

DyneTK

Developing Newton applications on Mac OS X
MS-Window and Linux

Matthias Melcher

mm@robowerk.com

WWNC 2007, Tokyo, Japan

. . . and Paul said . . .

„Modernize the development tools“

existing tools

- **„gcc“ tool chain**
 - ◆ **Incompatible in C++**
 - ◆ **No NewtonScript support**
- **Waba, NewtBasic, LittleLisp**
 - ◆ **Not Newton style, still not NewtonScript**
- **The original NTK**
 - ◆ **Does not run on OS X, MS-Windows, or Linux**
 - ◆ **We have no source code for NTK**
 - ◆ **But we have hundreds of functional NTK projects**

Refreshing NTK

How can we get NTK to run on modern systems with new features and with new target platforms?

What is DyneTK

- **DyneTK is a clean-room clone of NTK**
 - ◆ No parts of the original NTK were used
 - ◆ All code is written by observing NTK
- **Fully integrated developer environment**
 - ◆ script editor, visual editor, GUI editor, compiler, debugger
- **Platform independent**
 - ◆ written in C++ with Newt/0 and FLTK for all modern operating systems

DyneTK Goals

- **Fully compatible to existing projects**
 - ◆ **Load any existing project and go**
- **Open Source**
 - ◆ **We, the community, own DyneTK**
- **Open for improvements**
 - ◆ **Change whatever you never liked about NTK**
- **Open for new features**
 - ◆ **Easy to add features like named magic pointers**
 - ◆ **Flexible enough to add new compilers, etc.**

DyneTK preview

Here is DyneTK in action.

DyneTK today

- **DyneTK creates, loads, edits, compiles, packs, uploads, runs, and debugs NTK projects**
- **DyneTK source code compiles and runs on MS-Windows, Mac OS X, and Linux**
- **DyneTK is quite stable, but not ready for production**
- **DyneTK is available with full source code and instructions at Google Code as part of the „DyneE5“ project**

DyneTK technologies

- **DyneTK is based on standard C++**
- **The compiler and interpreter are based on NEWT/0 by Makoto Nukui and Paul Guyot**
- **The user interface is writing in FLTK, a great free cross platform GUI library**
- **DyneTK compiles using gcc or VisualC, Xcode support is easily possible**

The future of DyneTK

- **Implement missing features**
 - ◆ **Proto Templates**
 - ◆ **Sound and Image support**
 - ◆ **Many, many little features and dialogs**
- **Stabilizing and QA**
 - ◆ **Verify with existing NTK projects**
 - ◆ **Fix bugs with the help of users**
- **Direct link to Einstein**
- **New target platforms**

Who can help

- **Do you speak C++?**
 - ◆ **Be the QA department:**
Download and test the existing code.
 - ◆ **Become a developer:**
Add new code, fix bugs.
- **Do you speak NewtonScript?**
 - ◆ **Be a power user:**
Start testing DyneTK with existing NTK projects
 - ◆ **Become a developer:**
Start to rewrite parts of DyneTK in NewtonScript

Thanks!

- **DyneTK would not have been possible without the support of**
 - ◆ **Paul Guyot and Einstein**
 - ◆ **Makoto Nukui and NEWT/0**
 - ◆ **Simon Bell and his Newton code**
 - ◆ **The FLTK team**

and of course

- ◆ **Apple's original wonderful Newton Team**

End of presentation

Questions?

Contact

Matthias Melcher
mm@robowerk.com

Source Code

Google Code, code.google.com, part of „DyneE5“
Links on Sourceforge „DyneE5“