## Anubis - Feature #55 Compiler optimization

09/04/2008 05:00 PM - Anonymous

-			
Status:	New	Start date:	
Priority:	High	Due date:	
Assignee:		% Done:	0%
Category:	Compiler	Estimated time:	0.00 hour
Target version:	1.10		
Platform:		Triage Stage:	
Resolution:			
Description			
With this a type defined like this :			
public define [[PhysicalAddress]]			
mac_address(Word8 lowest, Word8 b1, Word8 b2, Word8 b3, Word8 b4, Word8 highest).			
The following expression takes about 300 ms to be compiled on my Core 2 Duo @ 2.2 GHz :			
mac address(255, 255, 255, 255, 255, 255)			
and use more than 4 000 000 temporary pairs.			
This is due to the fact that all parameters of this function are evaluated before the function itself. Thus as each parameter can have			
about 10 or 11 alternatives, the total number of alternative before the function is evaluated is 10!^6!!!			
A greater algorythm for function call evaluation should be :			
1. Creating the function interpretations list (only based on its name)			
1. Taking first parameter, and for each function interpretation, verify its compatibility			
1. If not compatible, remove the function interpretation from list			
1. Loop with next parameter until last, or until function interpretation list is empty			

## History

## #1 - 01/20/2009 09:00 AM - Anonymous

- Target version changed from 1.9 to 1.10
- Platform deleted (All)
- 3 deleted (Not started)

## #2 - 01/22/2009 10:12 PM - Alain Prouté

- Assignee deleted (Alain Prouté)
- Platform deleted (All)