Pensieve Header: Work on the "infinitesimal Alexander module"; continues 2008-09/AlexanderEulerSpaces.nb. See also <a href=http://katlas.math.toronto.edu/drorbn/bbs/show?shot=KAL-090701-153823.jpg>http://katlas.math.toronto.edu/drorbn/bbs/show?shot=KAL-090701-153823.jpg</a>.

```
<< KnotTheory`
```

```
Loading KnotTheory` version of April 20, 2009, 14:18:34.482.
Read more at http://katlas.org/wiki/KnotTheory.
```

```
{Alexander[Knot[4, 1]][t], Alexander[Knot[7, 7]][t]}
```

KnotTheory::loading : Loading precomputed data in PD4Knots`.

$$\left\{3-\frac{1}{t}-t,\ 9+\frac{1}{t^2}-\frac{5}{t}-5\,t+t^2\right\}$$

```
K = Knot[7, 7]; pd = PD[K];
gc =
  GC @@ pd /. X[i_, j_, k_, l_] :> If[PositiveQ[X[i, j, k, l]], Ar[l, i, +1], Ar[j, i, -1]];
False && (gc = GC[Ar[1, 3, +1], Ar[4, 2, -1]]);
gc
```

```
GC[Ar[4, 1, -1], Ar[10, 5, -1], Ar[8, 3, 1],
 Ar[2, 9, 1], Ar[14, 11, -1], Ar[12, 7, 1], Ar[6, 13, 1]]
```

Conventions for red objects:

1. Legs start just to the right of the index; ar[0,7] means a red arrow starting to the right of position 0 (that is, to the left of everything) and ending to the right of position 7).
2. If two (red) indices are the same, the heads are to the right of the tails.
3. w[] is the legless wheel object.
3. y[i,j,k] means "red Y with tails are i and j and head at k".

```
n = 2 Length[gc]; range = Range[0, n];
Short[AllRedObjects = Flatten[{
    Outer[ar, range, range], Outer[y, range, range, range], w[]
   }]]
```

```
{ar[0, 0], ar[0, 1], ar[0, 2], ar[0, 3], ar[0, 4], <<3591>>,
 y[14, 14, 11], y[14, 14, 12], y[14, 14, 13], y[14, 14, 14], w[]}
```

The relations associated with a red objects involve all the ways of "pulling one leg one unit to the left" :

```
ar[d[i_], j_] := ar[i, j] - ar[i - 1, j] + If[i - 1 == j, w[], 0];
ar[i_, d[j_]] := ar[i, j] - ar[i, j - 1]  + If[i == j,  -w[], 0];
y[d[i_], j_, k_] := y[i, j, k] - y[i - 1, j, k] + If[i - 1 == k, -w[], 0];
y[i_, j_, d[k_]] := y[i, j, k] - y[i, j, k - 1] + If[i == k, w[], 0] + If[j == k, -w[], 0];
```

Now let's form all the red relations; starting with the anti-symmetry of y:

```
RedRelations = {};
RR[rel_RuleDelayed] := AppendTo[RedRelations, rel];
SetAttributes[RR, Listable];
RR[y[t1_, t2_, h_] * _. :> y[t1, t2, h] + y[t2, t1, h]];
```