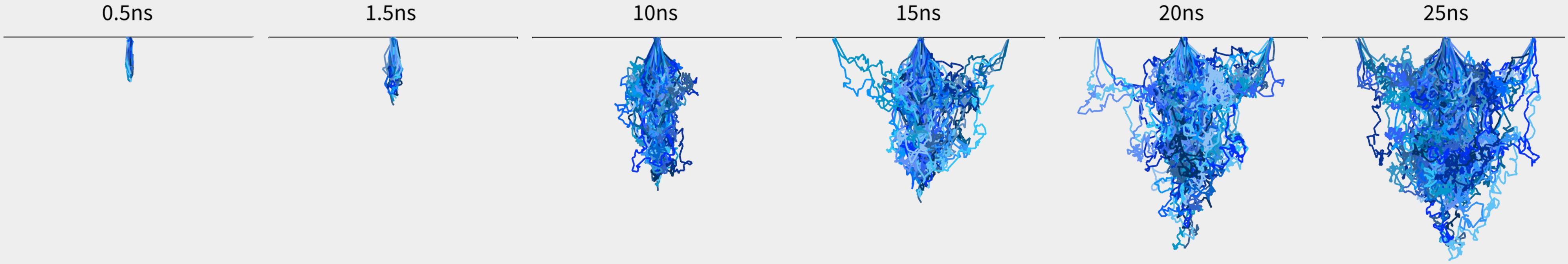
ap²
allpix squared

cern.ch/allpix-squared




# Allpix Squared 2.0 & Onwards

## Recent & Ongoing Developments for Semiconductor MC Simulations


**Simon Spannagel, DESY**

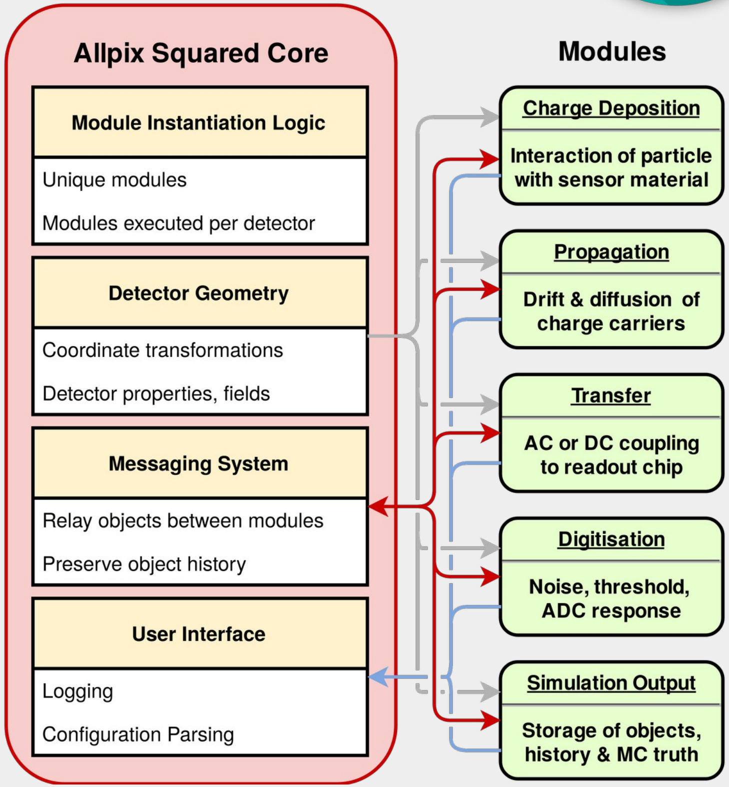

10th Beam Telescopes & Test Beams Workshop

21 June 2022

Lecce, for real this time

# Introduction: The Allpix$^2$ Framework

- Modular Semiconductor Pixel Detector MC simulation software
  - Microscopic: simulation of charge carrier motion
  - Extensible: single detectors / full systems
  - Configurable: no coding knowledge needed



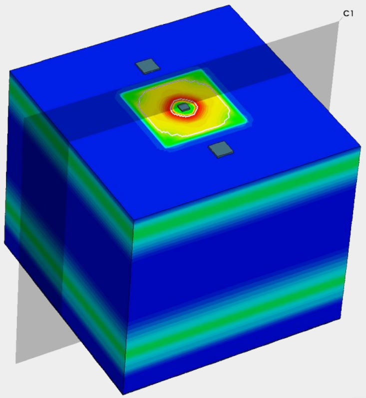

- Main development guidelines:
  - **Integration of Existing Toolkits**
  - **Well-Tested & Validated Algorithms**
  - **Low Entry Barrier for New Users**
  - **Clean & Maintainable Code**

# Integration of Existing Toolkits

- Many very powerful tools developed and employed over decades of detector R&D
- Leverage their capabilities by providing interfaces for their integration

- **Geant4** – simulating energy deposition of particles passing through matter
  - Extensive toolkit, detailed simulation of many interactions & processes (e.g. decays)
  - Cumbersome to use for beginners, complexity often overwhelming at first
  - Provide abstraction layer that auto-generates models and calls Geant4 kernel

- **TCAD** – solving Poisson’s equation using doping information
  - Detailed understanding of field configuration, sensor behavior
  - Tools & knowledge widely spread in community
  - Provide possibility to import results to complement MC simulations



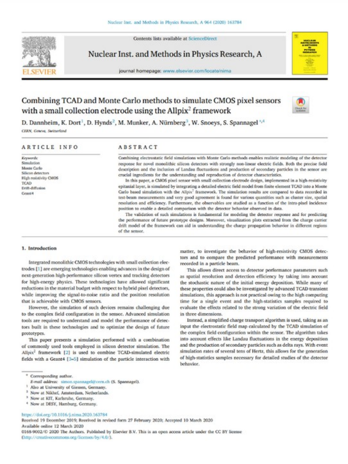
# Well-Tested & Validated Algorithms

- Simulations provide insights into physical processes – but only if they model them correctly!
  Validation of algorithms crucial and time-consuming process

- With Allpix Squared, we strive for
  - Validating as much as possible against known data
  - Publishing reference studies including full simulation configuration used
  - Providing automated test for every new feature


Nuclear Inst. and Methods in Physics Research, A
Allpix²: A modular simulation framework for silicon detectors

NIMA 901 (2018) 164 – 172
doi:10.1016/j.nima.2018.06.020

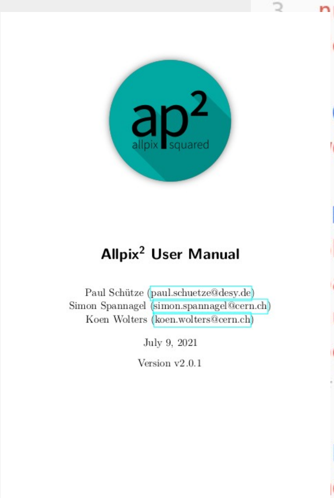

Nuclear Inst. and Methods in Physics Research, A
Combining TCAD and Monte Carlo methods to simulate CMOS pixel sensors with a small collection electrode using the Allpix² framework

NIMA 964 (2020) 163784
doi:10.1016/j.nima.2020.163784

In preparation…

# Low Entry Barrier for New Users

- Simulation frameworks often very complex:
  code complexity, lack of documentation, physics
- Allpix Squared attempts to facilitate quick starts:
  - Extensive documentation / user manual / help forum
  - Human-readable configuration files
  - Support for physical units
  - No coding or code-reading required
- Successfully used e.g. in university
  education, summer schools, …

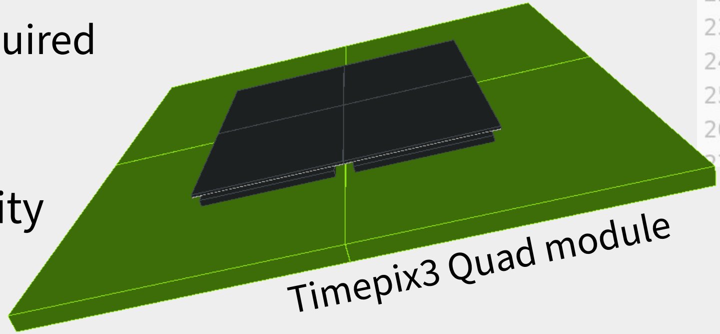

Allpix$^2$ User Manual

Paul Schütze (paul.schuetze@desy.de)
Simon Spannagel (simon.spannagel@cern.ch)
Koen Wolters (koen.wolters@cern.ch)

July 9, 2021

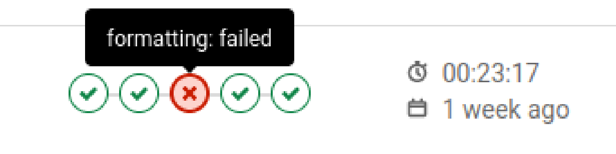
Version v2.0.1

```
[AllPix]
log_level = "INFO"
number_of_events = 500000
etectors_file = "telescope.conf"

GeometryBuilderGeant4]
orld_material = "air"

DepositionGeant4]
hysics_list = FTFP_BERT_LIV
article_type = "Pi+"
umber_of_particles = 1
eam_energy = 120GeV
 ...

ElectricFieldReader]
odel="linear"
bias_voltage=150V
depletion_voltage=50V

[GenericPropagation]
temperature = 293K
charge_per_step = 10
spatial_precision = 0.0025um
timestep_max = 0.5ns

[SimpleTransfer]
```

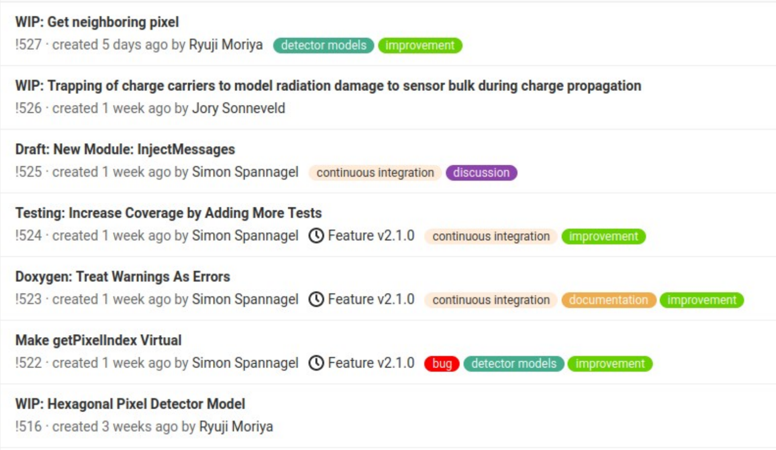

Timepix3 Quad module

# Clean & Maintainable Code

- Collaborative software development requires well-defined procedures – Otherwise quickly becomes unmaintainable

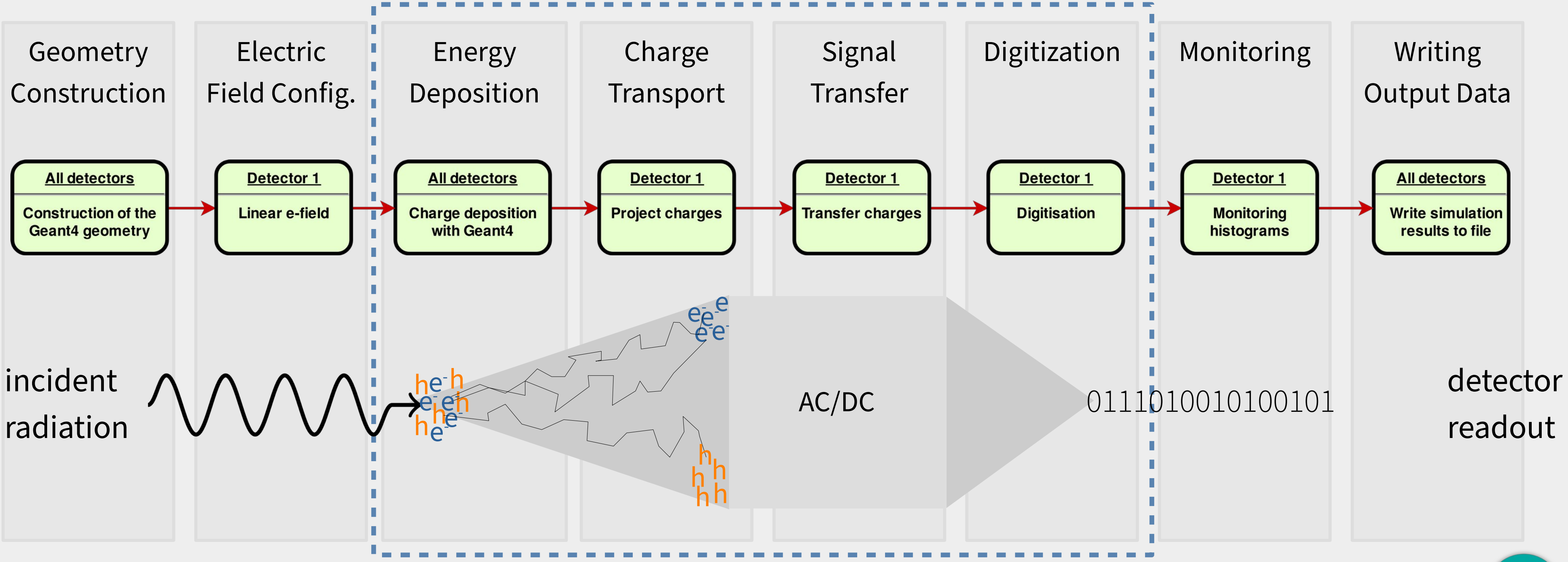



- Allpix Squared implements *best practices* for software development
  - Permissive open-source license: MIT
  - Extensive code reviews via merge requests
  - Strict enforcement of coding conventions & formatting
  - Regular static code analysis



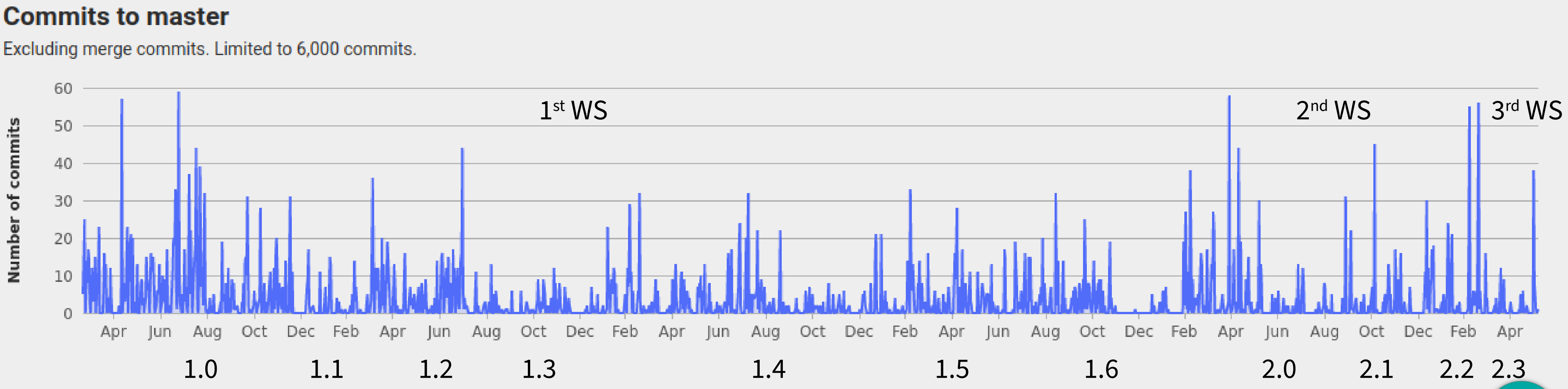
# Introduction: The Simulation Chain

- Building blocks follow individual steps of signal formation in detector
- Algorithms for each step can be chosen independently



# A Brief History of Allpix Squared

- Started end of 2016 at CERN in EP-LCG group, now main development at DESY
- Steady development – 44 releases by now
- Many applications in different fields, 49 contributors, 3 user workshops



# Recent Releases 2.0 – 2.3

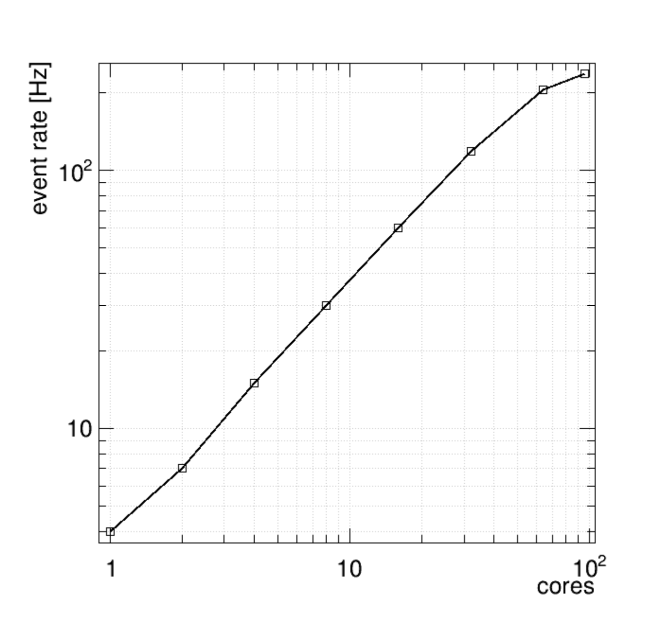
and new Features

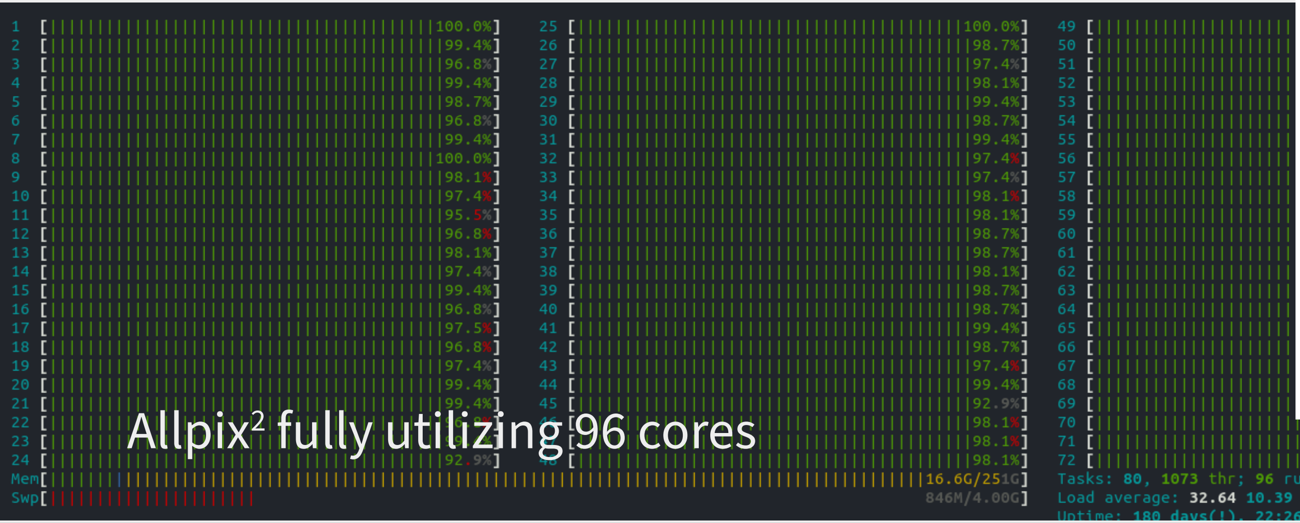
# Allpix Squared Releases 2.0 – 2.3

- 2.0: First major release introducing structural changes to framework since 1.0 (08/2017)
  - More than 1500 commits over previous feature release 1.6
  - Introduced fully parallel event processing (Started as Google Summer of Code project)
  - Further separation between physics models & algorithms
  - https://cern.ch/allpix-squared/post/2021-06-15-version-2.0.0/
- 2.1: Focus on smoothing out multithreading in some modules
  https://cern.ch/allpix-squared/post/2021-11-17-allpix-squared-2.1-released/
- 2.2: New features extending configurability & passive materials
  https://cern.ch/allpix-squared/post/2022-02-28-allpix-squared-2.2-released/
- 2.3: Introduction of multiple semiconductor sensor materials
  https://cern.ch/allpix-squared/post/2022-05-16-allpix-squared-2.3-released/

# Event-Based Seeding & Multithreading

- Efficient use of system resources / multiple cores
- Retaining strong reproducibility: exact same result, independent of # workers,
- Fully transparent to user / simulation

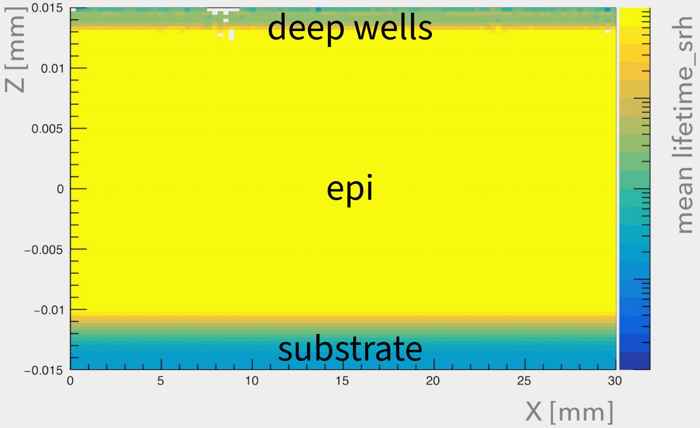



Allpix² fully utilizing 96 cores

# Recombination of Charge Carriers

K. Dort




- In many applications: fast signal formation
  no need for recombination – all e/h pairs reach electrodes
- Sometimes, finite charge carrier lifetime becomes interesting:
  - High-dopant regions
  - Low electric fields, signal formation via diffusion

- Allpix Squared supports position-dependent doping maps & lifetime calculation
  - Shockley-Read-Hall recombination: medium doping concentrations
  - Auger recombination: high doping concentrations
  - Combination

$$\tau^{-1}(N_d) = \begin{cases} \tau_{srh}^{-1}(N_d) + \tau_a^{-1}(N_d) & (minority) \\ \tau_{srh}^{-1}(N_d) & (majority) \end{cases}$$

- Application example: monolithic active pixel sensors

# Different Carrier Mobility Models

- Providing different charge carrier mobility models
  - Field dependent
  - Doping concentration dependent
  - Optimized for high-field situations
  - …
  - ***Custom models*** defined via configuration

```
[GenericPropagation]
temperature = 293K
mobility_model = "masetti"
```

$$\mu(E) = \frac{v_m}{E_c} \frac{1}{\left(1 + (E/E_c)^{\beta}\right)^{1/\beta}},$$

**Jacoboni/Canali**

- Description & reference provided in user manual
- Selected via configuration file

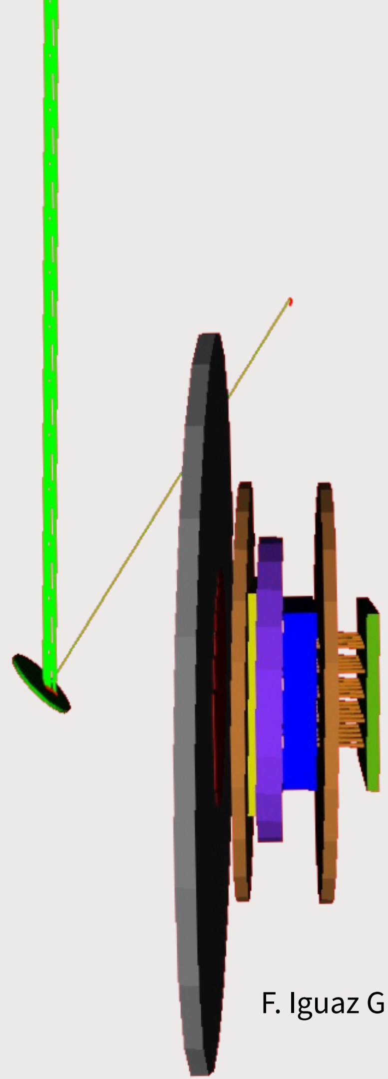
```
# Replicating the Jacoboni-Canali mobility model at T = 293K
mobility_model = "custom"

mobility_function_electrons = "[0]/[1]/pow(1.0+pow(x/[1],[2]),1.0/[2])"
mobility_parameters_electrons = 1.0927393e7cm/s, 6729.24V/cm, 1.0916

mobility_function_holes = "[0]/[1]/pow(1.0+pow(x/[1],[2]),1.0/[2])"
mobility_parameters_holes = 8.447804e6cm/s, 17288.57V/cm, 1.2081
```

# New Module: DepositionCosmics

- Simulation of cosmic rays with realistic particle and energy composition
- Utilizes the CRY framework
  - Database of cosmic ray composition and spectrum depending on altitude, latitude and date
  - Interfaces to Geant4
  - Inherits from DepositionGeant4 for sensor handling and energy deposition
- Applications: cosmic ray telescopes

# Import of GDML Geometries

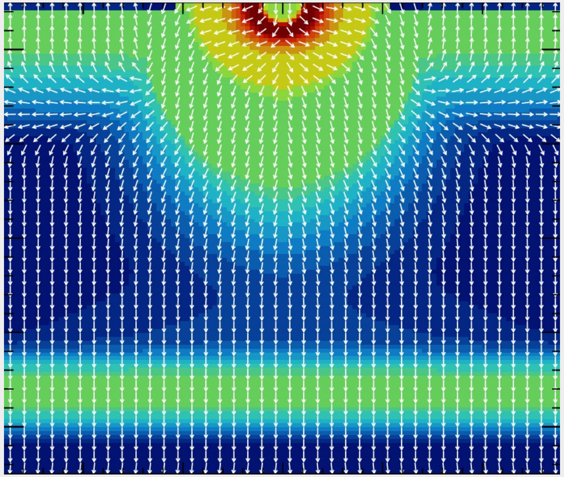
- GDML: Geometry Description Markup Language
  - Library of basic geometrical shapes
  - XML formatted geometry description: shape, dimensions, positioning and orientation
  - Features volume subtractions and mother/daughter volumes
- GDML imports are treated as passive volumes only
- Application examples: phantom definition for imaging, import of CAD models via GDML



F. Iguaz Gutierrez

# Custom Analytical Electric Field Functions

- From beginning on, implementations for
  - Linear electric fields → "standard" planar sensor
  - TCAD field maps → complex "known" sensors


A. Santra

- Often other models are required, now added:
  - Parabolic shape → double-peaked electric field after irradiation
  - Possibility for custom analytical field functions → approximations of complex sensors

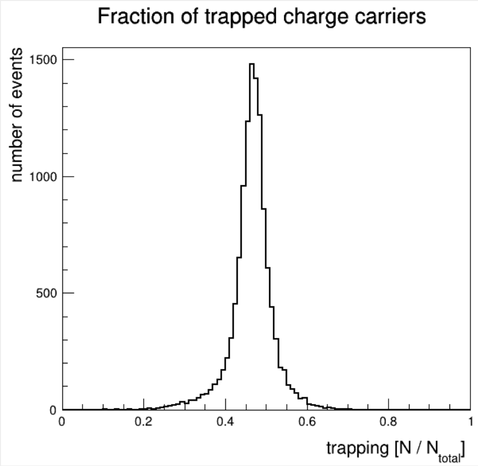
```
[ElectricFieldReader]
model = "custom"
field_function = "[0]*(x*x+y*y)"
field_parameters = 12500V/cm/cm
```

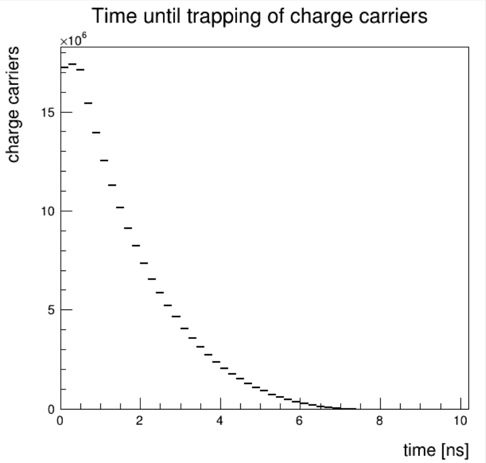
- Application example: approximating MAPS field

# Charge Carrier Trapping

- Simulation of radiation-induced trapping of charge carriers
- Different trapping models implemented
  - Ljubljana / Kramberger
  - Dortmund / Krasel
  - Interpolation of CMS Tracker measurements
  - Mandic / high fluences

- Possibility to define custom trapping functions via configuration
- Scaling with fluence & temperature (where applicable)
- Note: this only describes trapping!
  Effects such as changed electric fields have to be provided separately, either through field map from TCAD or analytic approximation of E-field



# Other Semiconductor Sensor Materials

- The Allpix Squared ~~Silicon~~ Semiconductor Detector Simulation Framework now allows for the definition of other sensor materials than silicon
- Definition of sensor materials impacts …
  - Material in Geant4 geometry
  - Charge carrier creation energy default
  - Fano factor default
- Short list of supported materials

→ New materials can easily be added by users (see FAQs in manual)

Table 6.1: List of default sensor material properties implemented in $\text{Allpix}^2$

| **Material** | **Charge Creation Energy** [eV] | **Fano factor** | **Sources** |
|---|---|---|---|
| Silicon | 3.64 | 0.115 | [25], [26] |
| Gallium Arsenide | 4.2 | 0.14 | [27] |
| Cadmium Telluride | 4.43 | 0.24 | [28], [29] |
| Cadmium Zinc Telluride $Cd_{0.8}Zn_{0.2}Te$ | 4.6 | 0.14 | [30], [31] |
| Diamond | 13.1 | 0.382 | [32], [32] |
| Silicon Carbide (4H-SiC) | 7.6 | 0.1 | [33], [34] |

# **Ongoing Projects**
and Developments

# Impact Ionization

Implementation of charge multiplication through impact ionization underway

- Multiple models available, selection via configuration file:
  - Massey
  - van Overstraeten-de Man
  - Okuto-Crowell
  - Bologna
- Fully documented in user manual
- **Implementation in Allpix$^2$ completed**, undergoing testing,
  Comparison with Weightfield2 & TCAD simulations

coefficient $\alpha$ and the length of the step $l$ performed in the respective electric field. If the electric field strength stays below a configurable threshold $E_{\text{thr}}$, unity gain is assumed:

$$g(E,T) = \begin{cases} e^{l\cdot\alpha(E,T)} & E > E_{\text{thr}} \\ 1.0 & E < E_{\text{thr}} \end{cases} \qquad (6.12)$$

The the following impact ionization models are available:

### 6.3.1 Massey Model

The Massey model [35] describes impact ionization as a function of the electric field $E$. The ionization coefficients are parametrized as

$$\alpha(E,T) = Ae^{-\frac{B(T)}{E}}, \qquad (6.13)$$

where $A$ and $B(T)$ are phenomenological parameters, defined for electrons and holes respectively. While $A$ is assumed to be temperature-independent, parameter $B$ exhibits a temperature dependence and is defined as

$$B(T) = C + D \cdot T. \qquad (6.14)$$

The parameter values implemented in Allpix$^2$ are taken from Section 3 of [35] as:

$$A_e = 4.43 \times 10^5\,/\text{cm} \qquad A_h = 1.13 \times 10^6\,/\text{cm}$$
$$C_e = 9.66 \times 10^5\,\text{V/cm} \qquad C_h = 1.71 \times 10^6\,\text{V/cm}$$
$$D_e = 4.99 \times 10^2\,\text{V/cm/K} \qquad D_h = 1.09 \times 10^3\,\text{V/cm/K}$$

for electrons and holes, respectively.

This model can be selected in the configuration file via the parameter `multiplication_model = "massey"`.

### 6.3.2 Van Overstraeten-De Man Model

The Van Overstraeten-De Man model [36] describes impact ionization using Chynoweth's law, given by

$$\alpha(E,T) = \gamma(T) \cdot a_\infty \cdot e^{-\frac{\gamma(T)\cdot b}{E}}, \qquad (6.15)$$

For holes, two sets of impact ionization parameters $p = \{a_\infty, b\}$ are used depending on the electric field:

$$p = \begin{cases} p_{\text{low}} & E < E_0 \\ p_{\text{high}} & E > E_0 \end{cases} \qquad (6.16)$$

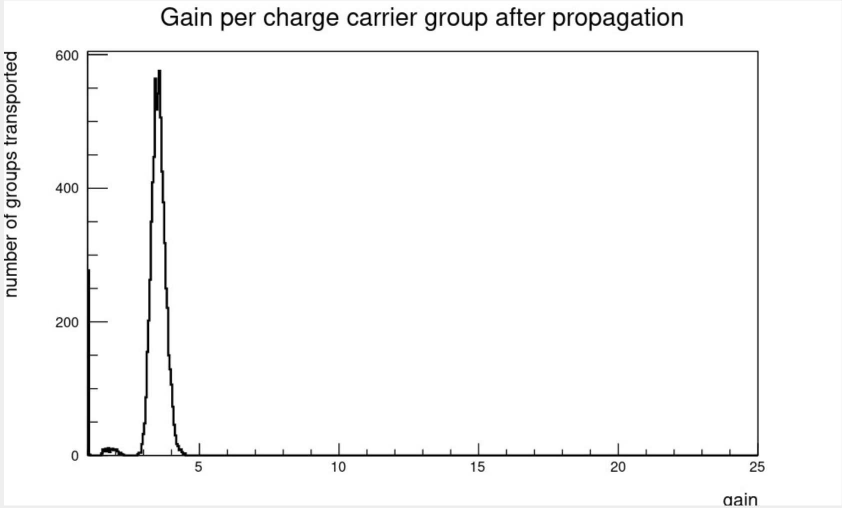
Temperature scaling of the ionization coefficient is performed via the $\gamma(T)$ parameter following the Synposys Sentaurus TCAD user manual as:

$$\gamma(T) = \tanh\left(\frac{0.063 \times 10^6\,\text{eV}}{28.6173 \times 10^{-5}\,\text{eV/K} \cdot T_0}\right) \cdot \tanh\left(\frac{0.063 \times 10^6\,\text{eV}}{28.6173 \times 10^{-5}\,\text{eV/K} \cdot T}\right)^{-1} \qquad (6.17)$$

# Impact Ionization – Example

- Enable/select via configuration file:
  - Multiplication model
  - Electric field strength threshold field for multiplication

```
[GenericPropagation]
temperature = 293K
multiplication_model = "massey"
multiplication_threshold = 100kV/cm
```



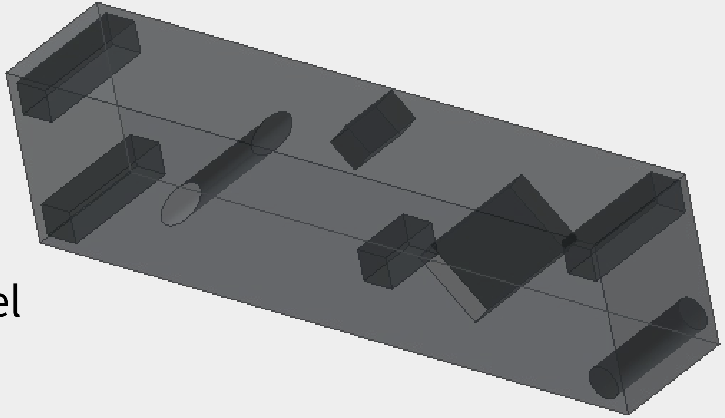

- Automatic check of propagation parameters (time stepping, …)

```
(WARNING) [I:GenericPropagation:lgad] Charge multiplication enabled with maximum timestep larger than 1ps
                                      This might lead to unphysical gain values.
```

- Status MR !472

# Simulation of 3D Sensors

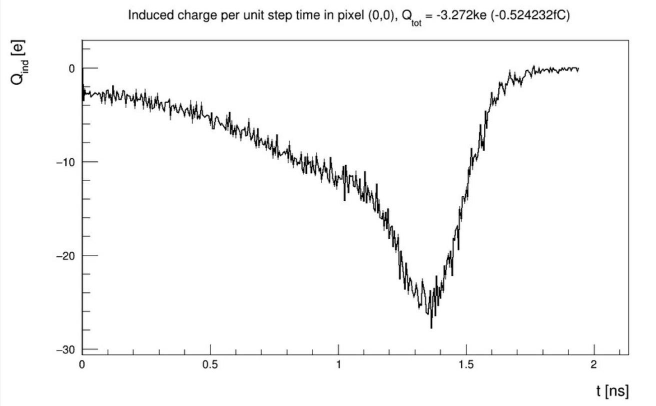
- Definition of per-pixel implants via detector model file
  - Position with respect to pixel center
  - Shape & orientation
  - Material
  - front/backside
- Implants are repeated for each pixel
- Add as many implants as required, syntax similar to support layers (PCB etc)
- Requires matching electric field map
- Proper collision detection algorithms of charge carriers with implant volumes; motion stops immediately at implant border



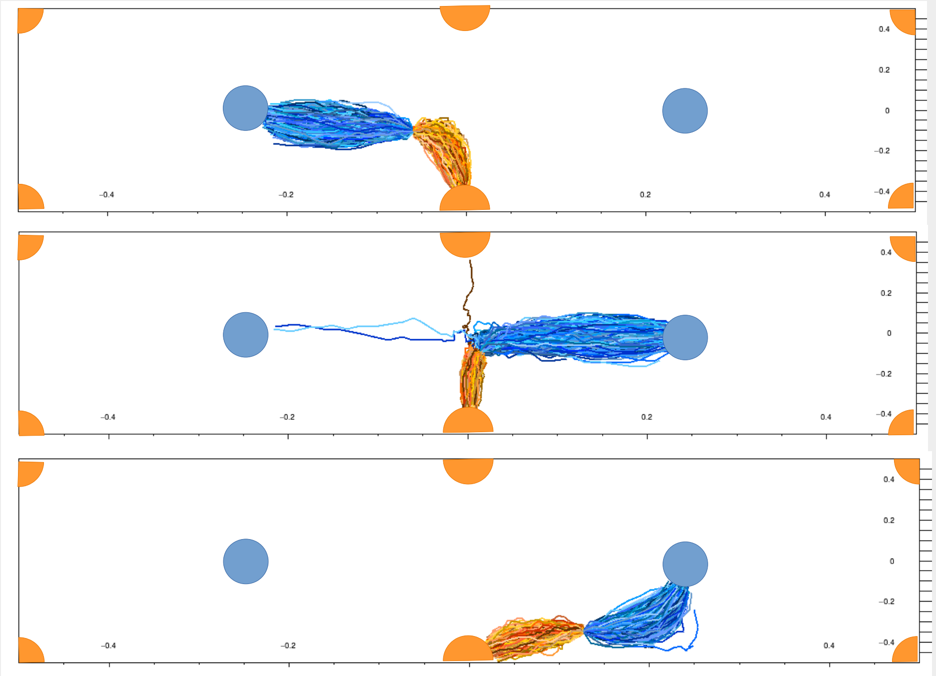
```
type = "monolithic"

number_of_pixels = 20 20
pixel_size = 250um 50um

sensor_thickness = 50um
sensor_excess = 10um

[implant]
type = frontside
shape = ellipse
size = 13um 13um 50um
offset = 62.5um 0
material = silicon

[implant]
type = frontside
shape = ellipse
size = 13um 13um 50um
offset = -62.5um 0
material = silicon

[implant]
type = backside
shape = ellipse
size = 13um 13um 49um
offset = -125um -25um
material = aluminum
```

# Simulation of 3D Sensors – Example

- First simulations with ATLAS 3D sensor geometry
  - Two central front-side columns (collect charge)
  - Six Ohmic backside contact columns
- Charge collection / sharing works as expected
- Transient simulations produce pulses
- Status: MR !672



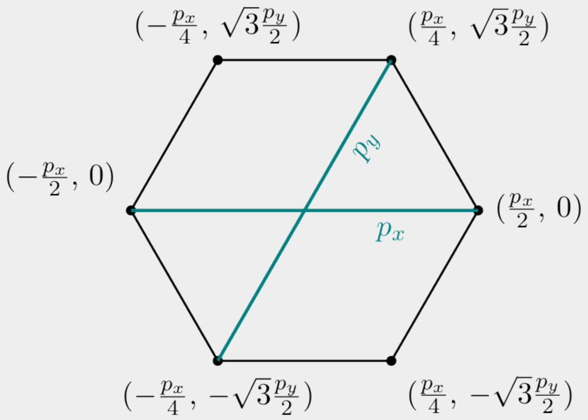

single pixel,
top view

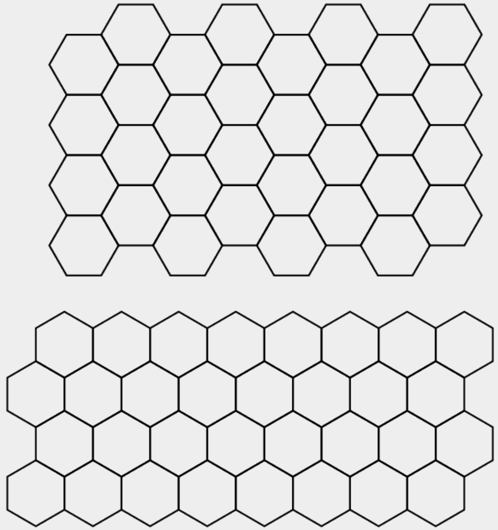


# Hexagonal Pixel Geometries

Extension of Allpix² geometry subsystem to enable simulation of different pixel shapes & matrix arrangements

- Hexagonal geometry interesting for many applications
  - Avoid problematic field regions in corners
    (small electrodes: low fields, large electrodes: high fields)
  - Symmetry more close to circle – more uniform response
- **Implementation in Allpix² completed**, undergoing testing:
  - Using **axial coordinate system**
  - Support for "pointy" & "flat" hexagon orientation,
    regular (same-pitch) and distorted (different pitch) hexagons
- Used already by several groups

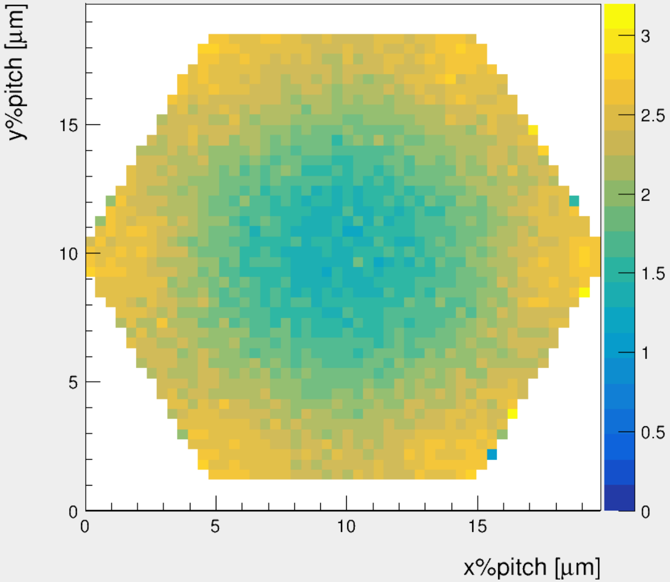


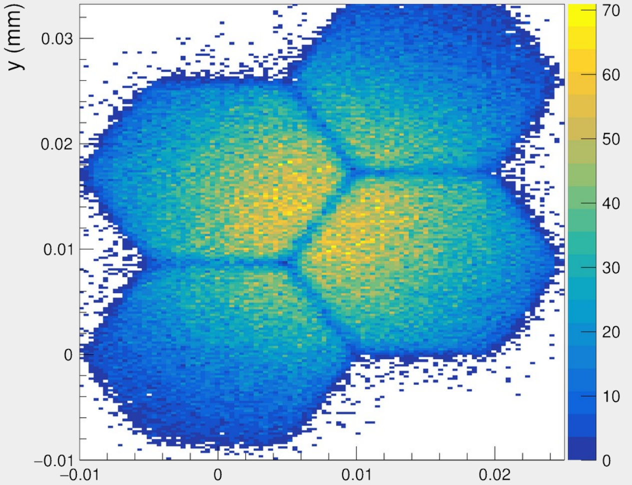



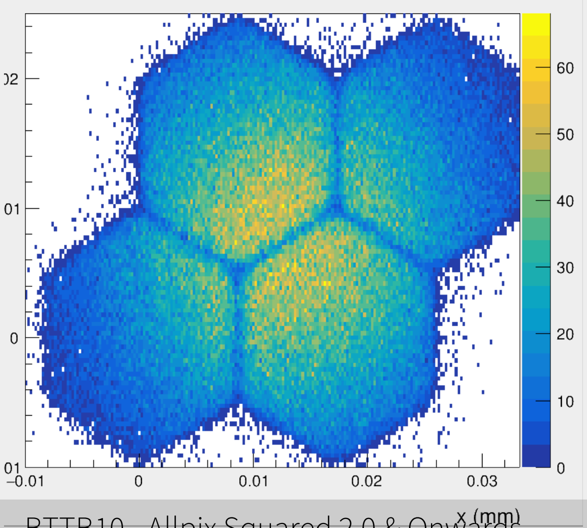
Other geometries already merged (e.g. radial strips @ ATLAS Itk)

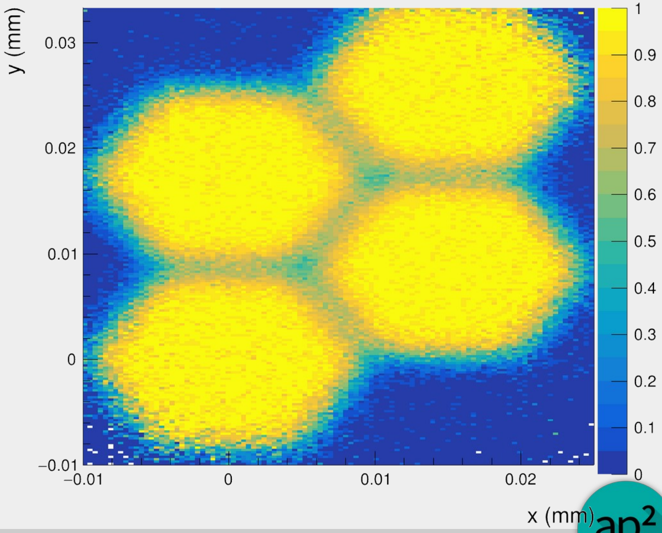
# Some Nice Plots – with Hexagonal Pixels

- Performed implementation tests with hexagonal pixel detector
  large pixels, linear electric field, high charge threshold
- Analysis of basic properties
  - Recovery of radial beam profile from pixel matrix
  - In-pixel cluster size distribution
  - Efficiency fall-off at hexagon edges



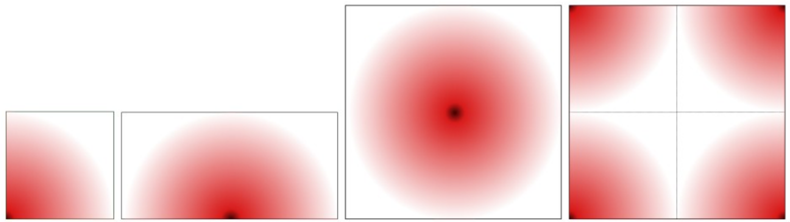



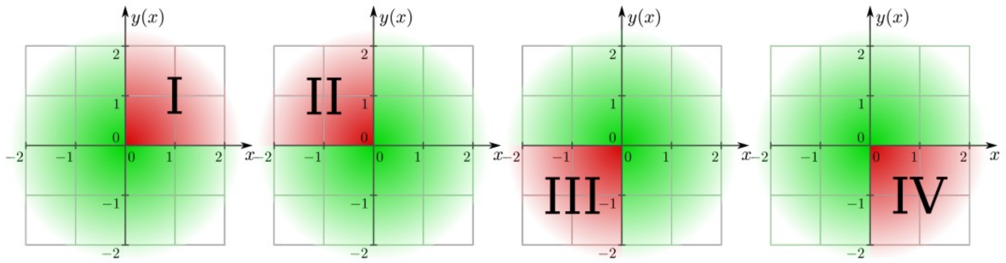



# Rework Loading Electric Field Maps

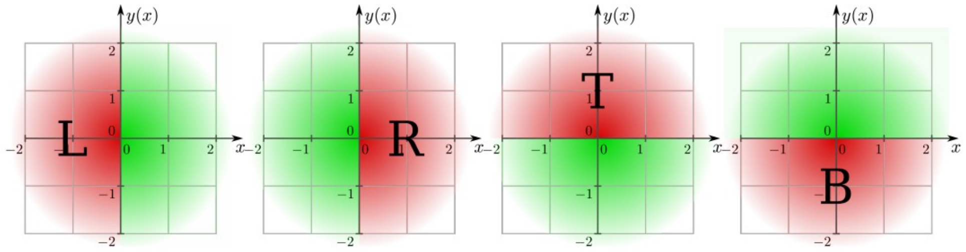
- Current loading of field maps is very flexible…
  …but only works for rectangular geometries!
- Started implementation of new mapping of fields to pixels
  per-pixel lookup of fields
- Has some advantages (lookup possible for any pixel shape)
  and some disadvantages ( not possible to provide multi-pixel fields)
- Final solution likely will keep current mapping (as default)
  with new possibility (full, half, quarter pixels)



(a) Quarter pixel (b) Half pixel (c) Full pixel (d) Four quarter pixels




(a) Quadrant I (b) Quadrant II (c) Quadrant III (d) Quadrant IV




(a) Left Half (b) Right Half (c) Top Half (d) Bottom Half

# In a nutshell…

# Summary & Outlook

**Allpix Squared**

- Flexible semiconductor Monte Carlo simulation framework ideas
- Steady development over many years, large & growing user base in many fields

**Recent Releases**

- Released one major (2.0) & three subsequent feature (2.1, 2.2, 2.3) versions since last BTTB
- Many new features, continuously extending reach of simulations:
  Multithreading, GDML objects, Recombination, different Mobility Models, Cosmics, Trapping, other Semiconductors, …

**Upcoming Major Release**

- We are working already towards the next major release 3.0
- Main ingredients: reworked geometry framework to allow for e.g. hexagons, polar coordinates etc
- Revamped website & documentation coming!

# Allpix Squared Resources

Website

https://cern.ch/allpix-squared

Repository

https://gitlab.cern.ch/allpix-squared/allpix-squared

Docker Images

https://gitlab.cern.ch/allpix-squared/allpix-squared/container_registry

User Forum:

https://cern.ch/allpix-squared-forum/

Mailing Lists:

allpix-squared-users https://e-groups.cern.ch/e-groups/Egroup.do?egroupId=10262858

allpix-squared-developers https://e-groups.cern.ch/e-groups/Egroup.do?egroupId=10273730

User Manual:

https://cern.ch/allpix-squared/usermanual/allpix-manual.pdf

DESY.