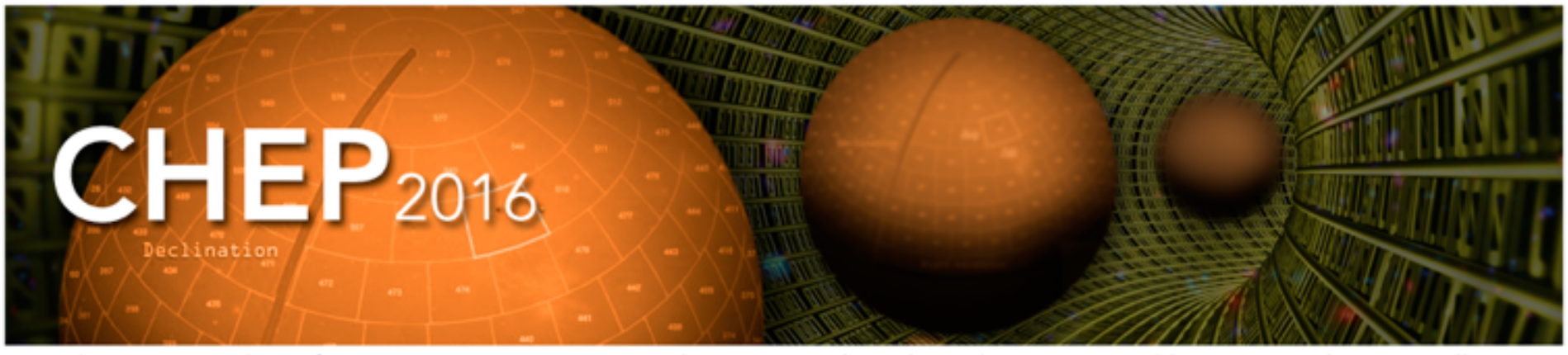
A browser-based event display for the CMS Experiment at the LHC using WebGL

Tom McCauley
University of Notre Dame, USA
thomas.mccauley@cern.ch
@tpmccauley
http://cern.ch/ispy-webgl



22nd International Conference on Computing in High Energy and Nuclear Physics, Hosted by SLAC and LBNL, Fall 2016

Outline

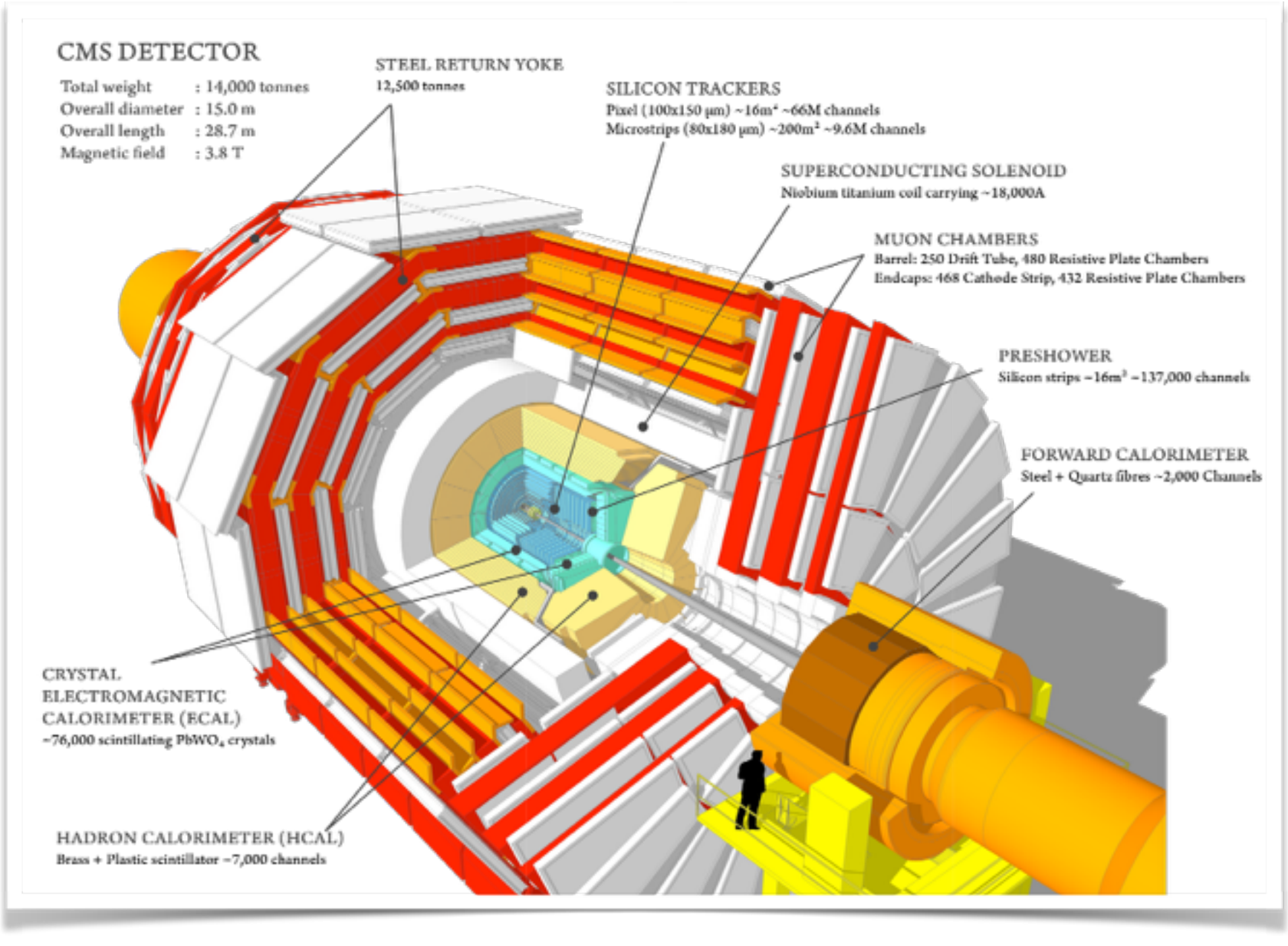
- CMS and the LHC
- Requirements
- WebGL and three.js
- Features
- Usage
- Future plans

CMS and the LHC

- CMS (Compact Muon Solenoid) is one of the two general-purpose experiments at the Large Hadron Collider at CERN

- Broad physics program: searches for SUSY and beyond the SM, measurements of QCD, electroweak, top, B, forward, heavy-ion, and Higgs physics

- Run 2 of LHC: collected (as of 6 Oct) 31 1/fb of proton-proton collision data at sqrt(s) = 13 TeV



http://cern.ch/cms

Requirements

- Production of high-quality public event display images for CMS Experiment
- Easy-to-use and easy-to-distribute application for the public (taking into account varied and possibly restrictive computing environments) *e.g.* for use in CMS masterclasses for high-school students
- For the developer: easy to develop, maintain, and distribute
- How to fulfill requirements (partly): use JavaScript, HTML, CSS, and WebGL in the browser
- Additional side benefits include smaller code base, fewer dependencies, and rapid prototyping and distribution

Input format

- Ease of distribution does not come entirely from using the browser: life is made partly easier by choice of input format

- ig format: a zip archive containing one or more JSON file which contain necessary information for rendering of event and detector

- Developed for predecessor desktop application iSpy (Coin3D+Qt+SoQt) and also used in browser-based display using HTML5 canvas-based rendering

- Extract information from CMS event format with CMSSW C++ code and convert to ig format; geometry exported as well

- Dependence on CMSSW is minimized & file format is human-readable, portable, and easily parsed using JavaScript, Python, C#, Ruby, …

```
Archive:  4lepton.ig
  inflating: Events/Run_178424/Event_666626491
  inflating: Events/Run_193575/Event_400912970
  inflating: Events/Run_195099/Event_137440354
```


```
{"Types": {"Event_V2": [["run", "int"],["event", "int"],["ls", "int"],["orbit", "int"]\
,["bx", "int"],["time", "string"],["localtime", "string"]]
,
"Products_V1": [["Product", "string"]]
,
"BasicClusters_V1": [["energy", "double"],["pos", "v3d"],["eta", "double"],["phi", "do\
uble"],["algo", "string"]]
,
"RecHitFractions_V1": [["detid", "int"],["fraction", "double"],["front_1", "v3d"],["fr\
ont_2", "v3d"],["front_3", "v3d"],["front_4", "v3d"],["back_1", "v3d"],["back_2", "v3d\
"],["back_3", "v3d"],["back_4", "v3d"]]
```



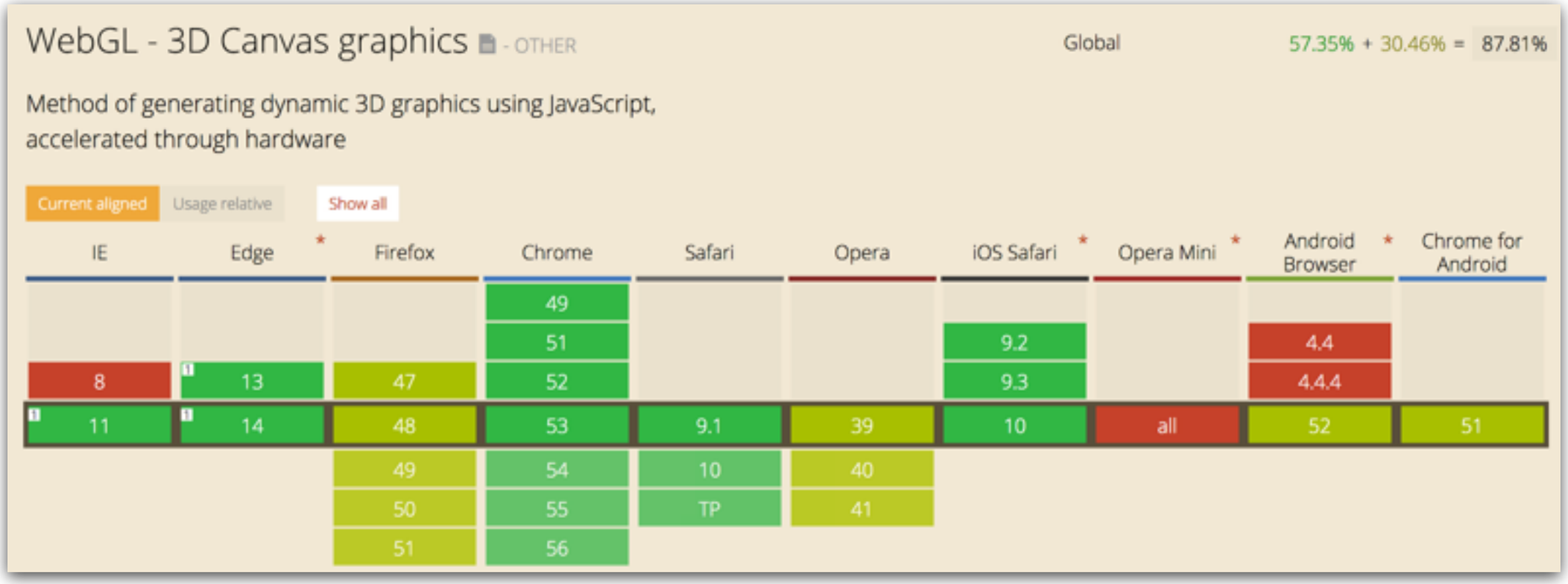
```
"TrackDets_V1": {type: ispy.BOX, on: false, group: "Tracking", name: "Matching Tracker Dets",
  fn: ispy.makeTrackerPiece, style: {color: "rgb(100%, 100%, 0%)", opacity: 0.5, linewidth: 0.5}},
"TrackingRecHits_V1": {type:ispy.POINT, on:false, group:"Tracking", name: "Tracking Rec Hits",
  fn:ispy.makeTrackingRecHits, style: {color: "rgb(100%, 100%, 0%)", size: 0.05}},
"SiStripClusters_V1": {type: ispy.POINT, on:false, group:"Tracking", name: "Si Strip Clusters",
  fn: ispy.makeTrackingClusters, style:{color: "rgb(80%, 20%, 0%)", size: 0.05}},
"SiPixelClusters_V1": {type: ispy.POINT, on:false, group:"Tracking", name: "Si Pixel Clusters",
  fn: ispy.makeTrackingClusters, style:{color: "rgb(100%, 40%, 0%)", size: 0.05}},
```


DOI: 10.1088/1742-6596/396/2/022002
DOI: 10.1088/1742-6596/396/2/022022

http://cms-outreach.github.io/ispy-webgl/
https://github.com/cms-outreach/ispy-analyzers

WebGL

“WebGL is a cross-platform, royalty-free web standard for a low-level 3D graphics API based on OpenGL ES 2.0, exposed through the HTML5 Canvas element as Document Object Model interfaces”: https://www.khronos.org/webgl/



http://caniuse.com/#search=WebGL

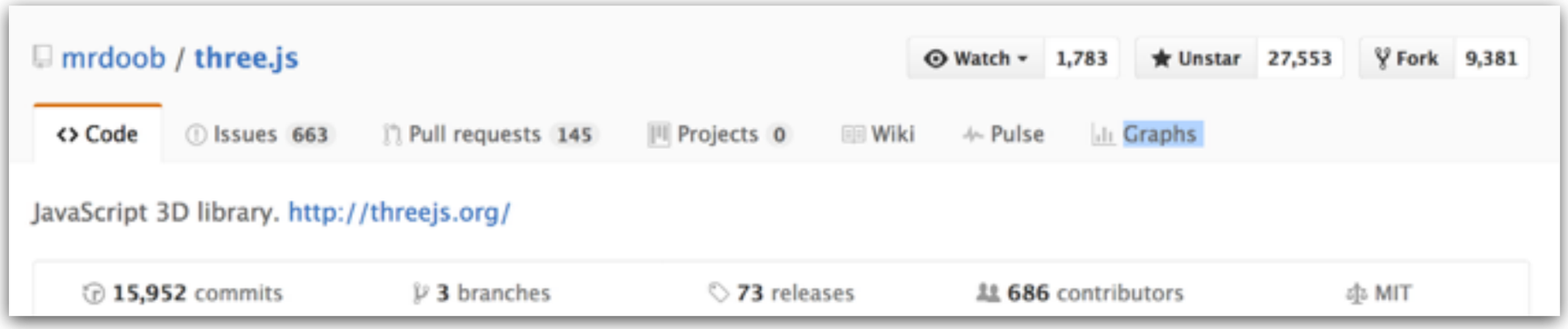
three.js

- A JavaScript API for WebGL; "raw" WebGL requires a lot of boilerplate and may include shader language

- Originally developed and currently managed by Ricardo Cabello @mrdoob

- Large open-source developer community

http://threejs.org



Features (i)

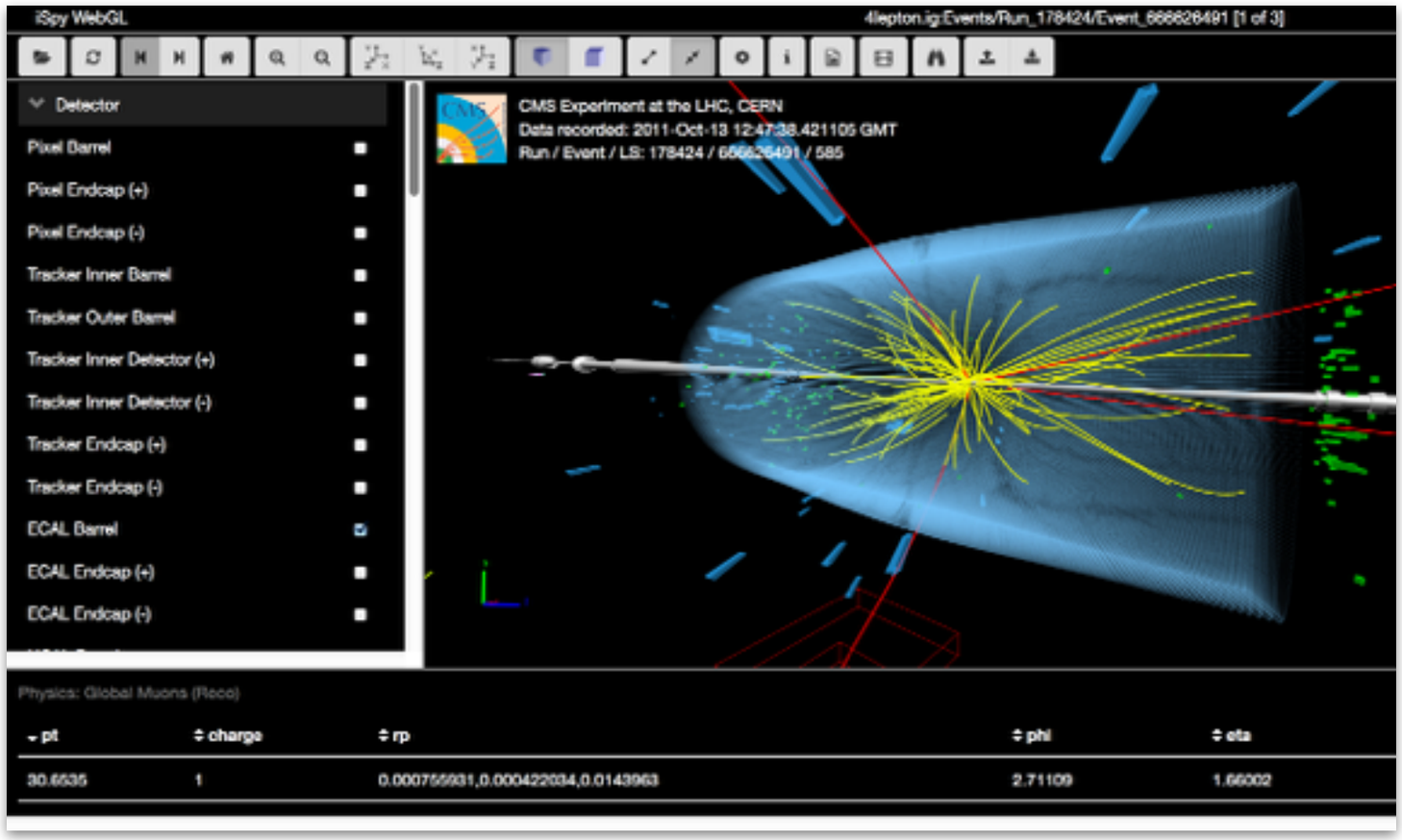
- Standard event display controls: load event, prev/next event, start view, zoom in/out, view along axes, perspective/orthographic views

- Can be run completely client-side and offline, with local file loading via FileReader API

- Tree, 3D, and sortable table views

- Correlated (between table and 3D views) picking of physics objects

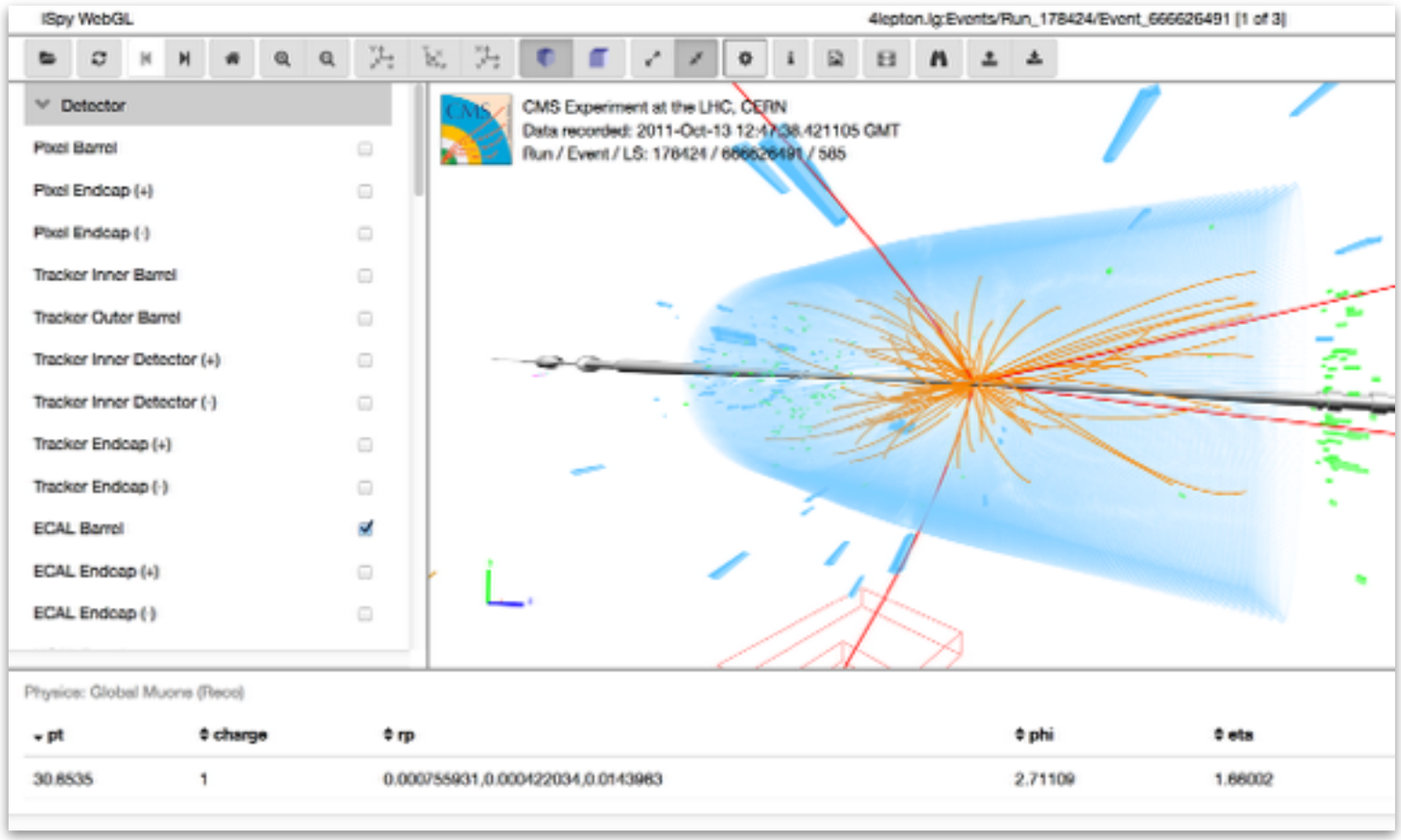
- Touch events (one-finger to rotate, two-finger zoom, and three-finger panning) enabled so works on mobiles, tablets, and touch screens in the browser



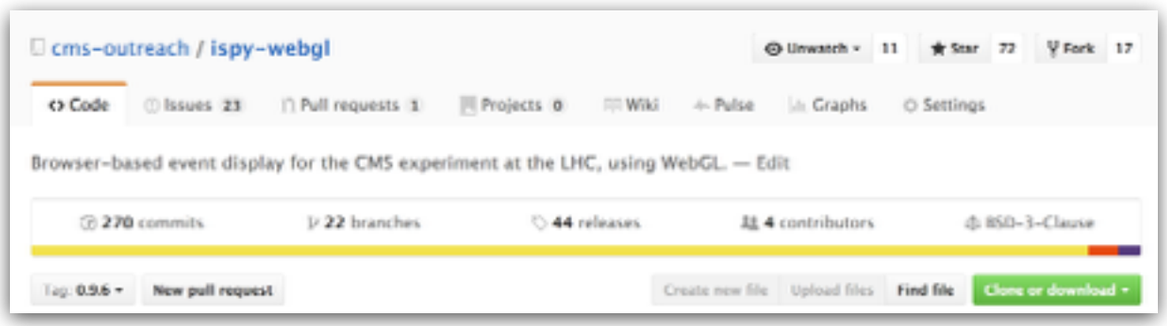
First release Dec 2014; fourth and latest release Apr 2016. Latest stable version at http://cern.ch/ispy-webgl

Features (ii)

- Switching between WebGL, canvas, and SVG renderers (defaults to canvas when WebGL not available)
- bootstrap.js for layout and therefore scaleable for mobile devices
- Default animation sequence
- Import/export of 3D files
- Stereo view for use with Google Cardboard in mobile browser



Code available at
https://github.com/cms-outreach/ispy-webgl



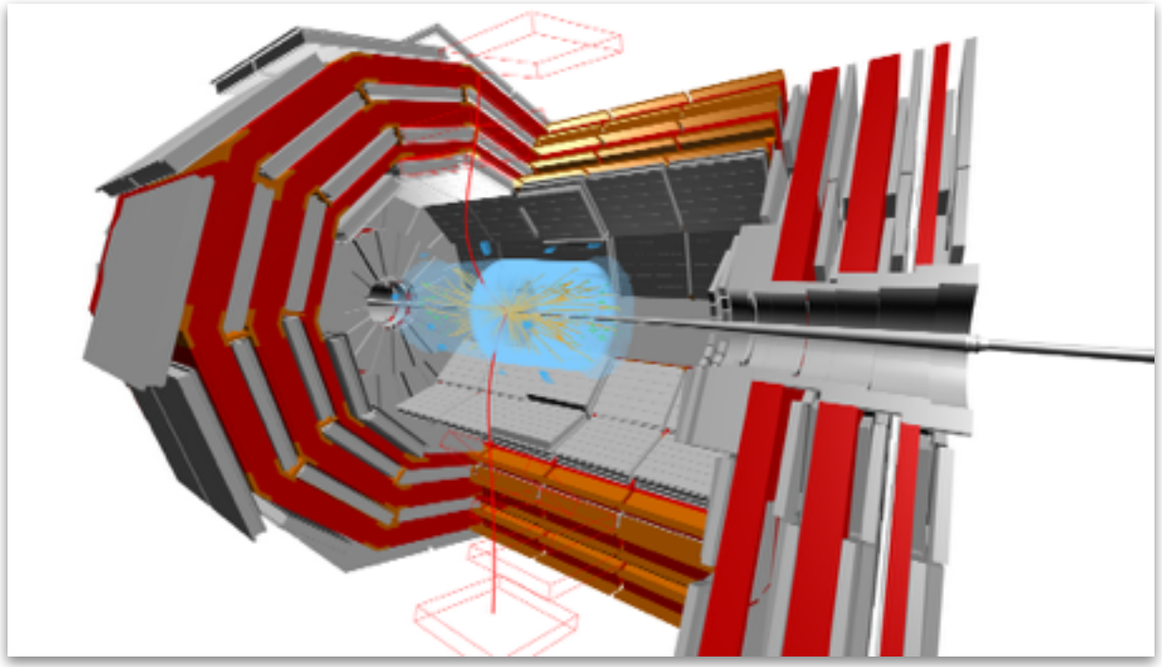
Import/export

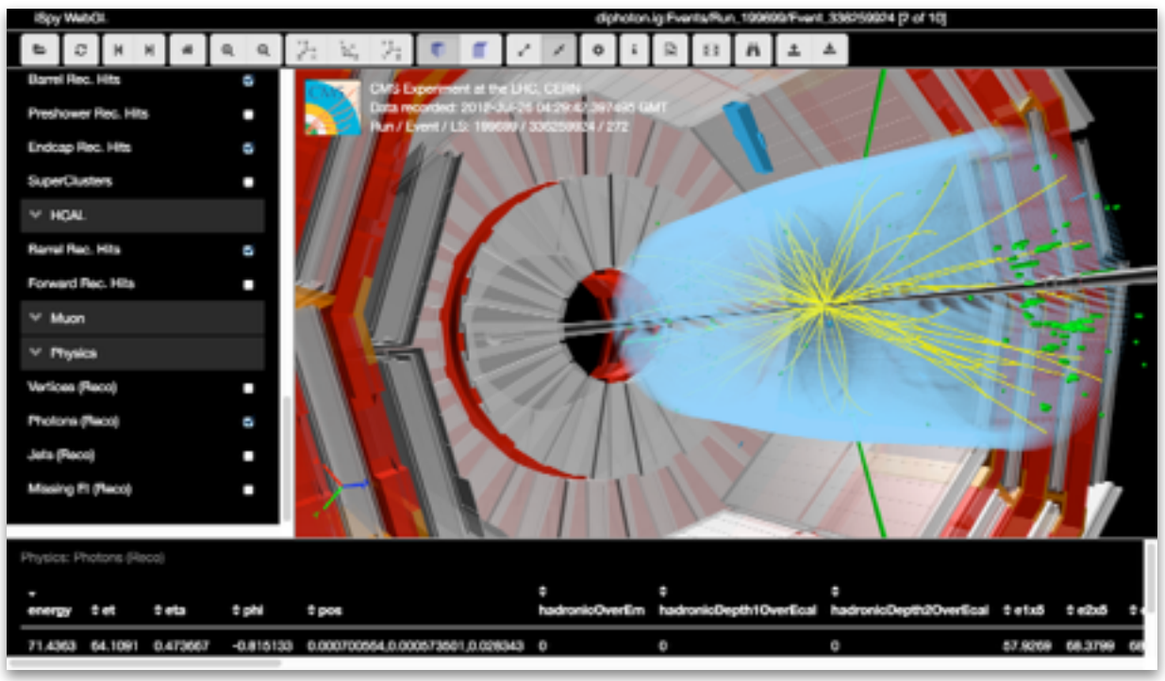
- The detailed simulation geometry of CMS can be read into SketchUp (http://www.sketchup.com)

- SketchUp exports to multiple 3D formats and three.js supports import and export of multiple formats

- For now: export .obj and .mtl from SketchUp and import into iSpy WebGL

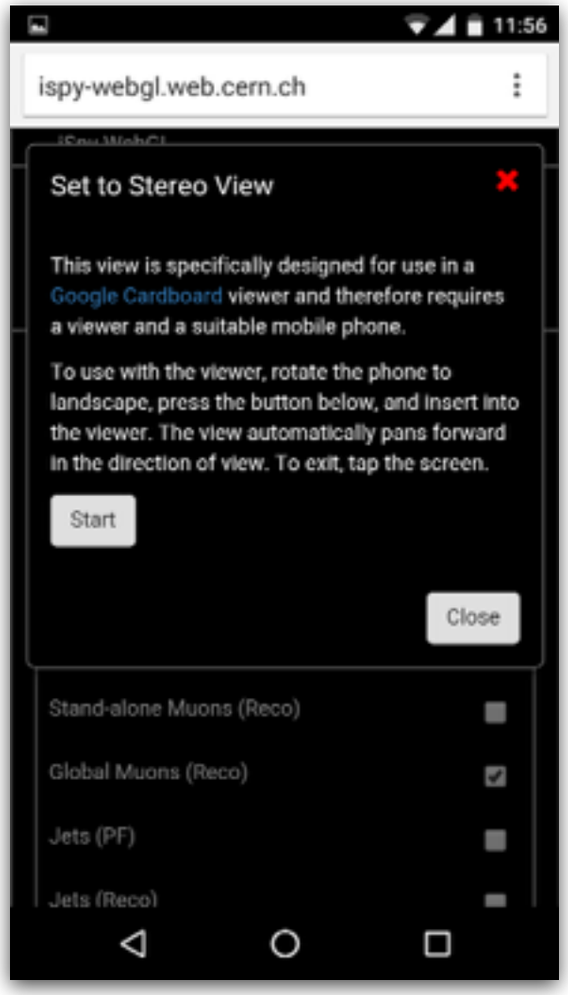
- iSpy WebGL can export event and detector information to .obj

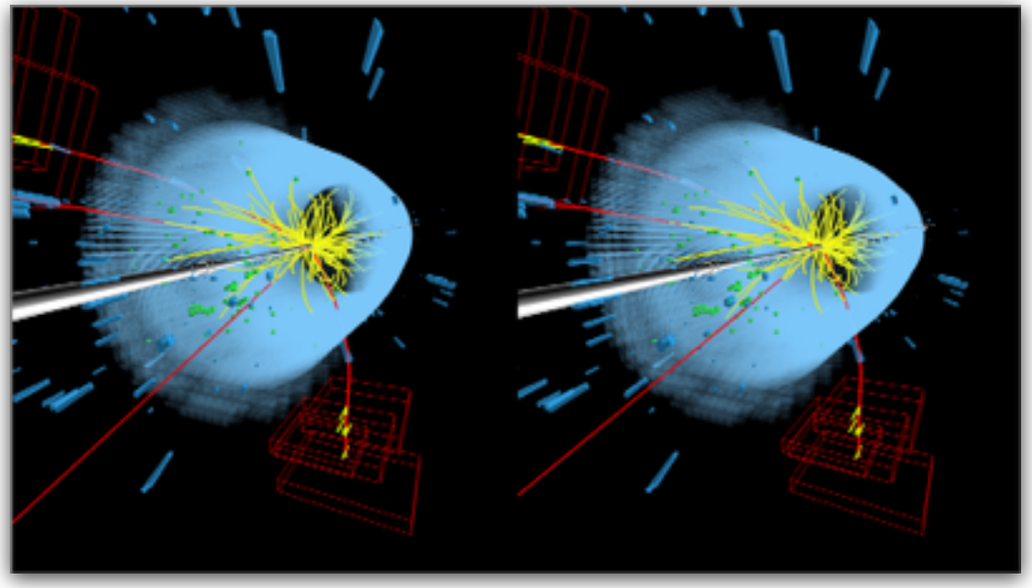




https://twiki.cern.ch/twiki/bin/view/CMSPublic/SketchUpCMS

Stereo view



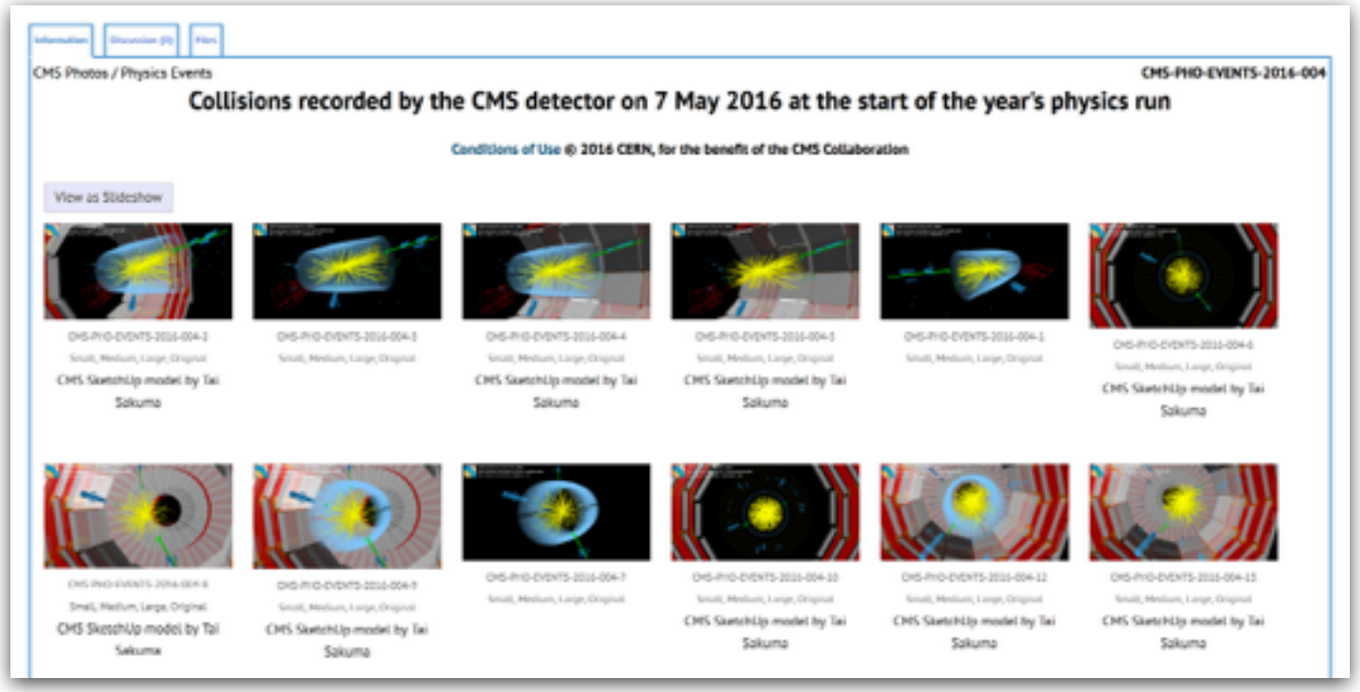


The main features of the stereo view were developed with two CERN summer students at the Mozilla Science Lab Global Sprint, 4-5 June 2015: https://science.mozilla.org/programs/events/global-sprint-2015

https://vr.google.com/cardboard/

Usage (i)

- Production of event display and animations for the public in Run 2 of the LHC
- Version used in CMS masterclasses (where students conduct simplified analyses of CMS data using visualization of events; developed by QuarkNet); as part of the International Masterclasses held 11 Feb - 23 Mar 2016 ~3000 students used display



https://cds.cern.ch/record/2151078

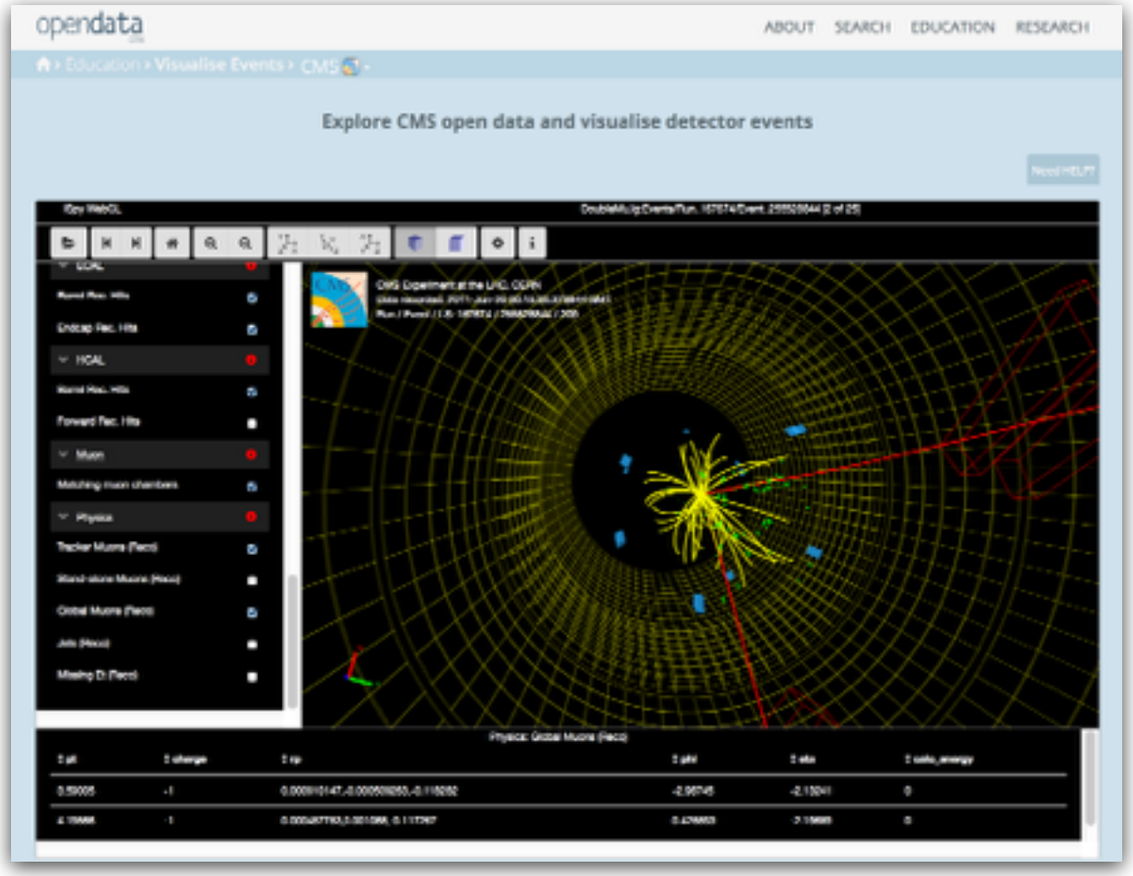
CMS masterclass in Addis Ababa, Aug 2016

https://quarknet.i2u2.org/
http://physicsmasterclasses.org/

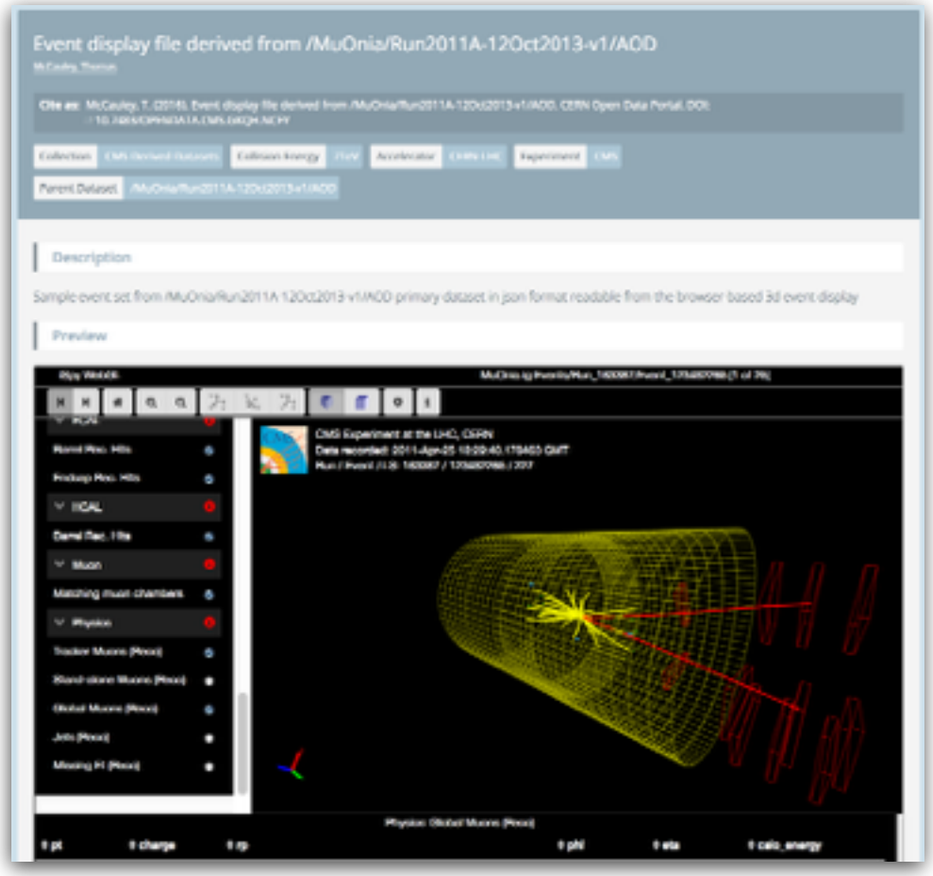
Usage (ii)

- Application in the CERN Open Data Portal (http://opendata.cern.ch) : a subset of events from each dataset released to the public by CMS is available for visualization; latest release was 18 Apr 2016: peak number of visits from distinct IPs in a day ~46k (thanks to Reddit AMA on open data release), settling down to ~100 per day by Jun 2016, ~66k visits total over this time period which is about 30% of the total number of visits

- As an Invenio (http://inveniosoftware.org/) previewer plugin for the ig file format (https://github.com/inveniosoftware/invenio-previewer-ispy)



http://opendata.cern.ch/visualise/events/CMS

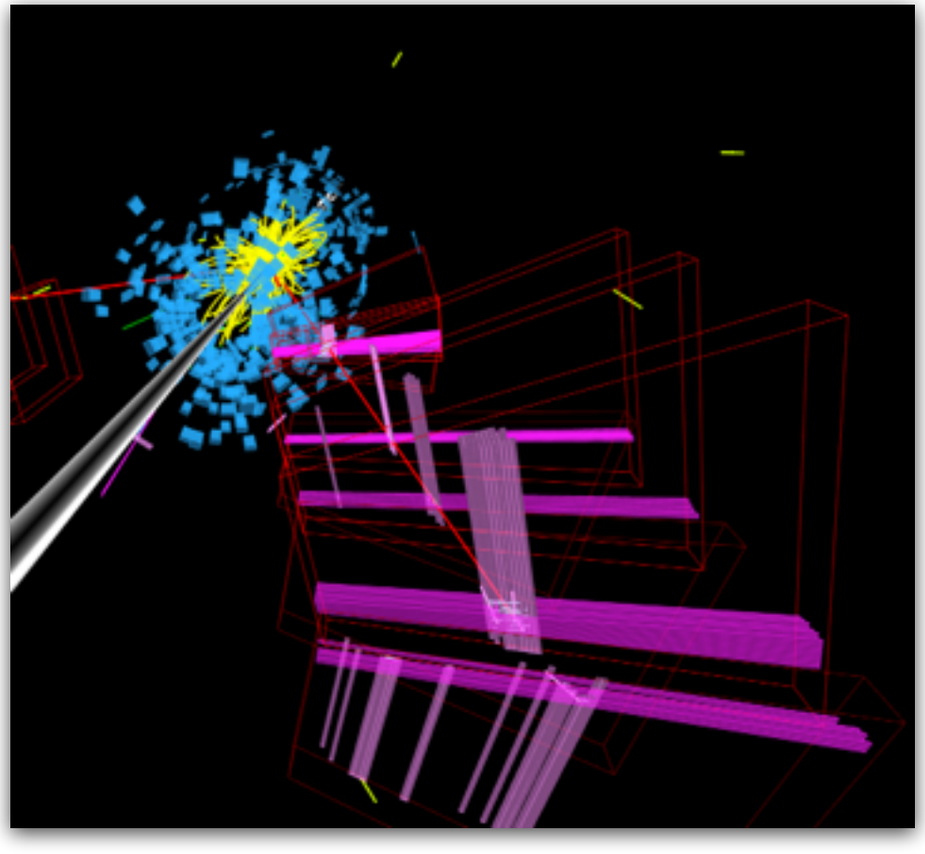


http://opendata.cern.ch/record/626

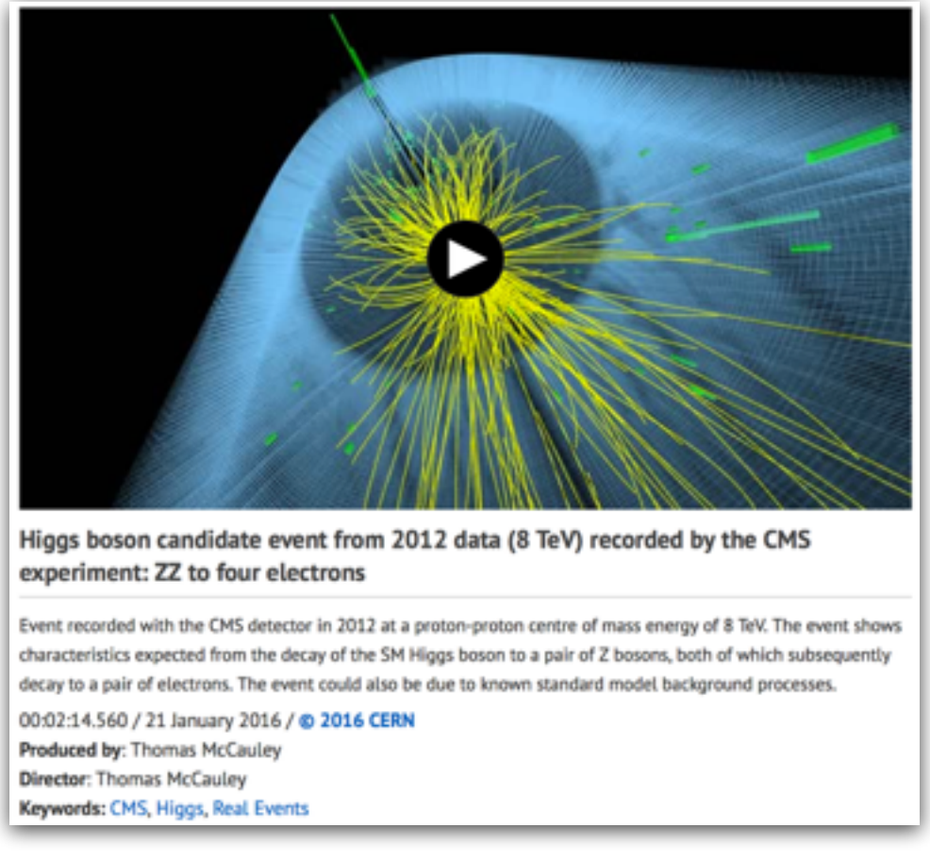
Usage (iii)

- The CSC muon subdetector of CMS uses the display for detailed technical examination of individual collision events to better understand event kinematics and detector behavior, and to explore muon track segment reconstruction algorithms

- Production of high-resolution animations for public exhibits and use: *e.g.* for “Uncertainty” exhibition at Alyce de Roulet Williamson Gallery, Pasadena (http://williamsongallery.net/uncertainty) and 3D film for CMS Point 5



reconstructed muon in the CSCs, credit: T. Cox



https://cds.cern.ch/record/2126012

Future plans

- Besides the usual maintenance, …
- Expose, in a user-friendly way, style configuration: *e.g.* colors, transparency, …
- Keep up with evolving versions of CMSSW and CMS geometry
- Dedicated R-phi and R-Z views
- Improved configuration and export of animations
- Keep up with and contribute to three.js development (*e.g.* OBJExporter and prototype GDMLImporter)
- Additional support for import/export of 3D geometry formats
- JavaScript framework?
- Explore the possibilities of evolving WebVR API
- Much further down the road: what does WebGL2 and OpenGL ES 3.0 offer?

Acknowledgements

- Past developers of desktop application and ig format: G. Alverson, G. Eulisse, S. Muzzafar, I. Osbourne, L. Taylor, & L. Tuura

- Past developers of HTML5 canvas browser-based display: M. Hategan & P. Nguyen

- 3D models of CMS using SketchUp: T. Sakuma

- Contributors to iSpy WebGL: L. Barnard, C. Logren, & M. Saunby

- T. Simko for usage numbers for CODP and K. Cecire and U. Bilow for usage numbers for masterclasses

- Thanks to Invenio team for assistance in developing for CERN Open Data Portal

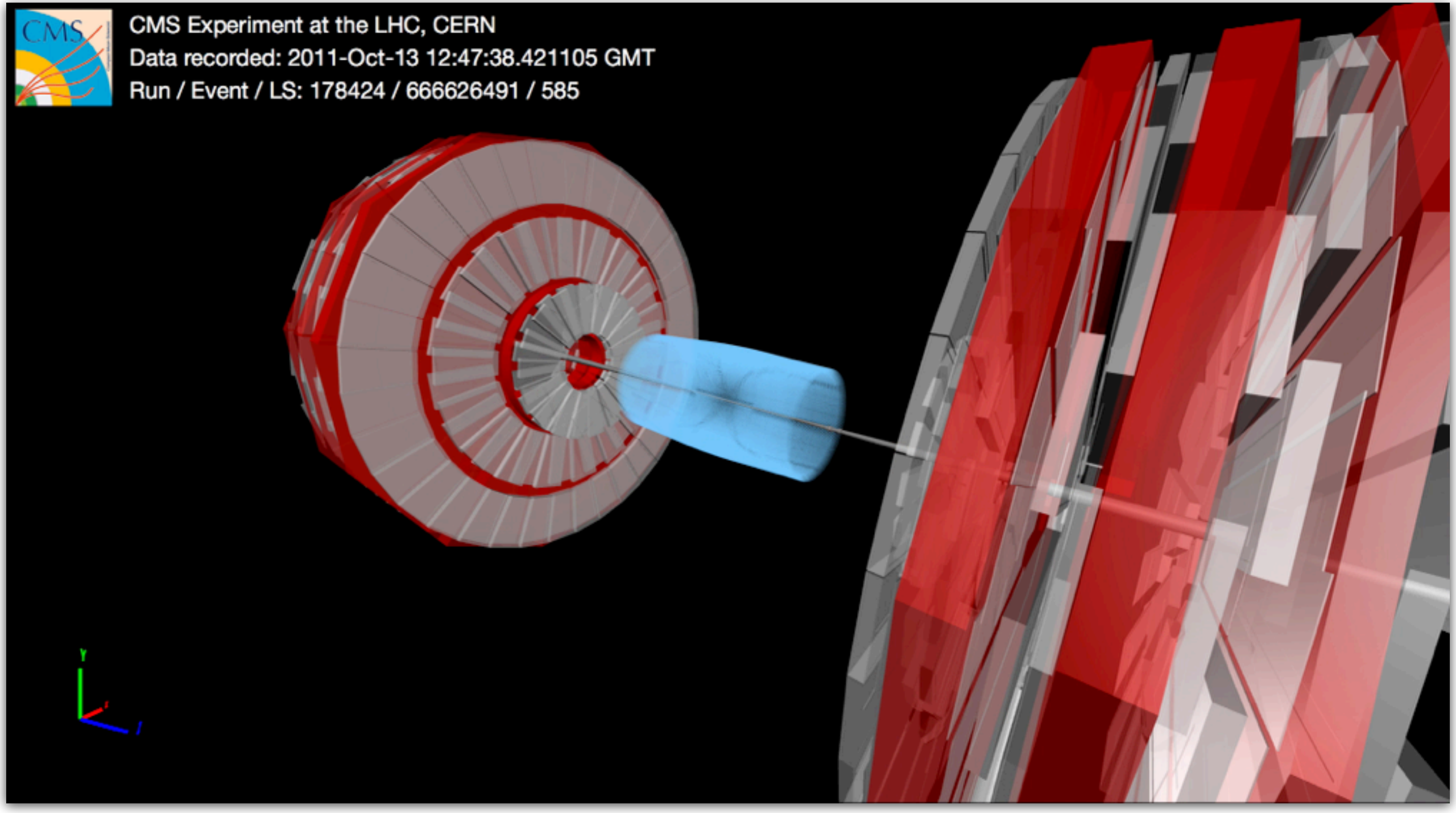
- Support from CMS Collaboration and QuarkNet

Thank you

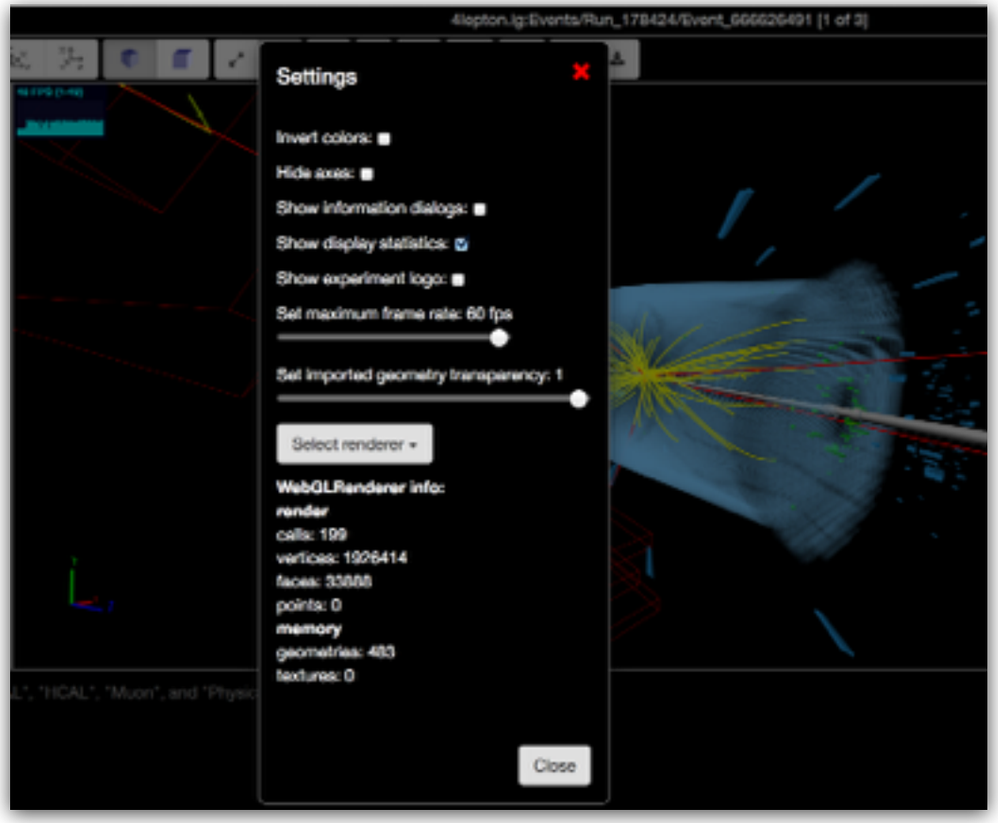
Backup

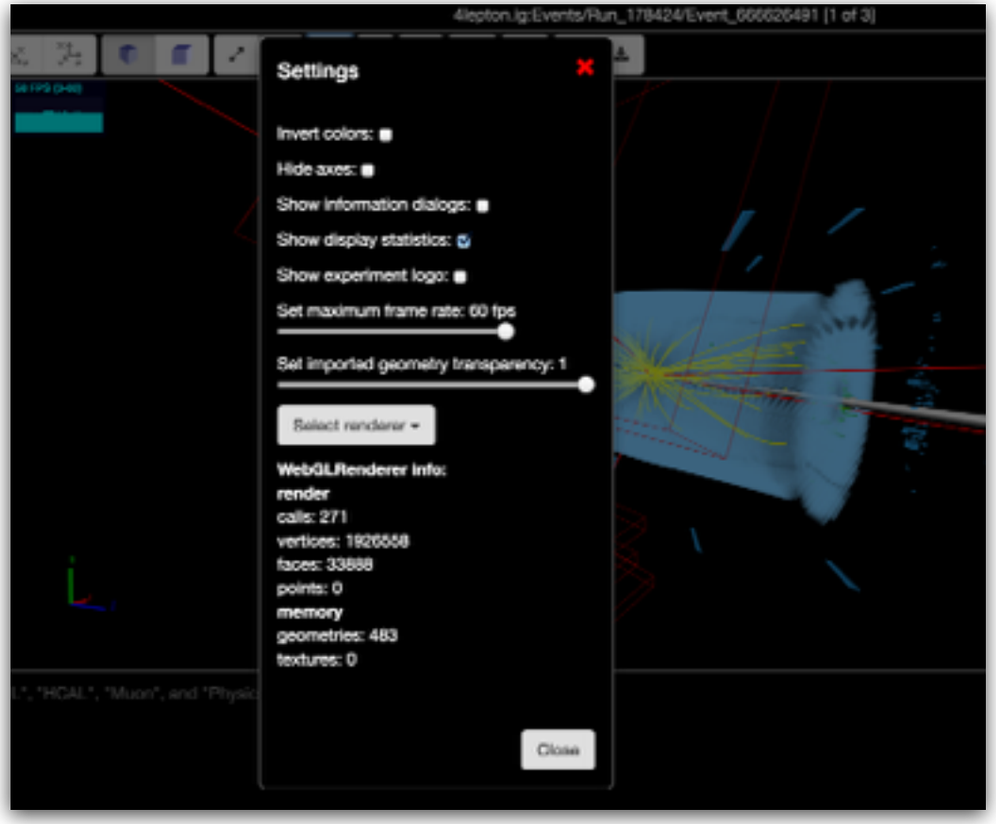
Public event displays

- First 13TeV collisions, for tuning the LHC, seen by CMS: https://cds.cern.ch/record/2017634
- Di-muon event recorded by the CMS detector (Run 2, 13 TeV): https://cds.cern.ch/record/2114775
- Highest-mass di-photon event recorded by CMS as of Dec '15 (Run 2, 13 TeV): https://cds.cern.ch/record/2114776
- Di-photon event recorded by the CMS detector (Run 2, 13 TeV): https://cds.cern.ch/record/2114779
- Large-mass di-jet event recorded by the CMS detector (Run 2, 13 TeV): https://cds.cern.ch/record/2114780
- Multi-jet event recorded by the CMS detector (Run 2, 13 TeV): https://cds.cern.ch/record/2114784
- Di-photon events recorded by the CMS detector (Run 2, 13 TeV, 0 T): https://cds.cern.ch/record/2139792
- Commissioning with low-intensity beams…during the early hours of 23 April 2016: https://cds.cern.ch/record/2148222
- Collisions recorded by the CMS detector on 7 May 2016 at the start of the year's physics run: https://cds.cern.ch/record/2151078
- Large-mass di-jet event recorded by the CMS detector (Run 2, 13 TeV): https://cds.cern.ch/record/2203615
- Higgs and associated vector boson event recorded by CMS (Run 2, 13 TeV): https://cds.cern.ch/record/2210657
- Higgs boson produced via vector boson fusion event recorded by CMS (Run 2, 13 TeV): https://cds.cern.ch/record/2210658

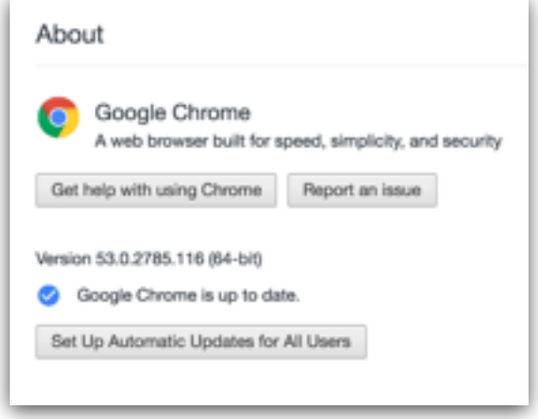


Looking at this on a pdf file? This is an animation which you can find here:
https://www.dropbox.com/s/3rngni84wm0hb20/4mu.mov?dl=0


4lepton.lg:Events/Run_178424/Event_666626491 [1 of 3]
Settings
Invert colors:
Hide axes:
Show information dialogs:
Show display statistics:
Show experiment logo:
Set maximum frame rate: 60 fps
Set imported geometry transparency: 1
Select renderer
WebGLRenderer info:
render
calls: 199
vertices: 1926414
faces: 33888
points: 0
memory
geometries: 483
textures: 0
Close

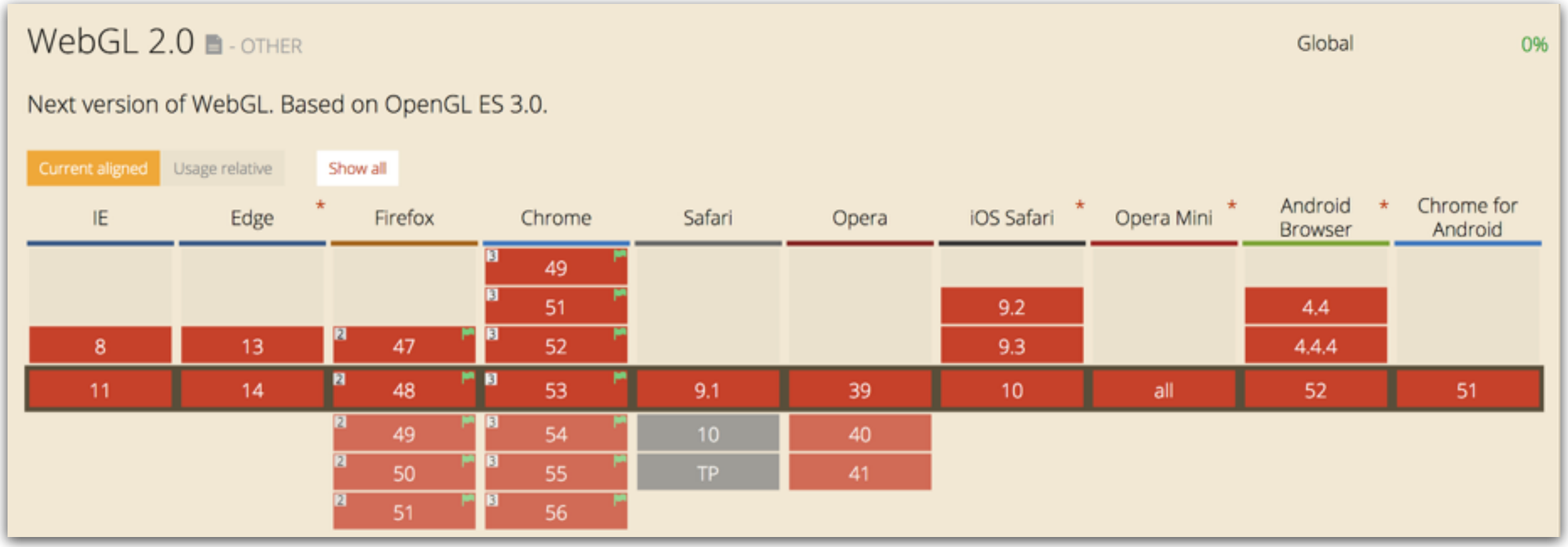

4lepton.lg:Events/Run_178424/Event_666626491 [1 of 3]
Settings
Invert colors:
Hide axes:
Show information dialogs:
Show display statistics:
Show experiment logo:
Set maximum frame rate: 60 fps
Set imported geometry transparency: 1
Select renderer
WebGLRenderer info:
render
calls: 271
vertices: 1926558
faces: 33888
points: 0
memory
geometries: 483
textures: 0
Close


Firefox
48.0.2 What's new
Restart Firefox to Update
Firefox is designed by Mozilla, a global community working together to keep the Web open, public and accessible to all.
Want to help? Make a donation or get involved!


About
Google Chrome
A web browser built for speed, simplicity, and security
Get help with using Chrome
Report an issue
Version 53.0.2785.116 (64-bit)
Google Chrome is up to date.
Set Up Automatic Updates for All Users


OS X
Version 10.9.5
Software Update...
Processor 2.5 GHz Intel Core i7
Memory 16 GB 1600 MHz DDR3


Built-in Retina Display
15-inch (2880 x 1800)
NVIDIA GeForce GT 750M 2048 MB graphics



http://caniuse.com/#search=WebGL2