NOVINT

Novint Technologies Developer's Program

- Novint is the pioneer of 3D touch for consumer computing.
- Novint plans to license our 3D touch software to game publishers and developers.
- Novint is actively seeking to acquire, incorporate touch into, or fund the development of a variety of games.

Talk Overview

- Overview
- Software Licensing
- Falcon Overview
- Haptics Programming
- Example: Game Integration (Half-Life 2)
- Example: Game Creation and Game Engines (Newton's Apple)

The Novint Falcon brings our sense of touch to computers

www.novint.com

Novint's 3D Touch Technology

- Novint's Technology fundamentally changes the ways that games are played.
- We have found it is inexpensive to incorporate touch into games, yet the improvement in game play it provides is dramatic.

NOVINT

Benefits

- Easy integration that Novint will complete and/or fund
- Creates more realistic, fun, and immersive applications
- Distinguishes your products in a crowded market
- Can create new revenue for both new and back-catalogue products

Licensing Philosophy

- Our goal is to continually increase the content available for the Falcon
- We will allow free access to our Software Development Kit (SDK) for developers.
- Novint will grant licenses, with a fair and reasonable royalty, for commercial use.

Falcon Overview

The Novint Falcon is the first controller that makes high-fidelity interactive three-dimensional touch possible and practical for consumer computing applications.

Falcon Overview

The Novint Falcon will begin shipping on June 18th, 2007

Preorders are currently available for $189 at www.novintfalcon.com

Supported Platforms

- Launch platform – PC, on Windows XP and Vista
- Anticipate migration to PS3 and XBox 360

API

- C++ SDK overview
- API layers
  - DHDLC (low-level drivers)
  - HDAL (mid-level device communication)
  - Falcon API (high-level programming toolset)
- Allows for integration into existing games and virtually every existing game engine

Questions

Developing with the Novint Falcon

- How does the Falcon work?
  - Device tracks a 3D cursor
  - Cursor encounters a virtual object
  - Currents actuate motors at 1 KHz
- Degrees of Freedom
  - 3DOF input, 3DOF output
  - Additional DOF possible through enabled grips

Programming With 3D Haptics

- Separate haptics and graphics loops
- Single point interaction
- Must Maintain device stability
  - Haptics wall (F = -kx)
- The user feels the sum of all forces

Incorporating Haptics Example: Half-Life 2™

- Summary of Effects
  - Movement Model
  - Vector-based damage
  - Vehicle physics translation
  - Weapon recoil
  - Player physics interpretation (jumping, footsteps, accelerations)
  - Object physics (weight, inertia)

Incorporating Haptics Example: HL2

The first step is to understand the general control loop for I/O devices in the Source engine.

```
void CInput::Init_All (void);

void CInput::ControllerMove( float frametime,
    CUserCmd *cmd );

void Cinput::Shutdown_All(void);
```

Incorporating Haptics Example: HL2

- Trigger haptic events from game events
  - Often can be done with a single function call to an existing Falcon API haptic interaction
- Can take advantage of existing physics calculations for rapid code integration
- Very little or no complex programmer code required for the current integration

Game Development Example: Newton's Apple

- High level overview
  - Shoot an Apple off a monkeys head
  - Allows for revolutionary gameplay centered around a fundamentally new technology
  - Lets you "be the character" and feel what they feel
- Game Engine
  - 3D Game Studio™

Game Development Example: Newton's Apple

- Bind a haptic DLL to the game engine

```
BOOL APIENTRY DllMain( HANDLE hModule, DWORD
   ul_reason_for_call, LPVOID lpReserved)
{
  engine_bind();
  return TRUE;
}
```

- Define your functions in the haptic DLL

```
// Called each graphics cycle to trigger async between threads
DLLFUNC void na1_syncHaptics()
{
    if (haptics::instance())
        haptics::instance()->syncFromServo();
}
```

Game Development Example: Newton's Apple

- Define your C-script DLL functions

```
//General haptic DLL defines
dllfunction na1_createHaptics();
dllfunction na1_startHaptics();
dllfunction na1_syncHaptics();
dllfunction na1_destroyHaptics();
// Get cursor position and Falcon button state
dllfunction na1_getXPos();
dllfunction na1_getYPos();
dllfunction na1_getZPos();
dllfunction na1_isHapticButtonDepressed();

// Newton's Apple specific DLL defines
dllfunction na1_setGrabPos();
dllfunction na1_setBowGrabState(a_state);
```

Game Development Example: Newton's Apple

- Sync with the device every game loop

```
while(1)
{
    na1_syncHaptics();

    // Update the cursor position
    my.x = na1_getXPos();
    my.y = na1_getYPos();
    my.z = na1_getZPos();
    ...
    wait(1);
}
```

- Send a flag to the DLL when clicking and releasing the Falcon button to shoot the arrow

NOVINT

www.novint.com

Questions

Hardware Specifications

• 3D Touch Workspace	4" x 4" x 4"
• Force Capabilities	> 2 lbs
• Position Resolution	> 400 dpi
• Quick Disconnect Handle change time	< 1 second
• Communication Interface	USB 2.0
• Size	9" x 9" x 9"
• Weight	4 lbs
• Power	30 watts, 100V- 240V, 50Hz-60Hz

This presentation contains statements that constitute “forward-looking statements.” Investors are cautioned that forward-looking statements involve risks and uncertainties that could cause actual results to differ materially from the statements. Factors that may cause or contribute to such differences include, among other things, our technology becoming obsolete, consumers not purchasing our products, changes in business conditions and the economy, litigation and other risk factors identified in the Company’s Form 10KSB filed with the Securities and Exchange Commission. The Company undertakes no obligation to update these forward-looking statements for revisions or changes after the date of this presentation.