

Name Manager & co

Reco classes

(BaseReco, RecoRaw & RecoMC)

- Change checks
 - Global checks done in TAGrecoManger
 - Campaign checks done in TAGcampaignManager
 - Run condition checks done in TAGrunManager
 - ➡ Calling of checks only in BaseReco now
 - Change SetRawHistogramDir/SetRecHistogramDir methods
 - Loop over all required actions regardless MC or Raw actions
 - ➡ Calling one method SetHistogramDir in BaseReco now
 - Change AddRecRequiredItem/AddRawRequiredItem methods
 - Loop over all actions regardless MC or Raw actions
 - ➡ Calling one method AddRequiredItem in BaseReco now
- ➡ Saving ~ 400 lines in BaseReco, RecoRaw and RecoMC

Introduction

- Number of names:
 - ~20 raw actions & ~10 MC actions
 - ~20 raw containers & 10 MC containers
 - ~20 parameter descriptors
- ➡ ~80 names in total
- ➡ Handling name, avoid hard coding
- ➡ Name manager
 - Returns parameter/data descriptor and action names according to the class name.

Examples with GSI2021 and GSI2021_MC campaigns

TAGnameManager (i)

- Parameter descriptor names: TAGnameManager::GetParaDscName(classeName)

	MC & Raw
TAGparGeo	tgGeo
TASTparGeo	stGeo
TASTparMap	stMap
TATWparMap	twMap
TAWDparMap	wdMap
TAWDparTime	wdTim
TABMparGeo	bmGeo
TABMparConf	bmConf
TABMparCal	bmCal
TABMparMap	bmMap
TAVTparGeo	vtGeo
TAVTparConf	vtConf
TAVTparMap	vtMap
TAMSDparGeo	msGeo
TAMSDparConf	msConf
TAMSDparMap	msMap
TAMSDparCal	msCal
TATWparGeo	twGeo
TATWparConf	twConf
TATWparCal	twCal
TACAprGeo	caGeo
TACAprConf	caConf
TACAprCal	caCal
TACAprMap	caMap

→ name: detector base+type file
(**TASTparGeo** → **stGeo**)

→ Changing two class name (include 'par' in name)

TAGnameManager (ii)

- Action names: TAGnameManager::GetActionDscName(className)

TAGactTreeWriter	evtActWriter	Raw
TAGactDaqReader	daqActReader	
TAGactNtuEvent	evtActEvent	
TAGactWDreader	wdActreader	
TASTactNtuHit	stActHit	
TABMactNtuRaw	bmActRaw	
TABMactNtuHit	bmActHit	
TAVTactNtuHit	vtActHit	
TAMSDactNtuRaw	msActRaw	
TAMSDactNtuHit	msActHit	
TATWactNtuHit	twActHit	
TACAactNtuHit	caActHit	
TABMactNtuTrack	bmActTrack	
TAVTactNtuCluster	vtActClus	
TAVTactNtuTrackF	vtActTrack	
TAVTactNtuVertexPD	vtActVertex	
TAMSDactNtuCluster	msActClus	
TAMSDactNtuPoint	msActPoint	
TATWactNtuPoint	twActPoint	
TACAactNtuClusterP	caActClus	

TAGactTreeReader	evtActReader	MC
TASTactNtuHitMC	stActHitMc	
TABMactNtuHitMC	bmActHitMc	
TAVTactNtuHitMC	vtActHitMc	
TAMSDactNtuHitMC	msActHitMc	
TATWactNtuHitMC	twActHitMc	
TACAactNtuHitMC	caActHitMc	

→ name: detector base+"Act"+action type
(TABMactNtuRaw → bmActRaw)

Alias:

```
#define FootParaDscName(className) TAGnameManager::GetParaDscName(className).Data()
#define FootActionDscName(className) TAGnameManager::GetActionDscName(className).Data()
#define FootDataDscMcName(className) TAGnameManager::GetDataDscMcName(className).Data()
```

TAGnameManager (iii)

- Container names: TAGnameManager::GetActionDscName(classeName)

TAGdaqEvent	daqEvent	Raw
TAGntuEvent	evtEvent	
TACAntuRaw	caRaw	Raw
TASTntuRaw	stRaw	
TATWntuRaw	twRaw	
TAWDntuTrigger	wdTrigger	
TASTntuHit	stHit	
TABMntuRaw	bmRaw	
TABMntuHit	bmHit	
TAVTntuHit	vtHit	
TAMSDntuRaw	msRaw	
TAMSDntuHit	msHit	
TATWntuHit	twHit	
TACAntuHit	caHit	
TABMntuTrack	bmTrack	
TAVTntuTrack	vtTrack	
TAVTntuVertex	vtVertex	
TAVTntuCluster	vtClus	
TAMSDntuCluster	msClus	
TAMSDntuPoint	msPoint	
TATWntuPoint	twPoint	
TACAntuCluster	caClus	

TAMCntuEvent	mcEvent	MC
TAMCntuPart	mcPart	
TAMCntuHit	Hit	
TAMCntuHit	Hit	
TAMCntuHit	Hit	
TAMCntuHit	Hit	
TAMCntuHit	Hit	
TAMCntuHit	Hit	

→ Same class leads to same name

- Special method for MC hit container:

TAGnameManager::GetDataDsc**Mc**Name(idx)
 Idx = { kST, kBM, kVTX, kITR, kMSD, kTW, kCAL}

TAMCntuHit	stHitMc
TAMCntuHit	bmHitMc
TAMCntuHit	vtHitMc
TAMCntuHit	msHitMc
TAMCntuHit	twHitMc
TAMCntuHit	caHitMc

→ name: detector base+container type

(TABMntu**Raw** → **bmRaw**)

→ Changing a class name (include 'ntu' in name)

TAGnameManager (iv)

- Interface:
 - In base classes (TAGparaDsc, TAGdataDsc, TAGaction)

```
class TAGparaDsc : public TAGnamed {
public:
    //! fail bit
    enum { kValid = BIT(15)
};

    explicit          TAGparaDsc(const char* name, TAGpara* p_para=0);
    explicit          TAGparaDsc(TAGpara* p_para=0);
    virtual           ~TAGparaDsc();
    . . .
};

TAGparaDsc::TAGparaDsc(TAGpara* p_para)
: TAGnamed("", 0),
  fpObject(0),
  fpObjectClass(0),
  fpConsumerList(0)
{
    if (!gTAGroot) Fatal("TAGparaDsc()", "TAGroot not instantiated");
    SetBit(kMustCleanup);

    SetName(gTAGroot->DefaultParaDscName(p_para->ClassName()));
    . . .
}
```

- Add new constructor (same way for the TAGdataDsc & TAGaction)
 - ➡ Overloading TAGroot::DefaultParaDscName() method

TAGnameManager (v)

- Interface:
 - Using the default name methods in TAGroot

```
class TAGroot : public TAGobject {
public:
    TAGroot();
    virtual ~TAGroot();
    . . .
    // Default action name
    const char* DefaultActionName();
    // Default data dsc name
    const char* DefaultDataDscName();
    // Default parameter dsc name
    const char* DefaultParaDscName();

    // Default action name
    const char* DefaultActionName(const char* className);
    // Default data dsc name
    const char* DefaultDataDscName(const char* className);
    // Default parameter dsc name
    const char* DefaultParaDscName(const char* className);
    . . .
};

const char* TAGroot::DefaultActionName(const char* className)
{
    return Form("%s", FootParaDscName(className));
}
. . .
```

➡ No need to set the name, generated via TAGnameManager class

ToDo (i)

- Interface in Reco classes:

- Parameters:

```
fpParGeoBm = new TAGparaDsc("bmGeo", new TABMparGeo());
```



```
fpParGeoBm = new TAGparaDsc(new TABMparGeo());
```

➡ Just removing parameter Dsc name

- Data containers:

```
fpNtuTrackBm = new TAGdataDsc("bmTrack", new TABMntuTrack());
```



```
fpNtuTrackBm = new TAGdataDsc(new TABMntuTrack());
```

➡ Just removing data Dsc name

ToDo (ii)

- Reminder, does not work in case of TAMCntuHit:

```
fpNtuMcBm = new TAGdataDsc("bmMc", new TAMCntuHit());
```



```
fpNtuMcBm = new TAGdataDsc(FootDataDscMcName(kBM), new TAMCntuHit());
```

- Actions:

```
fActClusVtx = new TAVTactNtuCluster("vtActClus", fpNtuHitVtx, fpNtuClusVtx, fpParConfVtx, fpParGeoVtx);
```



```
fActClusVtx = new TAVTactNtuCluster("vtActClus", fpNtuHitVtx, fpNtuClusVtx, fpParConfVtx, fpParGeoVtx);
```

- ➔ Cannot simply remove name, need to modify all mother classes



```
const Char_t* name = FootActionDscName("TAVTactNtuCluster");  
fActClusVtx = new TAVTactNtuCluster(name, fpNtuHitVtx, fpNtuClusVtx, fpParConfVtx, fpParGeoVtx);
```

Conclusions

- Name manager
 - Homogenous names for all actions, containers and parameters
 - Avoid hardcoding
 - Works fine, special case for MC hit container
 - Global data containers and actions not including yet
- ➡ Ready to be interfaced with the Reco classes
- ➡ Not backward compatible (since changed name of classes) !
- ➡ Should we keep the possibility to read back from L0 for Global reconstruction ??

Perspectives (i)

- Name manager
 - Including also the branch names

```
fActEvtWriter->SetupElementBranch(fpNtuMcEvt, TAMCntuEvent::GetBranchName());
```



```
fActEvtWriter->SetupElementBranch(fpNtuMcEvt);
```

```
void TAGactTreeWriter::SetupElementBranch(TAGdataDsc* p_data, Int_t i_size, Int_t i_compress)
{
    TAGdata* obj = p_data->Object();
    TString name(obj->ClassName());
    const char* branch = TAGnameManager::GetBranchName(name);
    TAGactTreeWriterBranch* p_chan = new TAGactTreeWriterBranch(p_data, branch, i_size, i_compress, kFALSE);

    AddDataIn(p_data, "TAGdata");          // ??? be more specific ???
    fpBranchList->Add(p_chan);
    return;
}
```

Perspectives (ii)

- Name manager

```
fActEvtReader->SetupBranch(fpNtuTrackBm, TABMntuTrack::GetBranchName());
```



```
fActEvtReader->SetupBranch(fpNtuTrackBm);
```

```
void TAGactTreeReader::SetupBranch(TAGdataDsc* p_data)
{
    TAGdata* obj = p_data->Object();
    TString name(obj->ClassName());
    const char* branch = TAGnameManager::GetBranchName(name);

    TAGactTreeReaderBranch* p_chan = new TAGactTreeReaderBranch(p_data, branch);
    AddDataOut(p_data, "TAGdata"); // ??? be more specific ???
    fpBranchList->Add(p_chan);
    return;
}
```

- ➡ Centralise branch name (also special case for TAMCntuHit)
- ➡ Lighter the interfaces
- ➡ Not backward compatible (since branch names will change for unity purpose)

Perspectives (iii)

- Branch Names

TAGntuEvent	evtevt.	Raw
TACAntuRaw	cadat.	
TASTntuRaw	stdat.	
TATWntuRaw	twdat.	
TAWDntuTrigger	wdtrig.	
TASTntuHit	strh.	
TABMntuRaw	bmdat.	
TABMntuHit	bmrh.	
TAVTntuHit	vtrh.	
TAMSDntuRaw	msdat.	
TAMSDntuHit	msrh.	
TATWntuHit	twrh.	
TACAntuHit	carh.	
TABMntuTrack	bmtrack.	
TAVTntuTrack	vttrack.	
TAVTntuVertex	vtvtx.	
TAVTntuCluster	vtclus.	
TAMSDntuCluster	msclus.	
TAMSDntuPoint	mspt.	
TATWntuPoint	twpt.	
TACAntuCluster	caclus.	

TAMCntuEvent	mcevt.	MC
TAMCntuPart	mctrack.	
TAMCntuHit	rh.	
TASTntuHit	strh.	
TAMCntuHit	rh.	
TABMntuHit	bmrh.	
TAMCntuHit	rh.	
TAVTntuHit	vtrh.	
TAMCntuHit	rh.	
TAMSDntuHit	msrh.	
TAMCntuHit	rh.	
TATWntuHit	twrh.	
TAMCntuHit	rh.	
TACAntuHit	carh.	



TAMCntuEvent	mcevt.	MC
TAMCntuPart	mctrack.	
TAMCntuHit	mcst.	
TASTntuHit	strh.	
TAMCntuHit	mcbm.	
TABMntuHit	bmrh.	
TAMCntuHit	mcvd.	
TAVTntuHit	vtrh.	
TAMCntuHit	mcms.	
TAMSDntuHit	msrh.	
TAMCntuHit	mctw.	
TATWntuHit	twrh.	
TAMCntuHit	mcca.	
TACAntuHit	carh.	