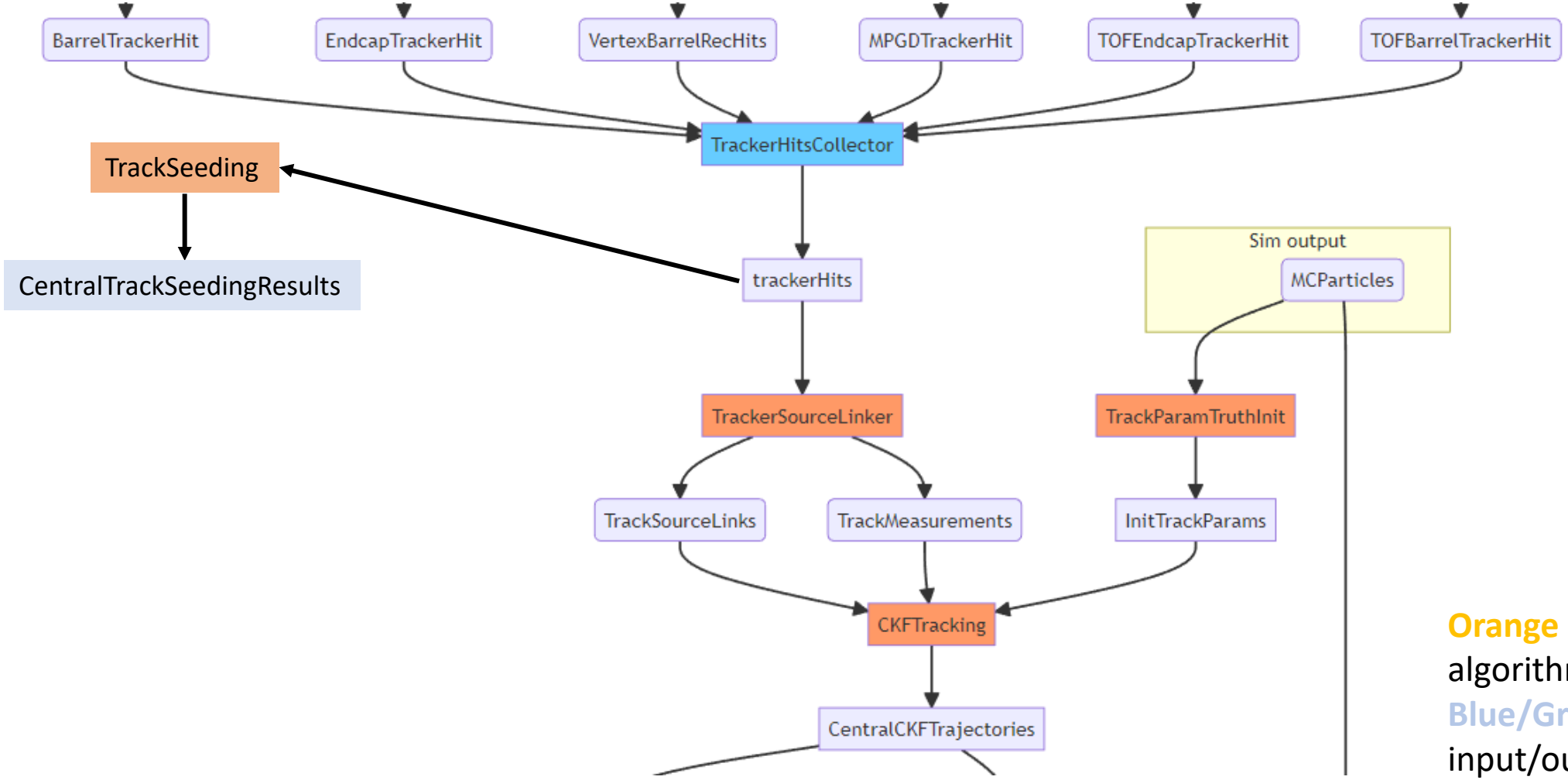
# How to use realistic seeding as input for CKF in EICRecon

Barak Schmookler

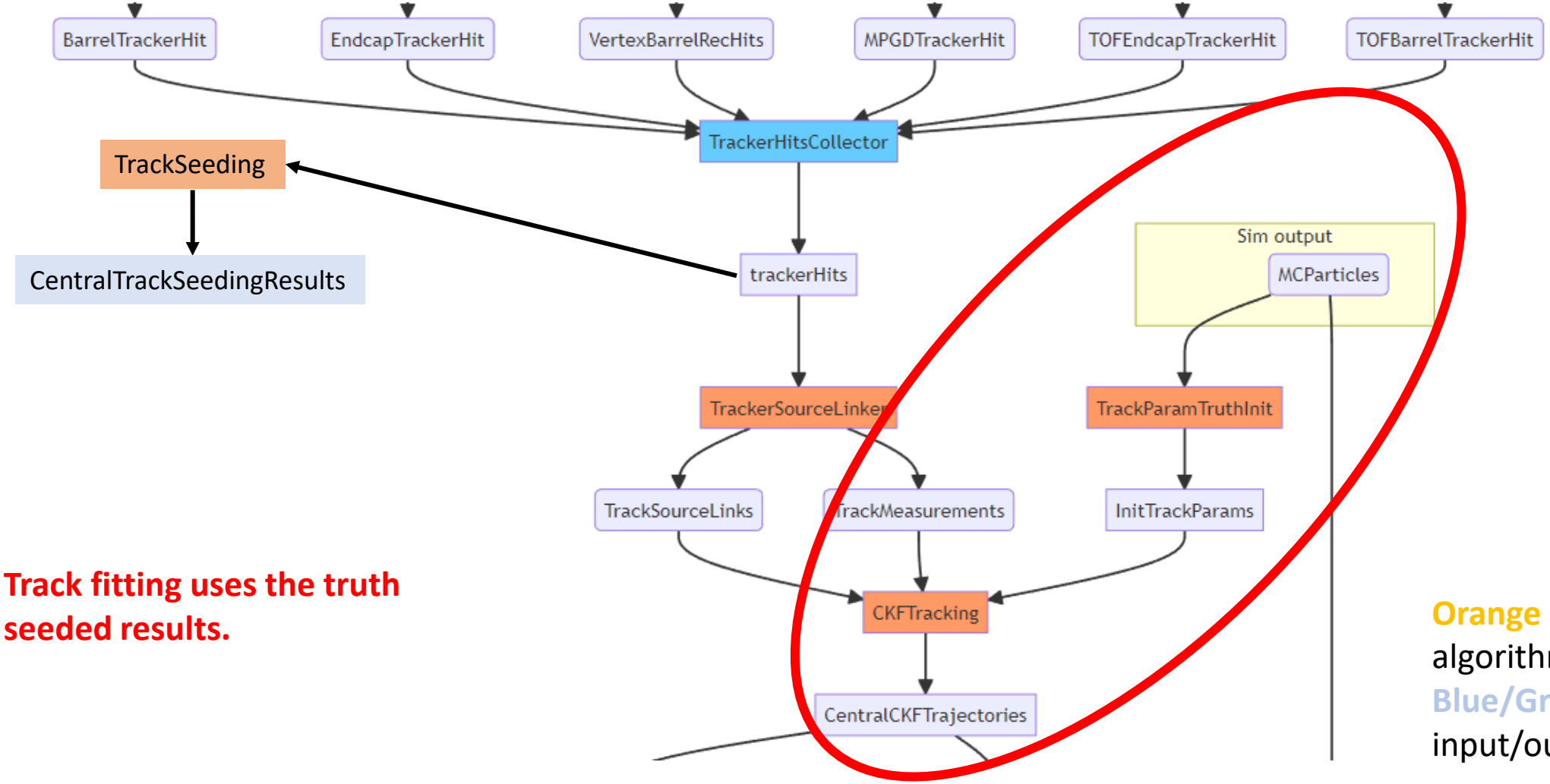
# Tracking logic in EICRecon




**Orange** boxes are algorithms.
**Blue/Grey** boxes are input/output data.

https://eic.github.io/EICrecon/#/design/tracking

# Tracking logic in EICRecon

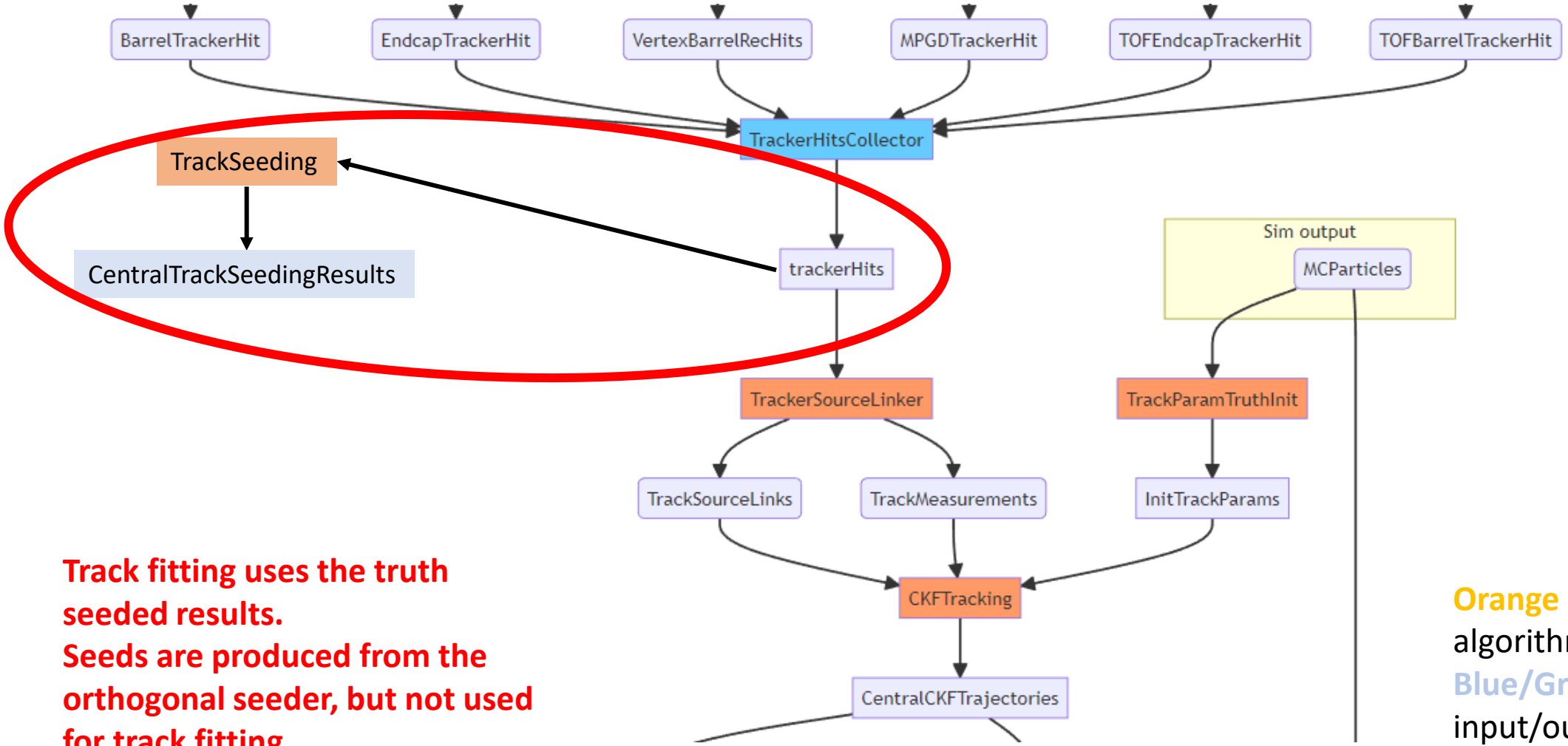



**Track fitting uses the truth seeded results.**

**Orange** boxes are algorithms.
**Blue/Grey** boxes are input/output data.

https://eic.github.io/EICrecon/#/design/tracking

# Tracking logic in EICRecon

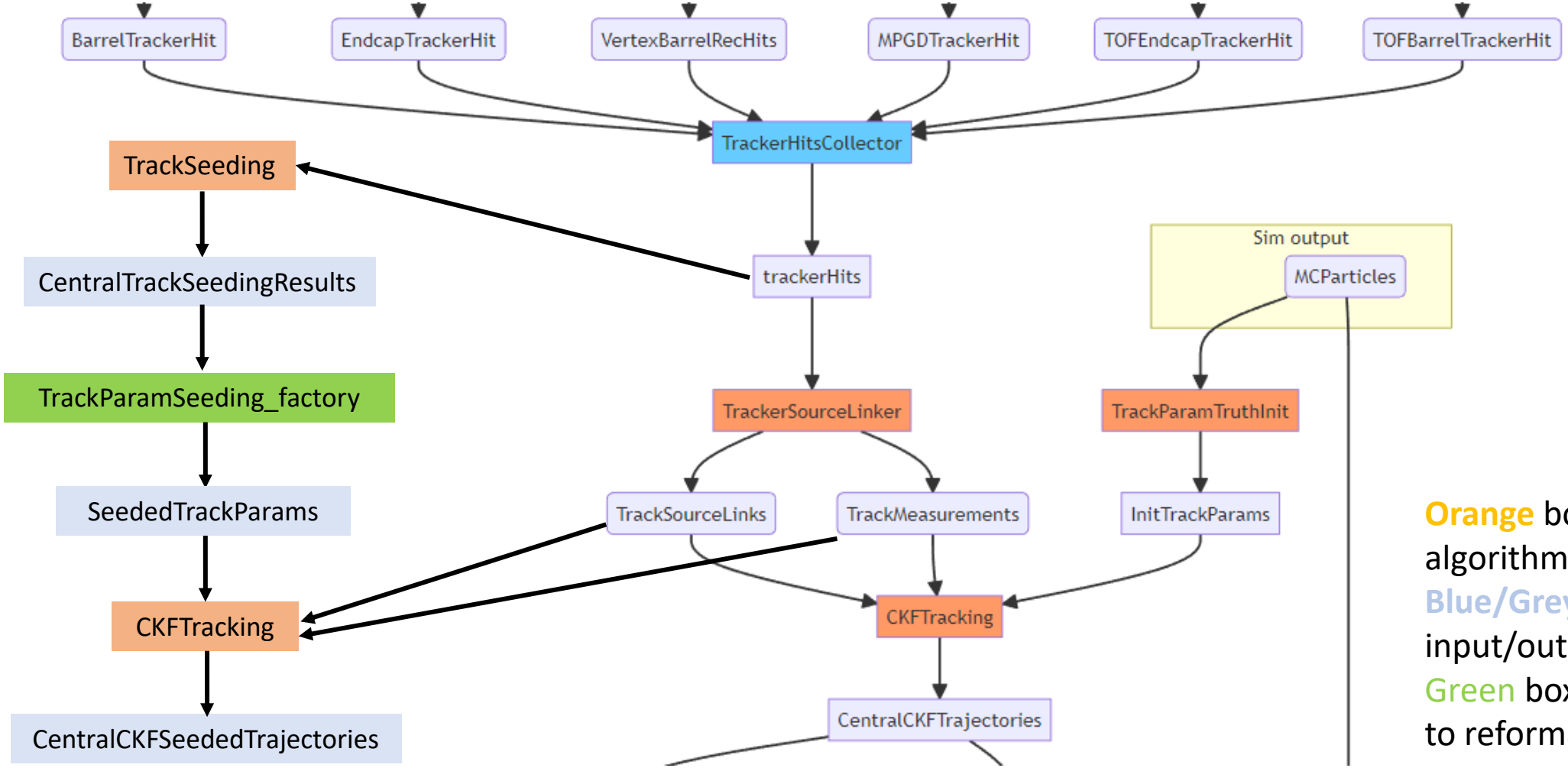



**Track fitting uses the truth seeded results.**
**Seeds are produced from the orthogonal seeder, but not used for track fitting.**

**Orange** boxes are algorithms.
**Blue/Grey** boxes are input/output data.

https://eic.github.io/EICrecon/#/design/tracking

# Update to use real seed for tracking




**Orange** boxes are algorithms.
**Blue/Grey** boxes are input/output data.
**Green** box is a factory to reformat data type.

https://eic.github.io/EICrecon/#/design/tracking

# How this can be used

- The code lives in our track-QA branch:

  https://github.com/eic/EICrecon/tree/track-qa-barak

- The new datatypes and factories shown above do not affect the previous workflow. So, they can be merged into the main branch without causing any changes to the standard output ROOT file.
- A user can access the tracks which use the realistic seeding by using the following in a Plugin:

```
auto trajectories = event->Get<eicrecon::TrackingResultTrajectory>("CentralCKFSeededTrajectories");
```

instead of (for truth seeded tracks):

```
auto trajectories = event->Get<eicrecon::TrackingResultTrajectory>("CentralCKFTrajectories");
```

# Summary

- We have added code in EICRecon which allows track fitting using the orthogonal seeder results.
- The output of this track fitting can be used in an EICRecon Plugin. The new code will have no effect on the normal output ROOT file which is created by EICRecon.
- Is creating a parallel structure the best way to implement the realistic seeding CFK option?