Providing QoS to a Foreign Personality

Michael Dales

UNIVERSITY of GLASGOW

Thursday 8\textsuperscript{th} July 1999
Motivation

- The *Pegasus* project is developing the *Nemesis* operating system.
- Designed to give QoS to multimedia applications.
- Structured differently to other OSs:
Motivation

- Nemesis uses a different programming model:
  - Uses a Single Virtual Address Space
  - Based on Closures

- Good for achieving Nemesis aims (QoS)
- Bad for support and learning

- Our Work — to port familiar personalities to Nemesis
  - Posix API (X/Open Release 4)
  - Partial X Windows (local machine only)
Issues

• Linkage
  – Moving plain libraries into Nemesis modules
  – Moving shared, stateful APIs to a Single Address Space OS

• Abstraction
  – Maintaining the Nemesis structure under a foreign design
Design - Stateful API

- Allows “traditional” libraries to exist in Nemesis.
- User links against a wrapper. This:
  - Provides Nemesis main function.
  - Initialises Nemesis style modules.
  - Provides static state.
Design - Stateful API

UNIVERSITY of GLASGOW
Computing Science

Nemesis: Cosners'99 © Michael Dales, Thursday 8th July 1999
Design - Stateful API
Design - Posix

- Posix in Unix is a very thin layer on top of the OS.

Must remodel this into a vertical structure.

No data is shared between instantiations.
Design - From X to NemX

- Traditional X design - server does work on data path.

This is a source of QoS crosstalk.

Need to redesign this.
Design - NemX

- Remodelled as client renderer
- Server only used for things like resource allocation.

- Integrates with traditional Nemesis apps too.
Performance - Posix

- Test application: GO simulator playing itself — has deterministic behaviour.
- Linux Results:
Performance - Posix

- Nemesis Results:

![Graph showing performance results with Nemesis/Posix and 4 different tasks over time.](image-url)
Performance - NemX

Resource contention under Nemesis and Linux

- Linux, no contention
- Nemesis, no contention

Time to draw a single point/ms

Frequency/%

0.01 0.1 1 10 100

UNIVERSITY of GLASGOW
Computing Science
Performance - NemX

Resource contention under Nemesis and Linux

Time to draw a single point/ms

Frequency/

Linux, with contention
Nemesis, with contention
Current Status

• So far ported:
  – Mash (Unix shell)
  – Basic Unix commands (ls, cat, grep, etc.)
  – Heretic (X Windows arcade adventure game)
  – XPacMan (X Windows arcade game)
  – mpeg_play (X Windows mpeg video player)
  – and others...

• Currently working on:
  – Posix: Adding BSD style socket support.
  – NemX: Porting the GTK widget set.
Conclusion

- Managed to resolve:
  - Linkage in SASOS issues.
  - Restructuring for Nemesis to preserve QoS.
- Moved apps to Nemesis — demonstrated they gain the stated benefits.