Subject: PROPOSAL: linear Scaler helper for different min/max domains
Posted by kohait00 on Wed, 01 Dec 2010 09:19:03 GMT

View Forum Message <> Reply to Message

hey all

often one ends up using the same pattern over and over again for traversing from i.e. pixel domain (Draw in Ctrl) to the representative Value domain, and back to notify on user action..its a nightmare

what about this one? its a *linear* scaler..


```
template<class T = double>
class Scaler
{
public:
 Scaler() : mn(0), mx(0) {}
 Scaler(const T& mn, const T& mx) : mn(mn), mx(mx) {}

 inline void Min(const T& t) { mn = t; }
 inline T Min() const { return mn; }
 inline void Max(const T& t) { mx = t; }
 inline T Max() const { return mx; }
 inline void MinMax(const T& _mn, const T& _mx) { mn = _mn; mx = _mx; }

 //scales local dimension value t to foreign dimensions d
 //returned in foreign dimension
 inline T To(const Scaler& d, const T& t) { return (t-mn)*(d.mx-d.mn)/(mx-mn)+d.mn; }
 //scales foreign dimension value t from foreign s to local dimension
 //return in local dimension
 inline T From(const Scaler& s, const T& t) { return (t-s.mn)*(mx-mn)/(s.mx-s.mn)+mn; }

 inline T operator() (const Scaler& s, const T& t) { return From(s, t); }
protected:
 T mn, mx;
};
```


using it like this (v for value domain, p for pixel domain)
here only for x axis i.e.

```
 Scaler<double> vsx, psx;

//from draw domain to value domain
 valx = vsx(psx, Point().x);
```

```
//from value domain to draw domain
 Point().x = (int) psx(vsx, valx);
```

i think also of type conversions to be able to convert scaler to scaler with different types..
trying to avoid virtualisation here, but thinking of a logarithmic scaler as well, but then probably
would need to virtualize an interface.

any hints / improvements welcome