The following is an attempt to write a denotational semantics for Nu's scripting system. This is a WIP.

Axiomatic semantics - Note µ:Axiom and µ:Value each have circular definitions to make it irreducible in this context.

rec µ:Axiom *a* = µ:String -> µ:Axiom *a*

and µ:Value *a* =

| µ:Value µ:Relation

| µ:Value µ:Address

| µ:Value µ:Name

| µ:Value µ:String

| ... // the remaining primitives

| µ:Value µ:Unit

µ:Values *a* = µ:Axiom (µ:Value *a*) "A consumable sequence of values."

µ:Effect = µ:Axiom Unit "An effect on the state of the world."

Derived Semantics

µ:Get *a* = µ:Name -> µ:Relation -> *a*

µ:Set *a* = µ:Name -> µ:Relation -> µ:Value *a* -> µ:Effect

µ:Define *a* = µ:Name -> µ:Relation -> µ:Value *a* | µ:Value *a*

µ:Stream *a* = µ:Name -> µ:Relation -> µ:Values *a* | µ:Stream *a* -> Lambda *b* *a* -> µ:Values *a*

µ:Variable *a* = µ:Name -> µ:Relation -> µ:Values *a* | µ:Stream *a* -> µ:Values *a*

µ:Equate *a* = µ:Name -> µ:Relation -> µ:Set µ:Name µ:Relation | µ:Stream *a* -> µ:Set µ:Name µ:Relation