# MAUS - Bug #1852

## Duplicate primaries in G4BL input

01 June 2016 16:21 - Rogers, Chris

| | | | |
|---|---|---|---|
| **Status:** | Open | **Start date:** | 01 June 2016 |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | Drielsma, François | **% Done:** | 0% |
| **Category:** | Simulation | **Estimated time:** | 0.00 hour |
| **Target version:** | Future MAUS release | | |
| **Workflow:** | New Issue | | |

**Description**

I notice that it seems rather easy to generate duplicate primaries in G4BL input deck.

I attach a couple of sample data sets and a script that finds duplicates and prints them. Is this intentional behaviour? It seems dangerous to me...

### History

**#1 - 02 June 2016 11:02 - Rajaram, Durga**

John -- any ideas on this?

**#2 - 06 June 2016 17:35 - Rogers, Chris**

*- File random_seed.patch added*

Potential patch attached. Nb: I note that most of the tests in test_mappybeamlinesimulation are failing right now. I put in a mockup CallG4bl thing which one could extend to make them pass... but did not dig too hard (too much else to dig into right now).

**#3 - 07 June 2016 09:51 - Rogers, Chris**

Its not quite right... need to be able to handle multiple input random seeds (for running several input jobs)... will try again today

**#4 - 07 June 2016 15:21 - Rogers, Chris**

I noticed that CallG4bl has a bug that the random_seed of generated primaries (which is used to seed geant4) is the same for every event in a given g4bl output file. It should be unique for each event. Instead of using just the g4bl random seed, I modified it to use g4bl random seed + line number (i.e. position in the g4bl file).

**#5 - 07 June 2016 16:46 - Rogers, Chris**

*- File random_seed.patch_2 added*

*- File test_simulate_beam.tar.gz added*

Another issue - the json cpp -> ROOT/C++ conversion can't handle more than 32 bit integers. It is possible to fix without too much hassle. For now I only permit 32bit random seeds (note this is 2 billion random seeds allowed, which means non-zero probability of a random seed collision).

So it looks like the latest version works - attached integration test in test_simulation_beam.tar.gz (along with unit tests in the random_seed.patch_2 which was a hack of stuff in duplicate_primaries.tar.gz).

**#6 - 07 June 2016 16:53 - Mohayai, Tanaz Angelina**

*- File primary.log added*

Hi Chris, I remember seeing similar duplicate primaries in my G4beamline input deck. This was when I was using MAUS v0.9.7; if I recall correctly, the issue was resolved when I switched to v1.4.0, which was the latest version at the time. I have also ran a recent simulation with v2.4.0 and according to the attached primary.log file (from your primary.py script), the duplicate primaries seem to be no longer present.

**#7 - 11 August 2016 13:04 - Dobbs, Adam**

Chris, I have applied the patch files to one of my local branches - are you happy for it to go into the trunk?

**#8 - 11 August 2016 13:05 - Dobbs, Adam**

Chris, I have applied the patch files to one of my local branches - are you happy for it to go into the trunk?

**#9 - 11 August 2016 13:11 - Rogers, Chris**

Yes, certainly!

### Files

| | | | |
|---|---|---|---|
| duplicate_primaries.tar.gz | 6.16 MB | 01 June 2016 | Rogers, Chris |
| random_seed.patch | 7.62 KB | 06 June 2016 | Rogers, Chris |
| random_seed.patch_2 | 12.9 KB | 07 June 2016 | Rogers, Chris |
| test_simulate_beam.tar.gz | 347 KB | 07 June 2016 | Rogers, Chris |
| primary.log | 76 KB | 07 June 2016 | Mohayai, Tanaz Angelina |