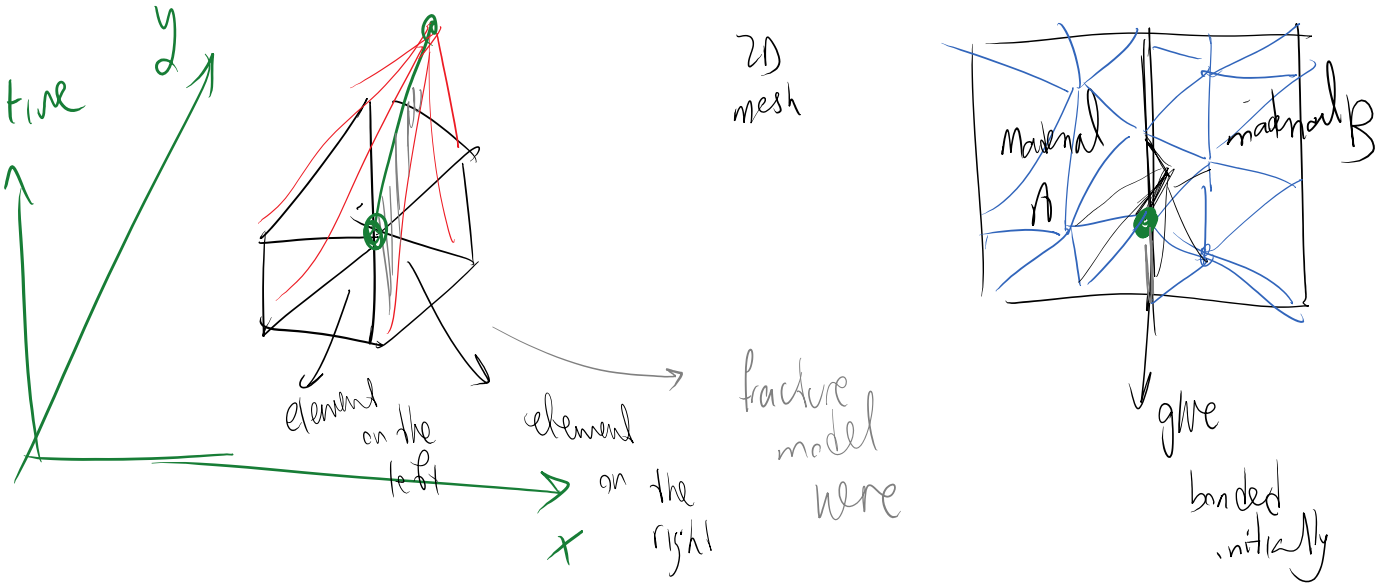
2018/04/16

Friday, April 13, 2018 9:12 PM



We pitch from a local minimum vertex (the time coordinate of the vertex is minimum relative to its star)

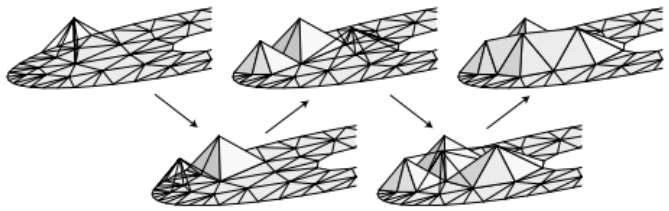


Fig. 4. Pitching tents in space–time.

mechanics here

Integration all of Damage

Damage model here

Damage model weighted residual

$$\int_{\Omega} \hat{D}\left(\dot{D} - D_{src}\right) dV + \int_{\partial\Omega} \hat{D}\left(\hat{D}^{*} - D\right) n_t \, dS = 0$$

Interior of PIC (Physics Integration cell)

Solid Mechanics

$$\int_{\Omega} \left(\hat{\ddot{u}}\rho + \hat{\nabla u}\,\sigma + \hat{u}\rho b\right) dV +$$

$$\int_{\partial\Omega} \hat{u}\,\sigma^{*} n_x \, dS - \hat{u}\,p^{*} n_t \, dS + [E]\hat{\sigma}\, n_t \, dS + [u]\hat{\sigma} n_x \, dS$$

$$+ \hat{u}_0 [u] n_t \, dS = 0$$

$p = \rho v = \rho \dot{u}$ linear momentum $v = \dot{u}$ velocity

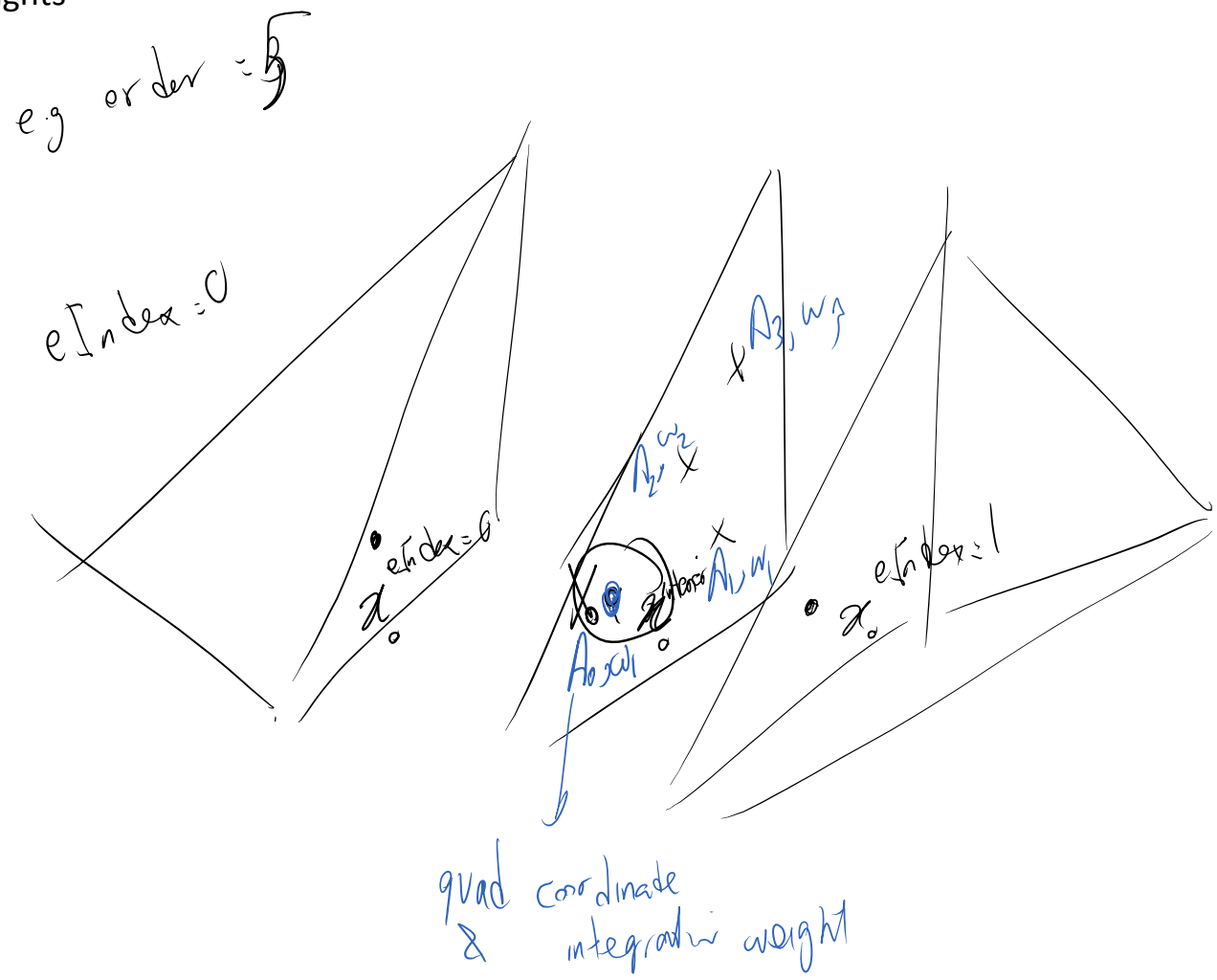
$E = \frac{1}{2}\left(\nabla u + \nabla u^{T}\right)$ strain $\sigma = C E$ stress

$[f] := f^* - f$

$\vec{n}_x, n_t$ are spatial & temporal normals to a face

The process for the solution of integration cell:

1. Set the integration order: for all the elements attached to PIC we ask them to give their integration order for the given type of cell integration.
2. From integration order -> integration points and weights



3. Loop over the quadrature points

3.1 Form Coord

1- Quadrature coordinate Alpha (A)

2- Global Cordinate X

3- basis cordinate

physics\PhyCoord.h

class ptCoords

Intenor basis cordinate

eCoord eCrdInterior; // the coordinate for element on interiorIntegration cell
vector<eCoord> eCrdFacet; // the vector of coordinates for elements on facetIntegration cells

facet basis coordinate

// storing integration related members:
// actual position of point
QCoord quadC; // quadrature coordiante for integration cell

Alpha

XCoord XC; // global cartesian coordinate

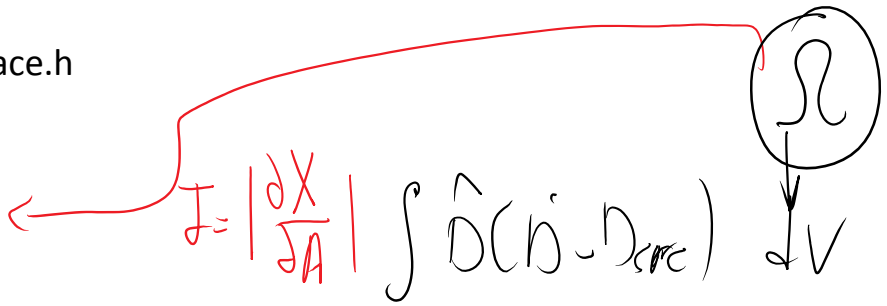
X

Last member of the coordinate:

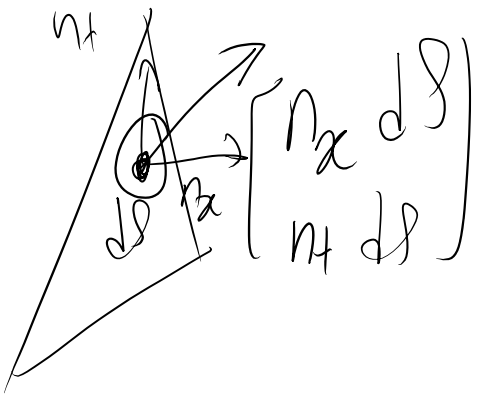
GeomPropInt* gpi;

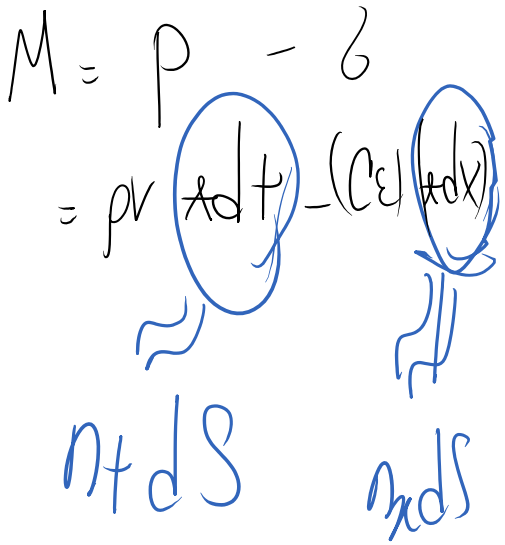
physics\PhyGeomInterface.h
class GeomPropInt

double omega2Alpha;



vector<PhyFacetForms> facet_dForms; for faces





The class Facet forms:

physics\PhyFacetForms.h

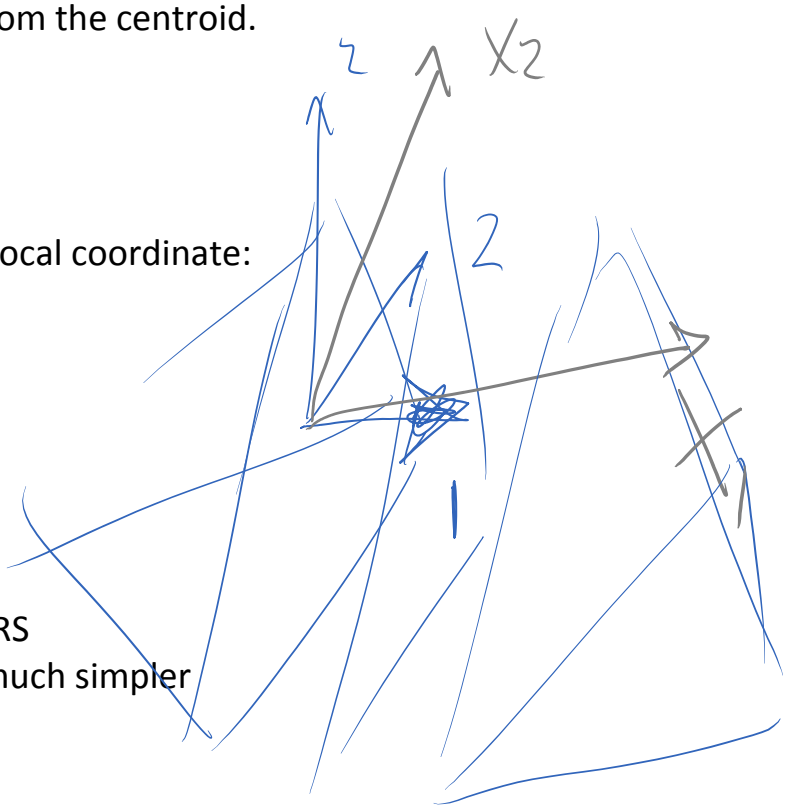
class PhyFacetForms
....
VECTOR sdxAll;

like SPACETIME normal * dS

Often GeomPropInt doesn't change much and basically for constant Jacobian PIC it's constant.

Trick:

We create a ptCoords (coordinate) at integration cell at the centroid and set it up IF the PIC is constant Jacobian. Then for any point if the element is constant Jacobian we ust borrow precomputed properties from the centroid. Otherwise, we calculate it per-point.

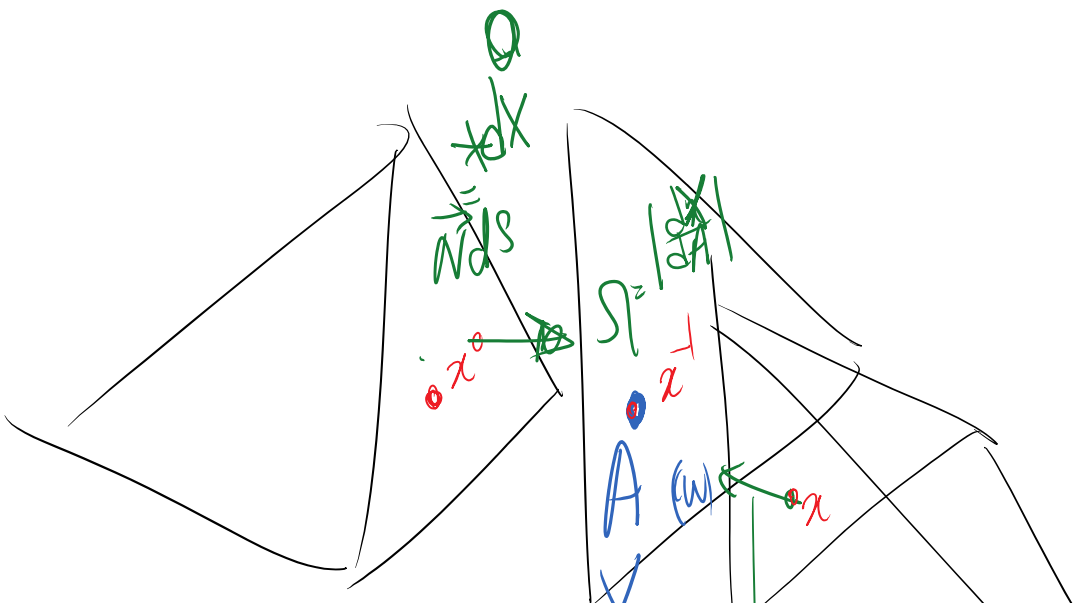


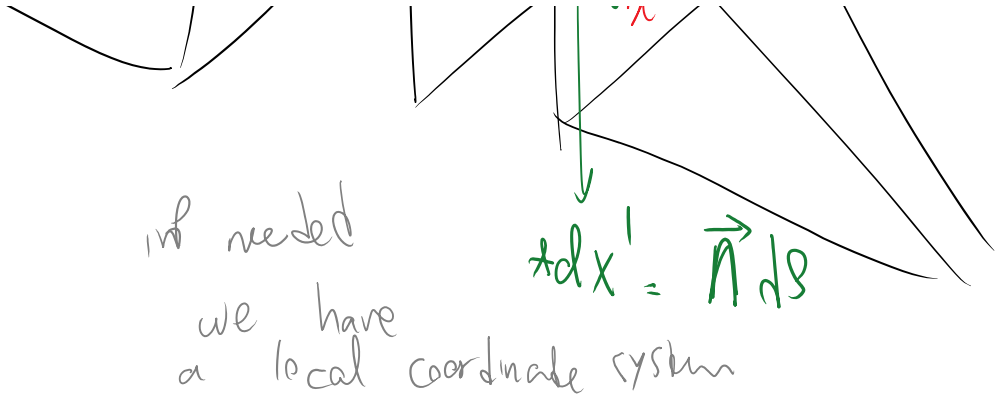
The last member of GeomPropInt is a local coordinate:

bool hasLocalCoord;
V2TENSOR localCoordBasis;

Benefits of a local coordinate:

1. Many terms become zero in the WRS
2. Target value calculation becomes much simpler





$$\int_\Omega \hat{D}\,(\dot{D} - D_{src})\,dV + \int_{\partial\Omega} \hat{\sigma}(\dot{D}^* - D)\,n_t\,dS = 0$$

$$\int_\Omega \left(\hat{\ddot{u}}\rho + \nabla\hat{\dot{u}}\,\sigma + \hat{u}\rho b\right)dV +$$

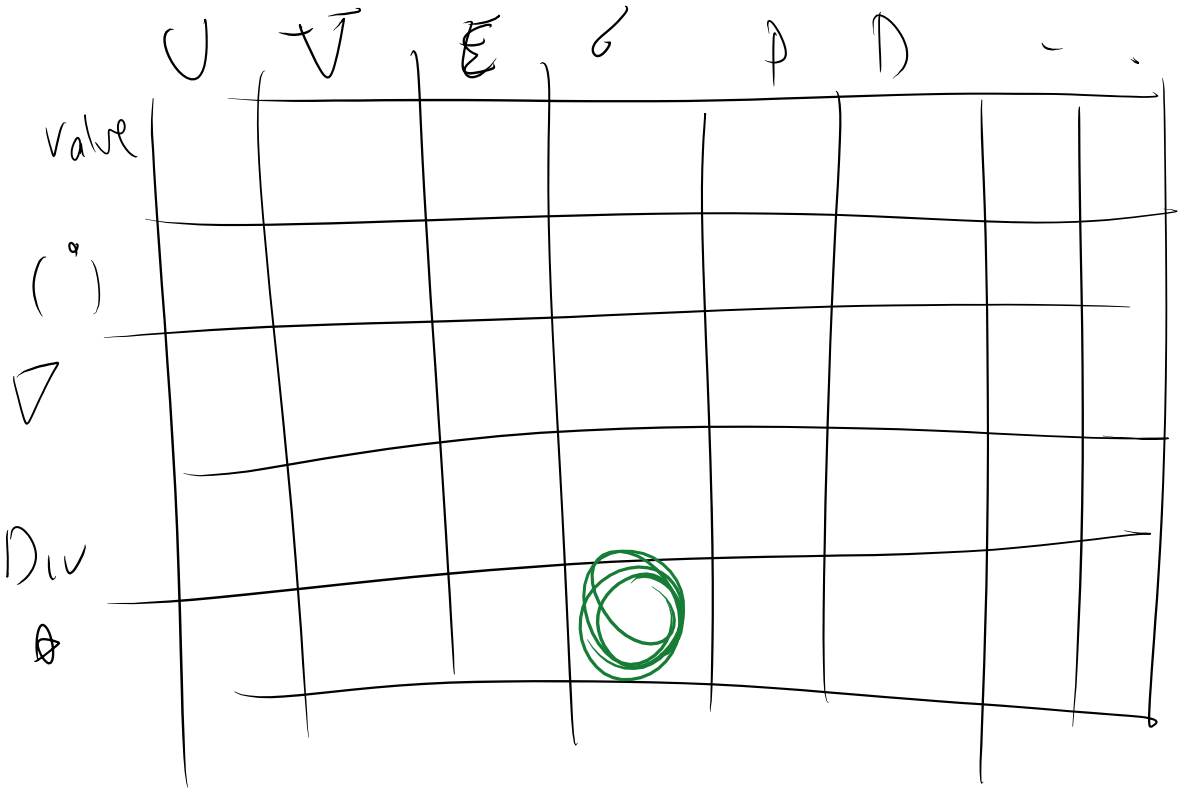
$$\int_{\partial\Omega} \hat{\dot{u}}\,\sigma^*\,n_x\,dS - \hat{u}\,p^*\,n_t\,dS + [E]\,\hat{\sigma}\,n_t\,dS + [\dot{u}]\,\hat{\sigma}\,n_x\,dS$$

$$+ \hat{u}_0\,[u]\,n_t\,dS = 0$$

Naming tensor fields

Tensor fields are recognized by their physics type and computation type

class PhyFldC → physics name (stress, disp. strain, ...)
phyFld phyF;
compT cT; → value, DT, DX, symDx, Div, Src, *



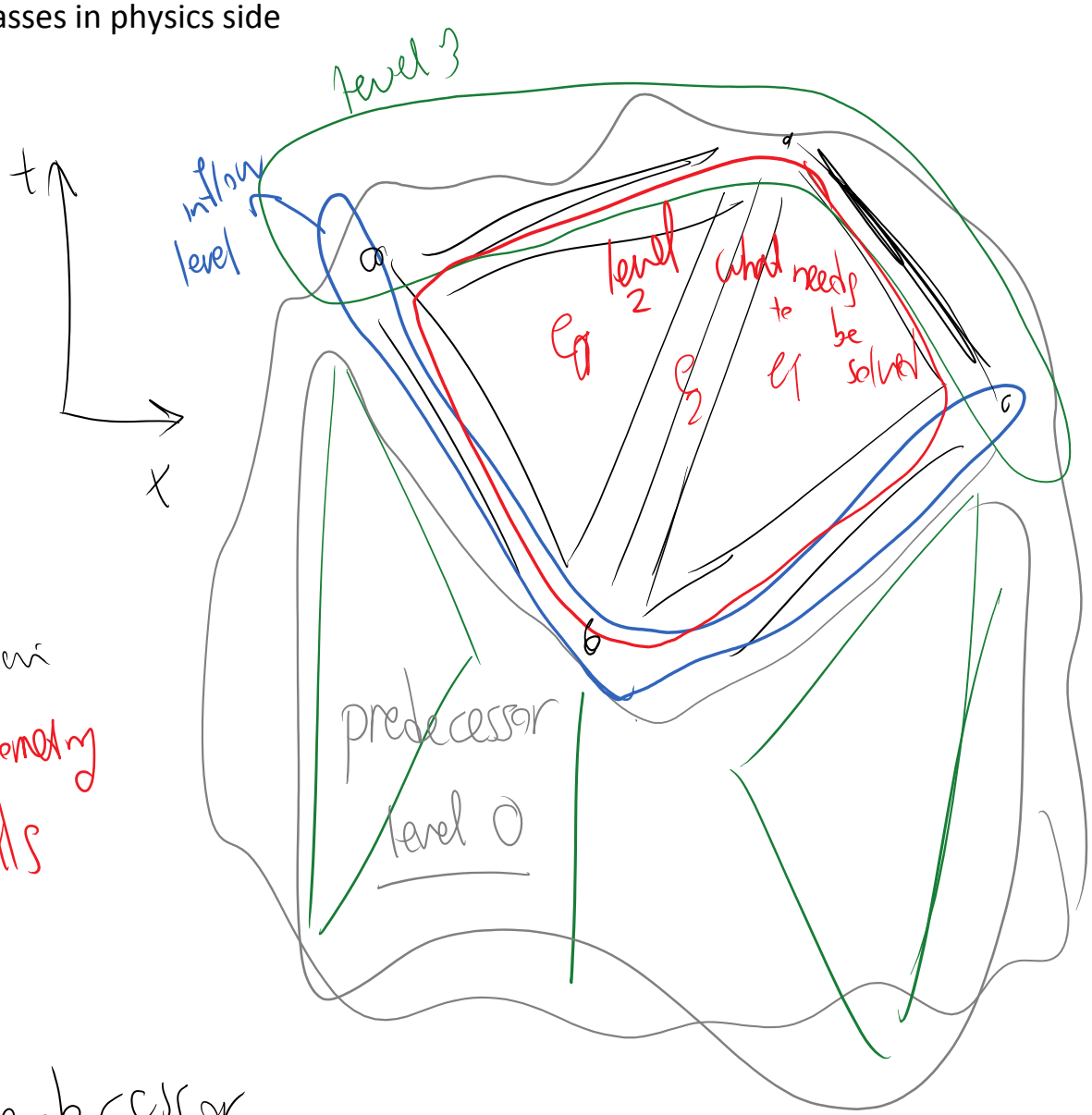
Each of the elements (interior and faces) require their storage for values.

To be continued next time

Hierarchy of classes in physics side

Model 3

Hierarchy of classes in physics side



patch is a collection of geometry cells

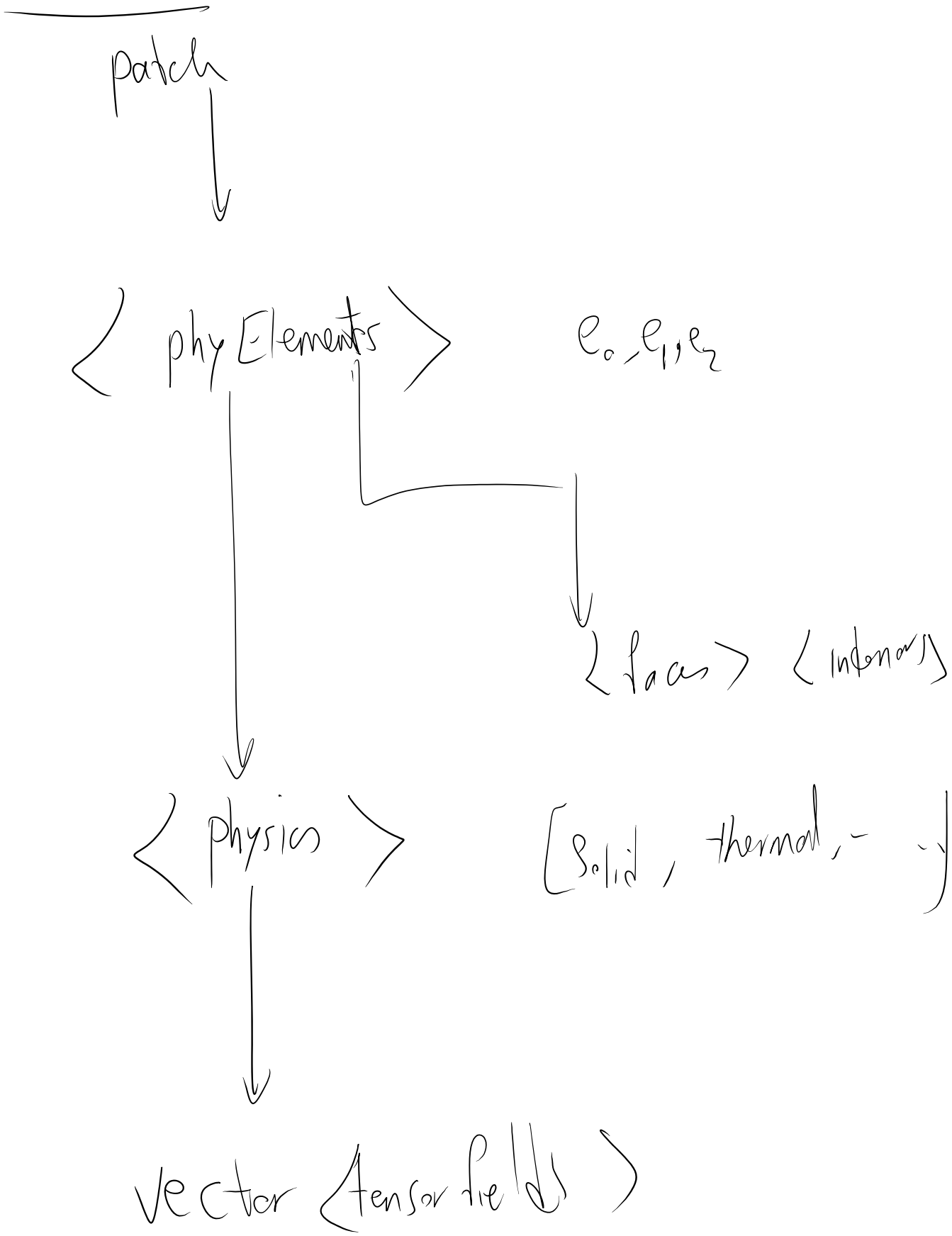
1- predecessor

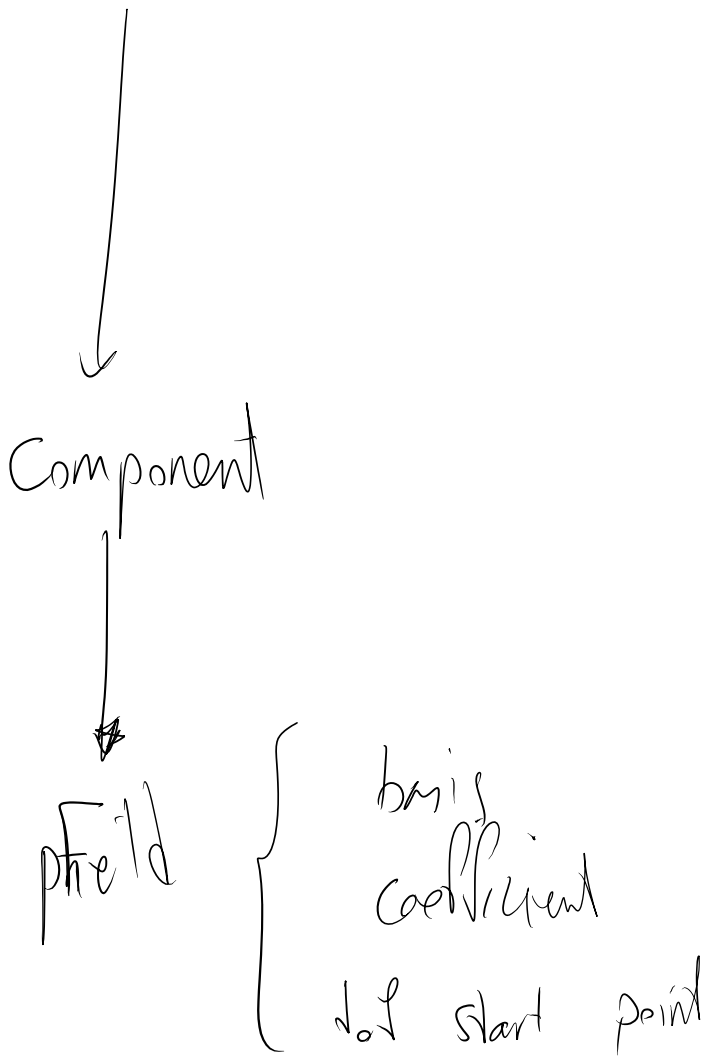
1. inflow

2- active

3- outflow

new elements


patch
< phyElements >
e0, e1, e2
< faces > < interfaces >
< physics >
[Solid, thermal, ...]
vector < tensor fields >


Component
pfield
basis
coefficient
dof start point