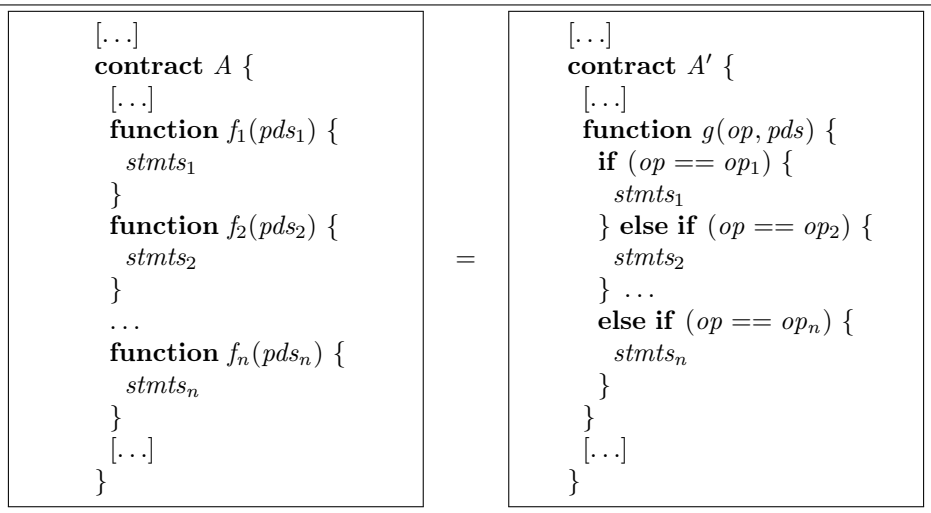
Rule 0.19 ⟨*Limit Number of Functions*⟩



where

$f_1, f_2, \ldots, f_n$ are related functions in contract A with similar purposes;

g is the consolidated function in contract A' that combines the functionality;

op is an operation selector parameter (e.g., enum or integer) that determines which logic to execute;

op_i represents the selector value corresponding to function f_i;

pds_i are the parameter declarations of function f_i;

pds are the unified parameter declarations in function g;

$stmts_i$ represents the statement sequence of function f_i.

provided

Functions $f_1, \ldots, f_n$ are semantically related and operate on similar data;

The consolidated function g maintains all functionality of the original functions;

The operation selector op unambiguously identifies which logic path to execute;

The consolidation reduces deployment costs without sacrificing security or readability;

Access control and validation logic remain equivalent in the consolidated version;

The number of external function selectors is reduced, lowering contract size.

Invariant:

Let s_i and s_i' be the initial state of A and A', respectively.

Let s_f and s_f' be the state reached by A and A', respectively, after $A.f_k()$ and $A'.g(op_k, \ldots)$ are executed from s_i and s_i', respectively.

Then, the coupling invariant is

$$\forall\, s_i, s_i', k \in \{1, \ldots, n\}\ .\ (s_i = s_i') \rightarrow (s_f = s_f')$$