Details:

{

Title : "Launch" ,

File Name : "Launch.mcsource" ,

Date Created: "2-15-2019" ,

Description : "The Launcher for the Engine."

};

Context(NotationEngine::Core::Launch)

Using:

{

StandardLibrary::Compiler::FundamentalTypes::UInt32 ::Linked;

StandardLibrary::Compiler::Execution ::ResultCode::Linked;

NotationEngine::Formal::Math::NumberTheory::

{

PrimitiveValues::

{

Native32 ::Source,

Native64 ::Source,

NativeEP ::Source,

PrimitiveValues::Source

},

Value::Source;

};

};

Module(Compiler.Level.1) //Equivalent to the abstraction level of C.

{

Alias(Standard.Version.2019)

{

Append:  
 {

//Standard Library

type UInt32 : SL::C::FT::UInt32 ;

type ResultCode : SL::C::E ::ResultCode;

//Notation Engine

type:

{

Native32 : NE::F::M::NT::PV::Native32,

Native64 : NE::F::M::NT::PV::Native64,

NativeEP : NE::F::M::NT::PV::NativeEP,

Value : NE::F::M::NT ::Value

};

};

Source(Version.1):

{

//Declarations

declare:

{

testUInt32 : static constant UInt32 ,

testNative32 : static variable Native32

}

declare testValue : static variable Value;

//Assignments

testUInt32 = UInt32(0U);

//Functions

function:

{

InitN32(void) -> void

{

testNative32 = Native32.Init(testUInt32);

return;

};

InitVal(void) -> void

{

testValue = Value.Init(testNative32);

return;

};

};

entryPoint execution(void) -> ResultCode

{

InitN32();

InitVal();

return ResultCode::Success;

};

}; //EndSource

}; //EndAlias

}; //EndModule