Aw-Calculus Programs for the WKO4 Paper

Pensieve header: Aw-calculus programs for the WKO4 paper.

"d" is "ht": along tube strands, heads appear before tails.

```
Print["AwCalculus` implements / extends ",
  Sort@{"**", "⋃", "≡", dA, deg, dm, dS,
    dΔ, dη, dσ, El, Es, hA, hm, hS, hσ, tA, tha, tm, tS, tσ, Γ, Λ},
  "."];
```

```
AwCalculus` implements / extends
 {**, ⋃, ≡, dA, deg, dm, dS, dΔ, dη, dσ, El, Es, hA, hm, hS, hσ, tA, tha, tm, tS, tσ, Γ, Λ}.
```

Utilities

```
deg /: (h_)^deg := DegreeScale[h];
```

The AT Presentation E_l of A^w

```
El[λ_, ω_][d_] := El[λ[d], ω[d]];
El /: El[λ1_, ω1_] ≡ El[λ2_, ω2_] := (λ1 ≡ λ2) && (ω1 ≡ ω2);
El /: El[λ1_, ω1_] ⋃ El[λ2_, ω2_] /; Support[λ1] ⋂ Support[λ2] == {} :=
 El[λ1 ⋃ λ2, ω1 + ω2];
```

ElStackingDef

```
El /: El[λ1_, ω1_] ** El[λ2_, ω2_] /; Support[λ1] == Support[λ2] :=
  El[BCH_tb[λ1, λ2], e^(-D_λ2)[ω1] + ω2];

El /: El[λ1_, ω1_] ** El[λ2_, ω2_] := NonCommutativeMultiply[
  El[λ1 ⋃ ((# → LS[0]) & /@ ⟨Complement[Support@λ2, Support@λ1]⟩), ω1],
  El[λ2 ⋃ ((# → LS[0]) & /@ ⟨Complement[Support@λ1, Support@λ2]⟩), ω2]
 ]

El[λ_, ω_] // dη[s_] :=
  El[(λ \ s) // LieMorphism[LW[s] → 0], ω // LieMorphism[LW[s] → 0]];
```

EldA

```
El[λ_, ω_] // dA := El[-λ, e^D_λ[ω] - j[λ]];

El[λ_, ω_] // dS := El[
   -λ // (-1)^deg,
   (e^D_λ[ω] - j[λ]) // (-1)^deg
  ];
```

EldDelta

```
El[λ_, ω_] // dΔ[a_, b_, c_] := El[
  (λ \ a) ⋃ ⟨b → λa, c → λa⟩ // LieMorphism[LW@a → LW@b + LW@c],
  ω // LieMorphism[LW@a → LW@b + LW@c]]
```

The Split Presentation E_s of A^w

```
Es /: Es[λ1_, ω1_] ≡ Es[λ2_, ω2_] := (λ1 ≡ λ2) && (ω1 ≡ ω2);
Es[λ_, ω_][d_] := Es[λ[d], ω[d]];
```

EsSampleDefs

```
Es /: Es[λ1_, ω1_] ⋃ Es[λ2_, ω2_] /; Support[λ1] ⋂ Support[λ2] == {} :=
  Es[λ1 ⋃ λ2, ω1 + ω2];
Es[λ_, ω_] // hm[x_, y_, z_] := Es[λ // hm[x, y, z], ω];
Es[λ_, ω_] // tm[u_, v_, w_] :=
  LieMorphism[LW@u → LW@w, LW@v → LW@w] /@ Es[λ, ω];
Es[λ_, ω_] // tha[u_, x_] := Es[λ // RCu[λx], (ω + Ju[λx]) // RCu[λx]];

tσ[us_List → vs_List][ser_LieSeries | ser_CWSeries | ser_AngleBracket] :=
  ser // LieMorphism[Thread[(LW /@ us) → (LW /@ vs)]];
tσ[u_, v_] := tσ[{u} → {v}];
tσ[us_List → vs_List][ξ_Es] := tσ[us → vs] /@ ξ;
hσ[xs_List → ys_List][λ_AngleBracket] :=
  Union[λ \ xs, ⟨Thread[ys → Table[λx, {x, xs}]]⟩];
hσ[x_, y_] := hσ[{x} → {y}];
hσ[xs_List → ys_List][Es[λ_, ω_]] := Es[λ // hσ[xs → ys], ω];
dσ[as_List → bs_List][ξ_] := ξ // tσ[as → bs] // hσ[as → bs];
dσ[a_, b_][ξ_] := ξ // tσ[a, b] // hσ[a, b];
```

Esdm

```
ξ_Es // dm[a_, b_, c_] := ξ // tha[a, b] // tm[a, b, c] // hm[a, b, c];

tm[u_, v_, w_][λ_AngleBracket] := λ // LieMorphism[LW@u → LW@w, LW@v → LW@w];
hm[x_, y_, z_][λ_AngleBracket] := Union[λ \ {x, y}, ⟨z → BCH[λx, λy]⟩];
tha[u_LW, x_][λ_AngleBracket] := λ // RCu[λx];
dm[a_, b_, rest__, c_][ξ_] := ξ // dm[b, rest, b] // dm[a, b, c];

Es /: Es[λ1_, ω1_] ** Es[λ2_, ω2_] /; Support[λ1] == Support[λ2] := Module[
   {supp, temps, ξ},
   supp = Support[λ1];
   temps = Complement[Characters["0123456789abcdefghijklmnopqrstuvwxyz"],
      ToString /@ supp][[1 ;; Length@supp]];
   ξ = Es[λ1, ω1] ⋃ (Es[λ2, ω2] // dσ[supp → temps]);
   MapThread[(ξ = ξ // dm[#1, #2, #1]) &, {supp, temps}] // Last
  ];

tA[u_][expr_] := expr;
hA[x_][Es[λ_, ω_]] := Es[Union[λ \ x, ⟨x → -λx⟩], ω];
dA[a_][μ_] := μ // hA[a] // tha[LW@a, a];
dA[a_, rest__][μ_] := μ // dA[a] // dA[rest];
Es[λ_, ω_] // dA := Es[λ, ω] // (dA @@ Support[λ])
```

```
tS[u_][λ_AngleBracket] :=
  ⟨Table[x → LieMorphism[LW@u → -LW@u][λx], {x, Support[λ]}]⟩;
tS[u_][Es[λ_, ω_]] := Es[λ // tS[u], ω // LieMorphism[LW@u → -LW@u]];
hS[x_][Es[λ_, ω_]] := Es[Union[λ \ x, ⟨x → -λx⟩], ω];
dS[a_][μ_] := μ // tS[a] // hS[a] // tha[LW@a, a];
dS[a_, rest__][μ_] := μ // dS[a] // dS[rest];
Es[λ_, ω_] // dS := Es[λ, ω] // (dS @@ Support[λ])
```

The $E_l \leftrightarrow E_s$ Conversions

```
Γ[El[λ_, ω_]] := Es[Γ[λ], ω];
Λ[Es[λ_, ω_]] := El[Λ[λ], ω];
```