Subject: Added Google Test
Posted by Klugier on Sun, 24 Jan 2016 13:15:04 GMT
View Forum Message <> Reply to Message

Hello,

Google Test is a unit test framework for C++. More information about library you can find on official GitHub project site: https://github.com/google/googletest.

This is native upp plugin, so it doesn't require any additional dependency. In addition it offers fast creating of test package via TheIDE template mechanism. Currently it's supporting MS Windows and GNU/Linux.

Current u++ port require dependency with Core package. This is because Google Test in one place has got memory leak and I manually turn it off via Upp::MemoryIgnoreLeaksBegin() and Upp::MemoryIgnoreLeaksEnd().

Below is the simply example that shows how to test Upp Core package (bazzar/GoogleTestExample):
StringTest.h:

```
#ifndef _GoogleTestExample_VectorTest_h_
#define _GoogleTestExample_VectorTest_h_

#include <Core/Core.h>
#include <plugin/gtest/gtest.h>

NAMESPACE_UPP

class StringTest : public testing::Test {
protected:
 virtual void SetUp();

protected:
 String sEmpty;
 String sCat;
 String sDog;
};

END_UPP_NAMESPACE

#endif
```

StringTest.h:

```
#include "StringTest.h"
```

```
NAMESPACE_UPP

void StringTest::SetUp()
{
 sCat = "Cat";
 sDog = "Dog";
}

TEST_F(StringTest, TestDefaultConstructor)
{
 String a;

 ASSERT_EQ(a, "");
}

TEST_F(StringTest, TestConstructor)
{
 String a("Test");
 ASSERT_EQ(a, "Test");
}

TEST_F(StringTest, TestGetCount)
{
 ASSERT_EQ(sEmpty.GetCount(), 0);
 ASSERT_EQ(sCat.GetCount(), 3);
 ASSERT_EQ(sDog.GetCount(), 3);
}

TEST_F(StringTest, TestClear)
{
 sCat.Clear();

 ASSERT_EQ(sCat, "");
 ASSERT_EQ(sCat.GetCount(), 0);
}

TEST_F(StringTest, TestCompare)
{
 ASSERT_EQ(sCat.Compare(sCat), 0);
 ASSERT_EQ(sCat.Compare(sDog), -1);
}

TEST_F(StringTest, TestEqual)
{
 ASSERT_TRUE(sCat.IsEqual(sCat));
 ASSERT_FALSE(sCat.IsEqual(sDog));
}
```

END_UPP_NAMESPACE

main.cpp

```
#include <plugin/gtest/gtest.h>

int main(int argc, char *argv[])
{
 testing::InitGoogleTest(&argc, argv);

 return RUN_ALL_TESTS();
}
```

Sincerely,
Klugier
