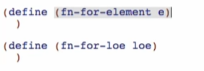
Making the Templates:

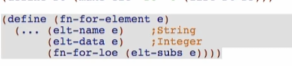


**fn-for-element Template:**

Compound data with the type produced (return values) for the Element template

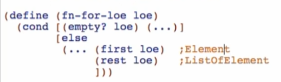


But reference rule says to wrap the selectors that produce non-primitive types (ListOfElement) in the appropriate function



**fn-for-loe Template:**

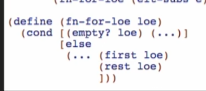
One of two cases with the type produced (return values) for the ListOfElement template

 -> 

* 1st case – atomic distinct empty

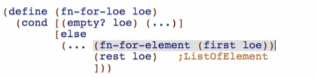


* 2nd case – compound cons

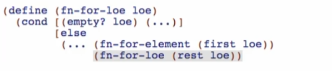


Replacing the type comments with the templates

Element (Reference)



ListOfElement (self-reference)

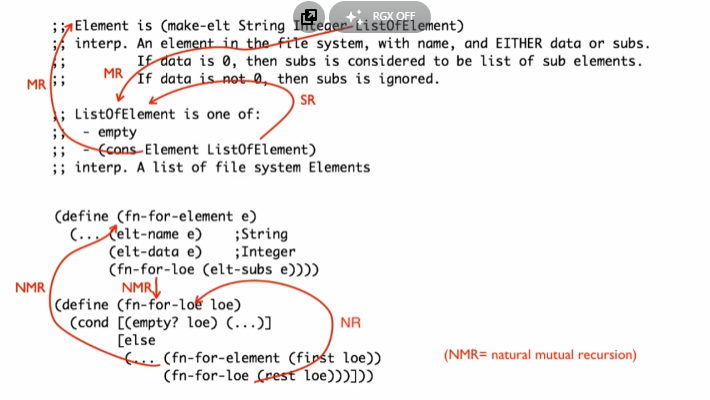


Note: Write templates for mutually referential data types together

Only new rule: make both templates simultaneously!

* Finish first structure to examples for both type definition
* Then make the templates together for both types to see the mutual recursion and recursion
* See overview below:

Overview:



Note: on correspondence, try to make natural recursion as last

Arrow Labels

#1 – NH – Natural Helpers

#2 – NMR – Natural Mutual Recursions

#3 – NR – Natural Recursions