrano, M. Pepe-Altarelli (World Scientific, Singapore, 2004), p. 314
84. A. Dotti, A. Lupi, C. Roda, Results from ATLAS tile calorimeter: a comparison between data and Geant4 simulation. Nucl. Phys. B, Proc. Suppl. **150**, 106–109 (2006)
85. N.C. Benekos, D. Rebuzzi, GEANT4 simulation of the ATLAS muon spectrometer, in *9th ICATPP Conference on Astroparticle, Particle, Space Physics, Detectors and Medical Physics Applications*, ed. by M. Barone, E. Borchi, A. Gaddi, C. Leroy, L. Price, Villa Erba, Como, Italy, 17–21 October, 2005 (World Scientific, Singapore, 2005)
86. K. Kordas, G. Parrour, S. Simion, GEANT4 for the ATLAS electromagnetic calorimeter, in *9th Conference on Calorimetry in High Energy Physics (CALOR 2000)*, ed. by B. Aubert, J. Colas, P. Nédélec, L. Poggioli, Annecy, France, 9–14 October 2000 (Lab. Annecy Phys. Part., Annecy-le-Vieux, 2000)
87. A. Kiryunin, D. Salihagic, P. Strizenec (ATLAS/LAr-HEC Collaboration), GEANT4 simulations of the ATLAS hadronic end cap calorimeter, in *9th Conference on Calorimetry in High Energy Physics (CALOR 2000)*, ed. by B. Aubert, J. Colas, P. Nédélec, L. Poggioli, Annecy, France, 9–14 October 2000 (Lab. Annecy Phys. Part., Annecy-le-Vieux, 2000)
88. P. Loch, R. Mazini, Comparison of experimental electron signals with GEANT3 and GEANT4 simulations for the ATLAS forward calorimeter prototype, in *9th Conference on Calorimetry in High Energy Physics (CALOR 2000)*, ed. by B. Aubert, J. Colas, P. Nédélec, L. Poggioli, Annecy, France, 9–14 October 2000 (Lab. Annecy Phys. Part., Annecy-le-Vieux, 2000)
89. A. Dell'Acqua et al., Development of the ATLAS simulation framework, in *Computing in High Energy and Nuclear Physics 2001 Conference (CHEP2001)*, IHEP, Beijing, China, September 3–7, 2001, ed. by H.S. Chen, (e-article, 2001)
90. A. Dell'Acqua et al., ATLAS simulation readiness for first data at LHC. J. Phys. Conf. Ser. **119**, 001001 (2007). In Proceedings of Computing in High Energy and Nuclear Physics 2007 Conference (CHEP2007), Victoria, BC, Canada, September 2–7, 2007, ed. R. Sobie, R. Tafirout, J. Thomson
91. C. Amsler et al.(Particle Data Group), Phys. Lett. B **667**, 1 (2008)
92. G. Battistoni et al., The FLUKA code: description and benchmarking. AIP Conf. Proc. **896**, 31–49 (2007). In Proceedings of the Hadronic Shower Simulation Workshop 2006, Fermilab 6–8 September 2006
93. A. Fassò, A. Ferrari, J. Ranft, P.R. Sala, FLUKA: a multi-particle transport code. CERN 2005-10, 2005, Also INFN/TC_05/11, SLAC-R-773
94. A. Rimoldi et al., First report of the simulation optimization group. ATL-SOFT-PUB-2008-002 (2008)
95. A. Rimoldi et al., Final report of the simulation optimization task force. ATL-SOFT-PUB-2008-004 (2008)
96. R. Brun, F. Rademakers, ROOT—an object oriented data analysis framework. Nucl. Instrum. Methods Phys. Res. A **389**, 81–86 (1997). In Proceedings AIHENP '96 Workshop, Lausanne, Sep. 1996. See also http://root.cern.ch
97. T. Kittelmann (ATLAS Collaboration), The virtual point 1 event display for the ATLAS experiment, in *Proceedings of Computing in High Energy and Nuclear Physics 2009 Conference (CHEP2009)*, Prague, Czech Republic, March 22–28, 2009
98. G. Taylor, Visualizing the ATLAS inner detector with Atlantis. Nucl. Instrum. Methods Phys. Res. A **549**, 183–187 (2005)
99. M. Virchaux, D. Pomarède, The PERSINT manual. ATL-SOFT-2001-003 (2001)
100. S. Gadomski, Model of the SCT detectors and electronics for the ATLAS simulation using Geant4. ATL-SOFT-2001-005 (2001)
101. W. Lampl et al., Digitization of LAr calorimeter for CSC simulations. ATL-LARG-PUB-2007-011 (2007)
102. D. Rebuzzi et al., Geant4 muon digitization in the ATHENA framework. ATL-SOFT-PUB-2007-001 (2007)
103. F. James, Comput. Phys. Commun. **60**, 329–344 (1990)
104. S. Baranov et al., ATLAS radiation background task force summary. ATL-GEN-2005-001 (2005)
105. C. Zeitnitz, T.A. Gabriel, The GEANT-CALOR interface and benchmark calculations for ZEUS calorimeters. Nucl. Instrum. Methods A **349**, 106–111 (1994)
106. I. Azhgirey, I. Baishev, K.M. Potter, V. Talanov, Cascade simulations for the machine induced background study in the IR1 of the LHC. LHC Project Note 324 (2003)
107. I. Azhgirey, I. Baishev, K.M. Potter, V. Talanov, Machine induced background in the low luminosity insertions of the LHC. LHC Project Report 567 (2002)
108. E. Barberio et al., The Geant4-based ATLAS fast electromagnetic shower simulation. ATL-SOFT-CONF-2007-002 (2007)
109. E. Barberio et al., The fast shower simulation in the ATLAS calorimeter. J. Phys. Conf. Ser. **119**, 001001 (2007). In Proceedings of Computing in High Energy and Nuclear Physics 2007 Conference (CHEP2007), Victoria, BC, Canada, September 2–7, 2007, ed. R. Sobie, R. Tafirout, J. Thomson
110. E. Richter-Was, D. Froidevaux, L. Poggioli, ATLFAST 2.0 a fast simulation package for ATLAS. CERN-ATL-PHYS-98-131 (1998)
111. S. Dean, P. Sherwood, Athena-Atlfast. http://www.hep.ucl.ac.uk/atlas/atlfast/, 2008
112. K. Edmonds et al., TheFast ATLAS track simulation (FATRAS). ATL-SOFT-PUB-2008-01 (2008)
113. G. Soyez, The SISCone and anti-kt jet algorithms (2008). 0807.0021 [hep-ph]
114. M. Cacciari, G.P. Salam, Dispelling the N**3 myth for the k(t) jet-finder. Phys. Lett. B **641**, 57–61 (2006). hep-ph/0512210

115. A. Salzburger, S. Todorova, M. Wolter, The ATLAS tracking geometry description. ATL-SOFT-PUB-2007-004 (2007)
116. A. Salzburger, The ATLAS track extrapolation package. ATL-SOFT-PUB-2007-05 (2007)
117. J.L. Henning, S.P.E.C. CPU2000, Measuring CPU performance in the new millenium. IEEE Comput. 28–35 (2000)
118. C.P.U. Normalization Criterion, http://www.egee.cesga.es/EGEE-SA1-SWE/accounting/guides/NormalizationCriterion, 2009
119. A. De Salvo, F. Brasolin, Benchmarking the ATLAS software through the kit validation engine, in *Proceedings of Computing in High Energy and Nuclear Physics 2009 Conference (CHEP2009)*, Prague, Czech Republic, March 22–28, 2009
120. GCC, the GNU Compiler Collection. http://gcc.gnu.org, 2008
121. Scientific Linux CERN, 4 (SLC4). http://linux.web.cern.ch/linux/scientific4, 2008
122. Y. Perrin, F. Orellana, M. Roy, D. Feichtinger, The LCG Savannah software development portal. CERN Yellow Report CERN 2005-002 (2005), pp. 609–612
123. T. Carli, H. Hakobyan, A. Henriques-Correia, M. Simonyan, Measurement of pion and proton longitudinal shower profiles up to 20 nuclear interaction lengths with the ATLAS time calorimeter. ATL-TILECAL-PUB-2007-008 (2007)
124. P. Adragna et al., Testbeam studies of production modules of the ATLAS tile calorimeter. ATL-TILECAL-PUB-2009-002 (2009)
125. P. Speckmayer, Comparison of data with Monte Carlo simulations at the ATLAS barrel combined testbeam 2004. J. Phys. Conf. Ser. **160** (2009). In Proceedings of the XIII International Conference on Calorimetry in High Energy Physics (CALOR 08), Pavia, Italy, May 26–30, 2008, ed. M. Fraternali, G. Gaudio, M. Livan
126. A.E. Kiryunin et al., GEANT4 physics evaluation with testbeam data of the ATLAS hadronic end-cap calorimeter, J. Phys., Conf. Ser. **160** (2009). In Proceedings of the XIII International Conference on Calorimetry in High Energy Physics (CALOR 08), Pavia, Italy, May 26–30, 2008, ed. M. Fraternali, G. Gaudio, M. Livan
127. M. Cooke et al., In situ commissioning of the ATLAS electromagnetic calorimeter with cosmic muons. ATL-LARG-PUB-2007-013 (2007)
128. M. Campanella, A. Ferrari, P.R. Sala, S. Vanini, First calorimeter simulation with the FLUGG prototype. ATL-SOFT-99-004 (1999)