

```

(*****)
(* Generalized Arrow: *)
(* *)
(* A generalized arrow is a monoidal functor from an enriching category to an enriched category. *)
(* *)
(*****)

```

Generalizable All Variables.

```

Require Import Preamble.
Require Import General.
Require Import Categories_ch1_3.
Require Import Functors_ch1_4.
Require Import Isomorphisms_ch1_5.
Require Import ProductCategories_ch1_6_1.
Require Import OppositeCategories_ch1_6_2.
Require Import Enrichment_ch2_8.
Require Import Subcategories_ch7_1.
Require Import NaturalTransformations_ch7_4.
Require Import NaturalIsomorphisms_ch7_5.
Require Import MonoidalCategories_ch7_8.
Require Import Coherence_ch7_8.
Require Import Enrichment_ch2_8.
Require Import Enrichments.
Require Import RepresentableStructure_ch7_2.
Require Import PreMonoidalCenter.
Require Import PreMonoidalCategories.
Require Import BinoidalCategories.

```

```

Class GeneralizedArrow (K:Enrichment)(C:Enrichment) :=
{ ga_functor_obj      : enr_v K -> C
; ga_functor          : Functor      (enr_v_mon K) (enr_c_pm C) ga_functor_obj
; ga_functor_monoidal : PreMonoidalFunctor (enr_v_mon K) (enr_c_pm C) ga_functor

```

```

(* We require that the host language (but NOT the guest language) be pure, i.e. all morphisms central, to simplify
 * things.  If this doesn't suit you, just consider the "host language" here to be the pure sublanguage of the
 * host language, and toss on the inclusion functor to the full language *)
; ga_host_lang_pure   : CommutativeCat (enr_c_pm C)
}.

```

```

Coercion ga_functor_monoidal : GeneralizedArrow >-> PreMonoidalFunctor.

```

```

Implicit Arguments GeneralizedArrow [ ].

```

```
Implicit Arguments ga_functor_obj      [ K C ].
Implicit Arguments ga_functor         [ K C ].
Implicit Arguments ga_functor_monoidal [ K C ].
```