

ACTS Meeting

(a coding example)

2019-3-18

A test example for Propagator

- Boost unit test : 在独立的环境中测试acts-core的代码

```
// - simple extrapolation test
BOOST_DATA_TEST_CASE(
    test_extrapolation_,
    bdata::random((bdata::seed = 0,
                  bdata::distribution
                  = std::uniform_real_distribution<>(0.4 * units::_GeV,
                                                      10. * units::_GeV)))
    ^ bdata::random((bdata::seed = 1,
                    bdata::distribution
                    = std::uniform_real_distribution<>(-M_PI, M_PI)))
    ^ bdata::random((bdata::seed = 2,
                    bdata::distribution
                    = std::uniform_real_distribution<>(1.0, M_PI - 1.0)))
    ^ bdata::random((bdata::seed = 3,
                    bdata::distribution
                    = std::uniform_int_distribution<>(0, 1)))
    ^ bdata::xrange(ntests),
    pT,
    phi,
    theta,
    charge,
    index)
{
```

- 可以设置test的数目，相当于事例数

```
const int ntests    = 500;
bool      debugMode = true;
```

Propagator : ATLAS 径迹外推

- 径迹传播 Surface A -> Surface B
- Transport of par&cov (Jacobi)
- 通过Stepper计算径迹的运动方程
 - $dr/ds = T$ $dT/ds = q/p * (T \times B)$
 - Runge Kutta 方法确定每一个step的精度
- 通过Navigator 确定径迹路径上经过了哪些平面
- 平面上材料反应(能损, 多次散射),遇到测量 (Kalman)

A basic example of Propagator

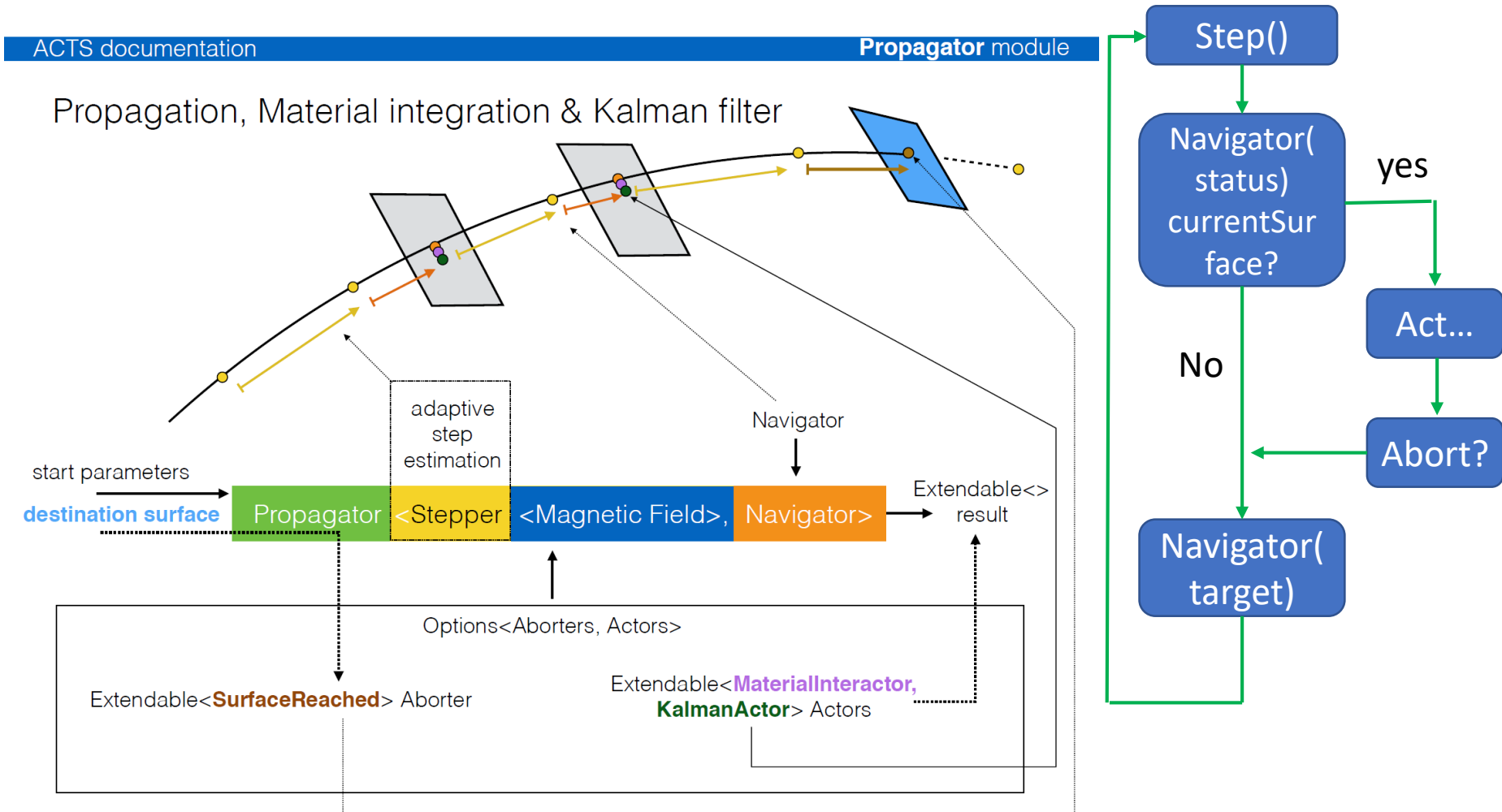
```
// build Geom
CylindricalTrackingGeometry cGeometry;
auto tGeometry = cGeometry();
// Get the navigator and provide the TrackingGeometry
Navigator navigator(tGeometry);
//construct Stepper
EigenStepperType estepper(bField);
//construct Propagator
EigenPropagatorType epropagator(std::move(estepper), std::move(navigator));
//the options
PropagatorOptions<> options;
//in boost test : do propagate() //
BOOST_DATA_TEST_CASE(
    test_extrapolation_,
    //.....
    //.....)
{
    //the track par
    Vector3D pos(x, y, z);
    Vector3D mom(px, py, pz);
    CurvilinearParameters start(std::move(covPtr), pos, mom, q);
    BOOST_CHECK(epropagator.propagate(start, options).endParameters != nullptr);
}
```

构建几何

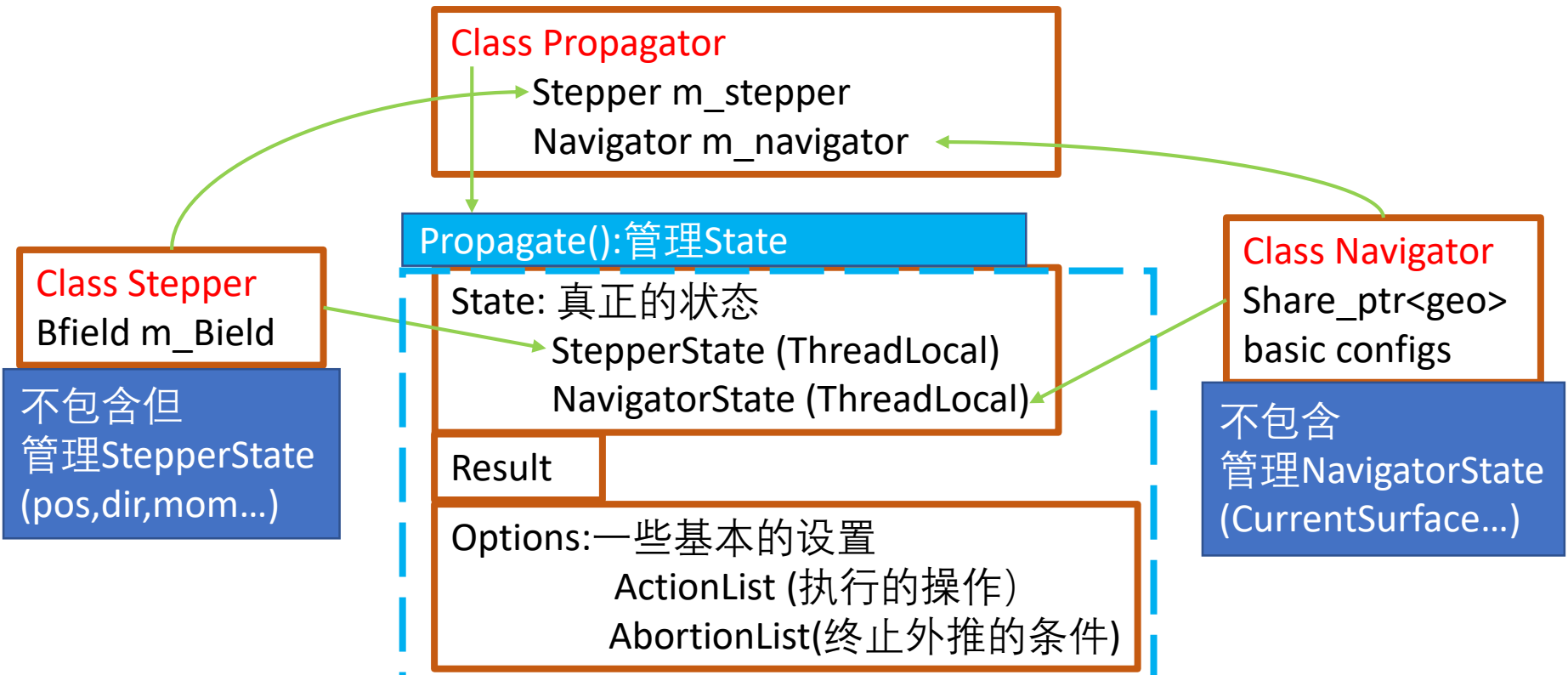
构建Propagator

Unit test: propagate()

A sketch From ACTS meeting



Structure – nested struct



State: 外推中只由外推函数propagate()构建一次，保存径迹状态
Stepper/Navigator类通过nested-struct，只提供方法负责处理对应的State，分离了数据和方法

meta programming

- With boost::mpl (现在准备换成boost::hana)
- 编译期处理
 - Deal with the type on compile-time constants
 - computing factorial of a number

```
// This struct is a meta-function which normally maps to BoundParameters...
template <typename T, typename S>
struct s
{
    using type = BoundParameters;
};

// ...unless type S is int, in which case it maps to Curvilinear parameters
template <typename T>
struct s<T, int>
{
    using type = CurvilinearParameters;
};
```

根据模板实例化时的参数值确定，径迹参数用global frame 还是local frame

- ActionList/AbortList
 - A list of Actor/Aborter
- Actor :在径迹传播的时候要做什么
 - Kalman filter, material interactor , debug...
 - 通过Operator()调用
- Actionlist:基于std::tuple的扩展数据类型

```
PropagatorOptions<ActionList<MaterialInteractor, DebugOutput>> options;
```

- You can put actors when you write a test

```
template <typename... actors_t>
struct ActionList : public detail::Extendable<actors_t...>
{
```

- 在编译期解析要添加的actor和递归调用顺序

```
template <typename first, typename... others>
struct action_list_impl<first, others...>
template <typename last>
struct action_list_impl<last>
```


递归计算的一个例子

- 计算工具：例如Eigen
- 许多的设计方式模仿boost/std
- 语言更新快(c++17)

The ACTS on CEPC

- Geometry
 - Use DD4HEP – 放进了CEPC的几个子探测器 (by 韩雨亭)
- Acts-core
 - Seeding
 - Track finding
- Acts-framework : now in a simple Gaudi
- Acts-fatras