

T: Basetype
IV: Intervaltype

ErrorPolicy

```
+base_t: typedef = T
+iv_t: typedef = IV
+af_t: typedef = AffineForm<T, ?, ?, ?, ?, ErrorPolicy, IV>
+aerror_t: typedef = ArithmeticError<T>
+special_t: typedef = {PolicyDefinedType}

+pre_op(af1:af_t&,af2:const af_t&): bool
+pre_op(af:af_t&,s:base_t): bool
+pre_op(af:af_t&,iv:const iv_t&): bool
+pre_op(af:af_t&): bool
+post_op(af1:af_t&,af2:const af_t&,err:const aerror_t&): void
+post_op(af:af_t&,s:base_t,err:const aerror_t&): void
+post_op(af:af_t&,iv:const iv_t&,err:const aerror_t&): void
+post_op(af:af_t&,err:const aerror_t&): void
+special(val:special_t): void
+new_form(af:af_t&,s:base_t): bool
+new_form(af:af_t&,iv:const iv_t&): bool
```