6800 LEAGUES UNDER THE SEA

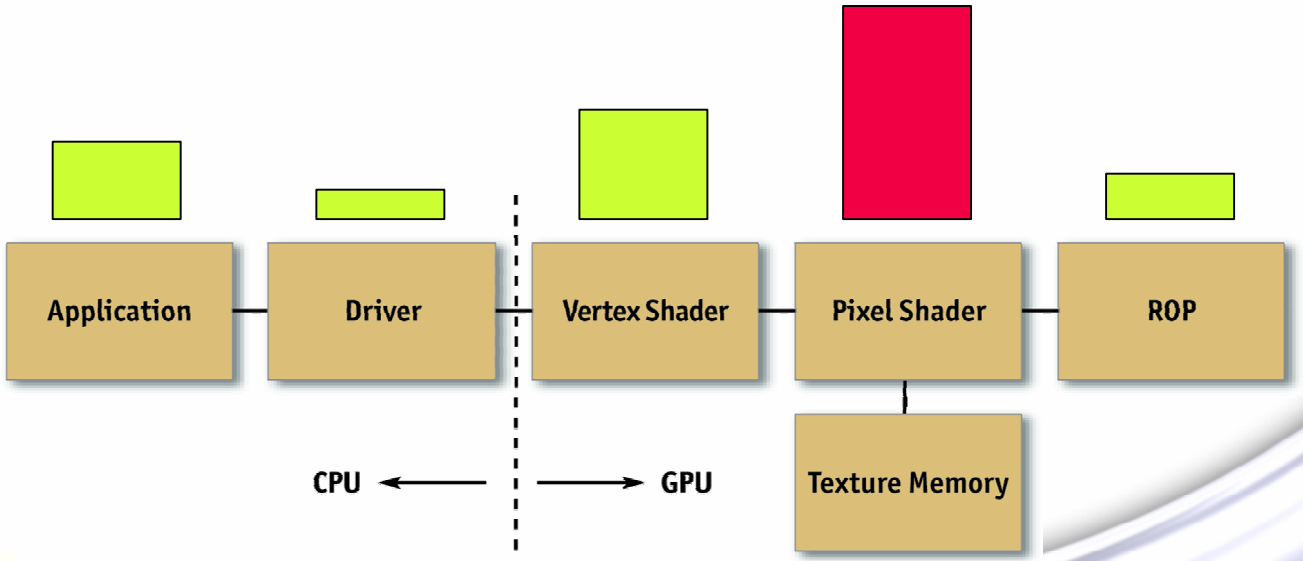
NVIDIA®

NVPerfHUD

Cyril Zeller

What's the Goal?

- Provide a lightweight tool
  to monitor and analyze in real-time the game performance
- For a pipelined architecture, performance analysis means identifying how the various parts of the pipeline influence the frame rate (bottleneck identification)



NVIDIA.

What Is It?

A graphic overlay that displays the evolution over time of basic performance metrics for the game

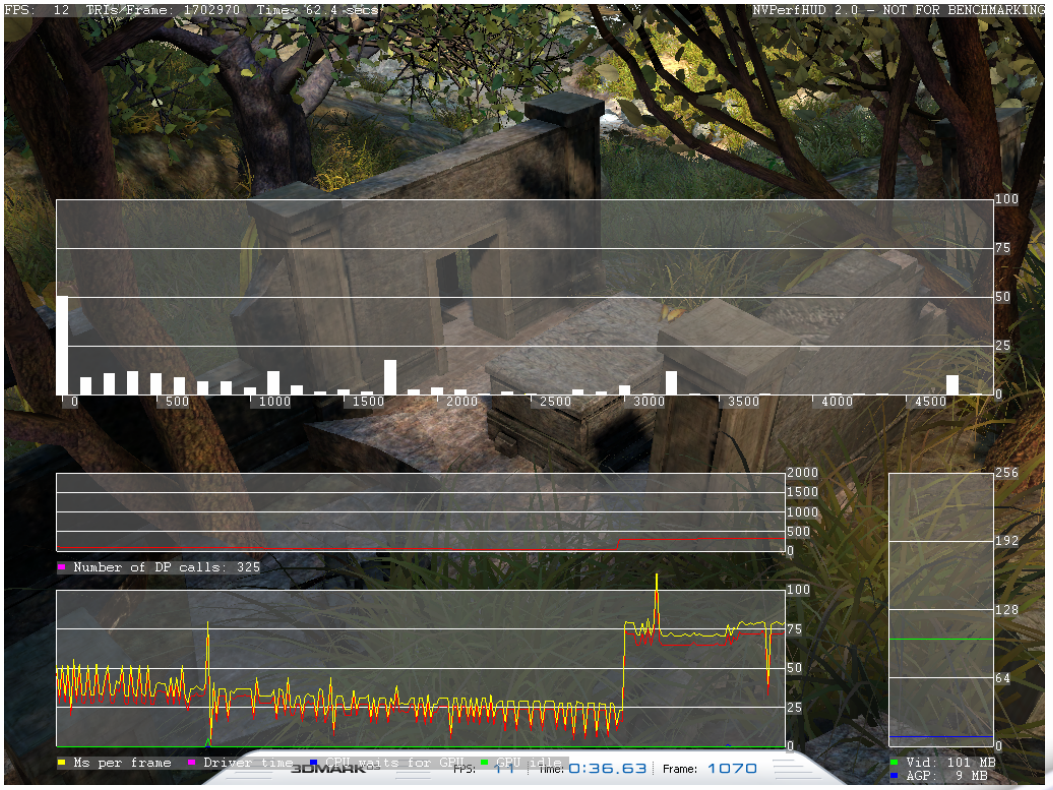


Image courtesy of FutureMark Corp.

NVIDIA.

What Is It?

A set of controls to perform basic experiments at any time

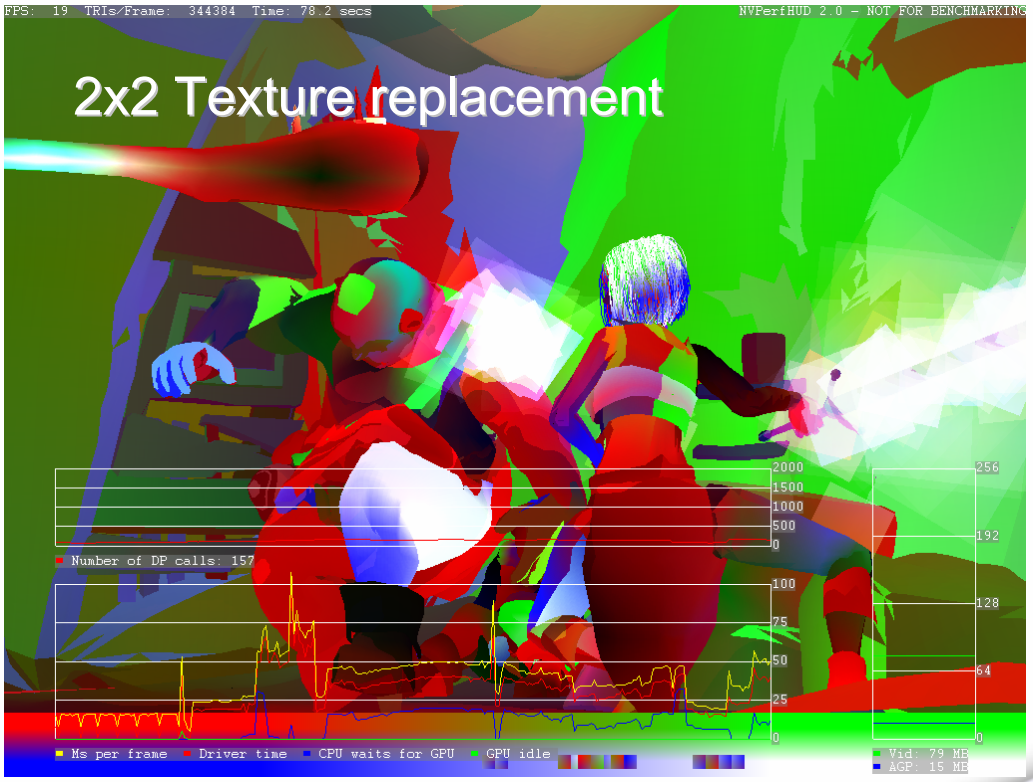
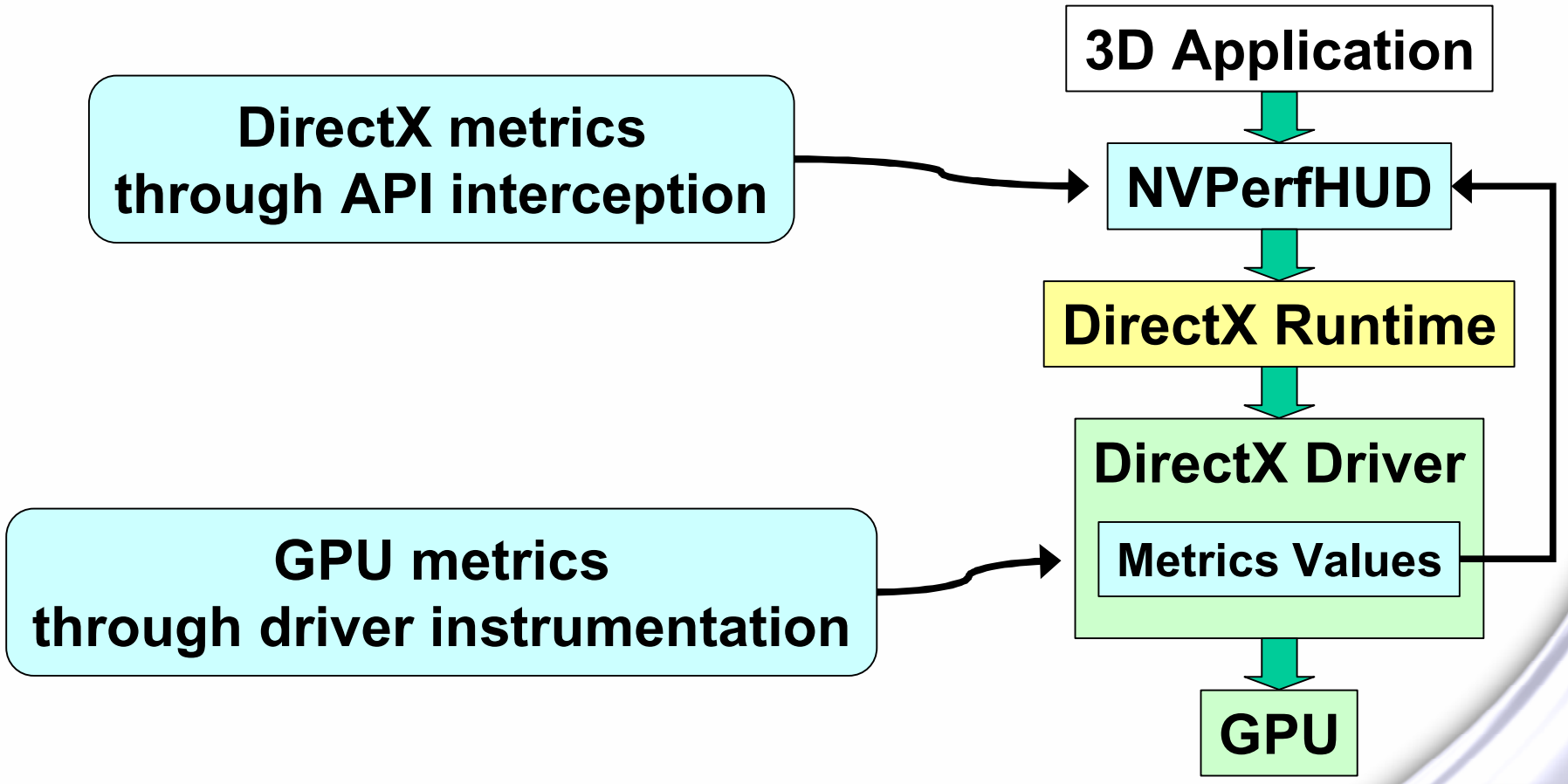


Image courtesy of FutureMark Corp.

NVIDIA.

How Does It Work?



NVIDIA.

What's the Overhead?

- Overlay graph and API interception can cost up to 1.3%
- Driver instrumentation can cost up to 6%
- Upper bound for total cost is 7%

What About Security?

- **Problem:** NVPerfHUD could be used by unauthorized third parties to analyze your application

- **Solution:** To prevent this, NVPerfHUD won't work by default with your application unless you've modified the application to use a device that is:
  - Based on the "NVIDIA NVPerfHUD" adapter
    - This adapter gets created when the application is started through NVPerfHUD
  - And of type `D3DDEVTYPE_REF`
    - The application won't actually used the reference rasterizer if the "NVIDIA NVPerfHUD" adapter has been selected

What About Security?

```
// Set default settings
AdapterToUse = D3DADAPTER_DEFAULT
DeviceType   = D3DDEVTYPE_HAL
#if SHIPPING_VERSION
// When building a shipping version, disable NVPerfHUD (opt-out)
#else
// Look for the "NVIDIA NVPerfHUD" adapter
for each Adapter
    if (Adapter == "NVIDIA NVPerfHUD")
    {
        // If it is present, override the default settings
        // to enable NVPerfHUD (opt-in)
        AdapterToUse = adapter;
        DeviceType   = D3DDEVTYPE_REF;
        break;
    }
#endif
CreateDevice(AdapterToUse, DeviceType)
```

Demo

6800 LEAGUES UNDER THE SEA

NVIDIA.

What's Next?

- NVPerfHUD:
  - Advanced bottleneck analysis
  - Improved GUI
- NVPMAPI:
  - An API to expose driver instrumentation to applications (VTune, PIX for Windows, game engine, ...)
- Learn more at:
  http://developer.nvidia.com/object/nvperfhud_home.html