

NetKernel

NetKernel[®] and the Resource Oriented Cloud



<1060[®]>
research

<1060[®]>
research

© 2010, 1060 Research Ltd

NetKernel

What is NetKernel?

True Cloud Computing
Composite Architecture
Application Server

NetKernel

Resource Oriented
Computing Abstraction

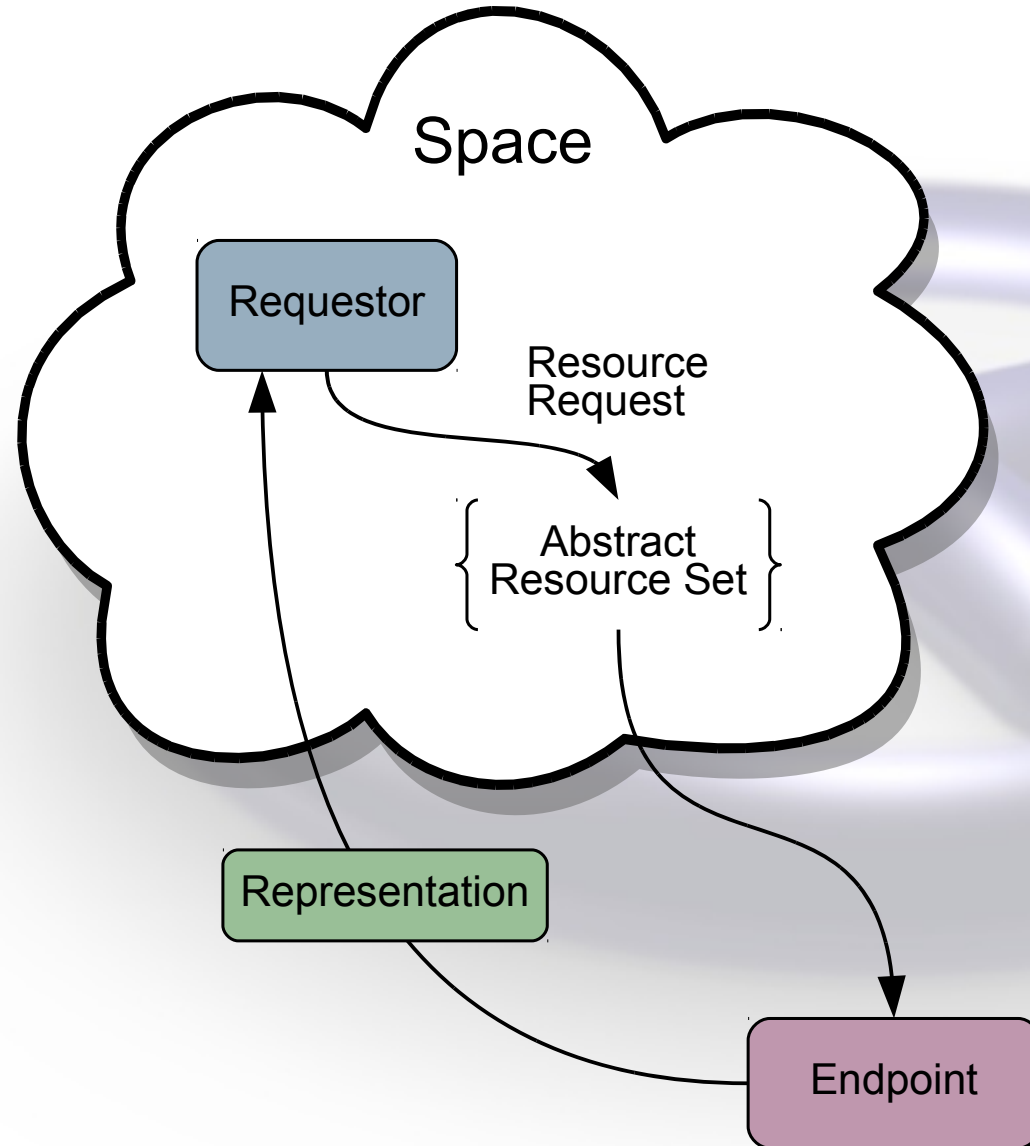
WWW

REST

Unix

>10 years R&D

Resource Oriented Abstraction

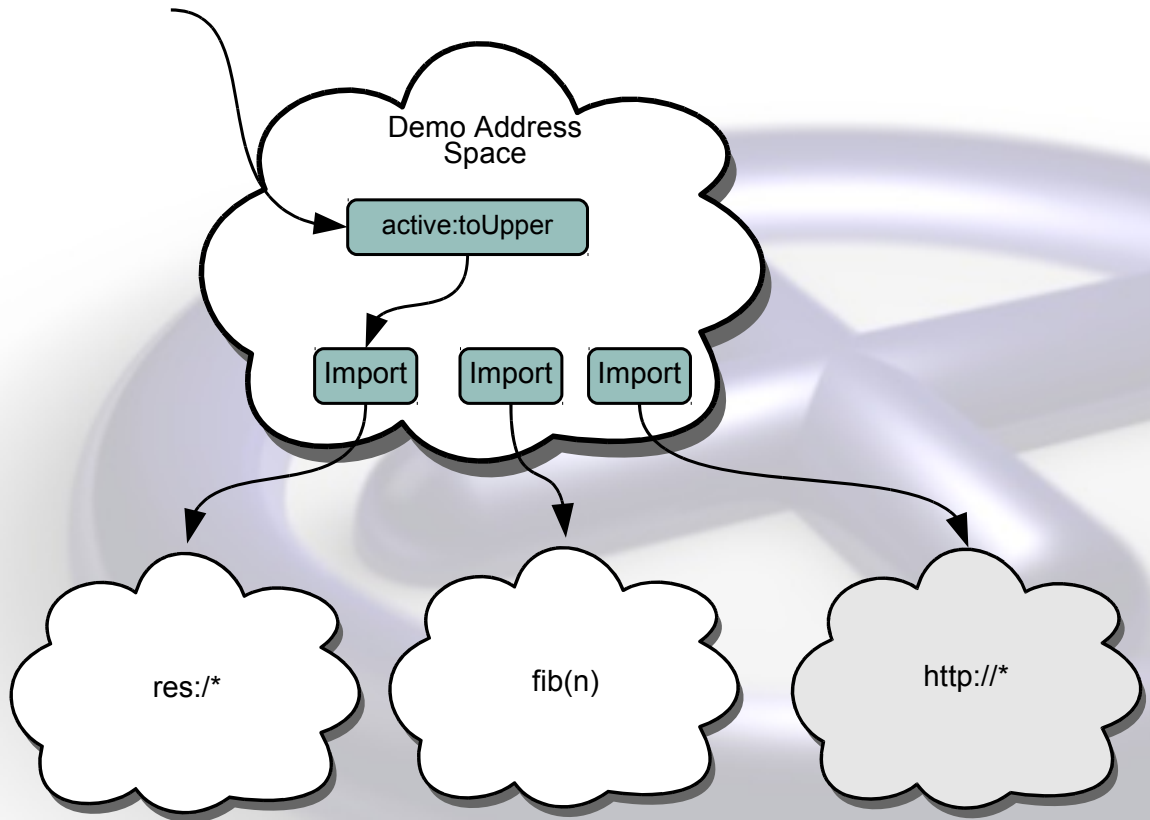


Resource Requests

Demonstration of ROC Concepts

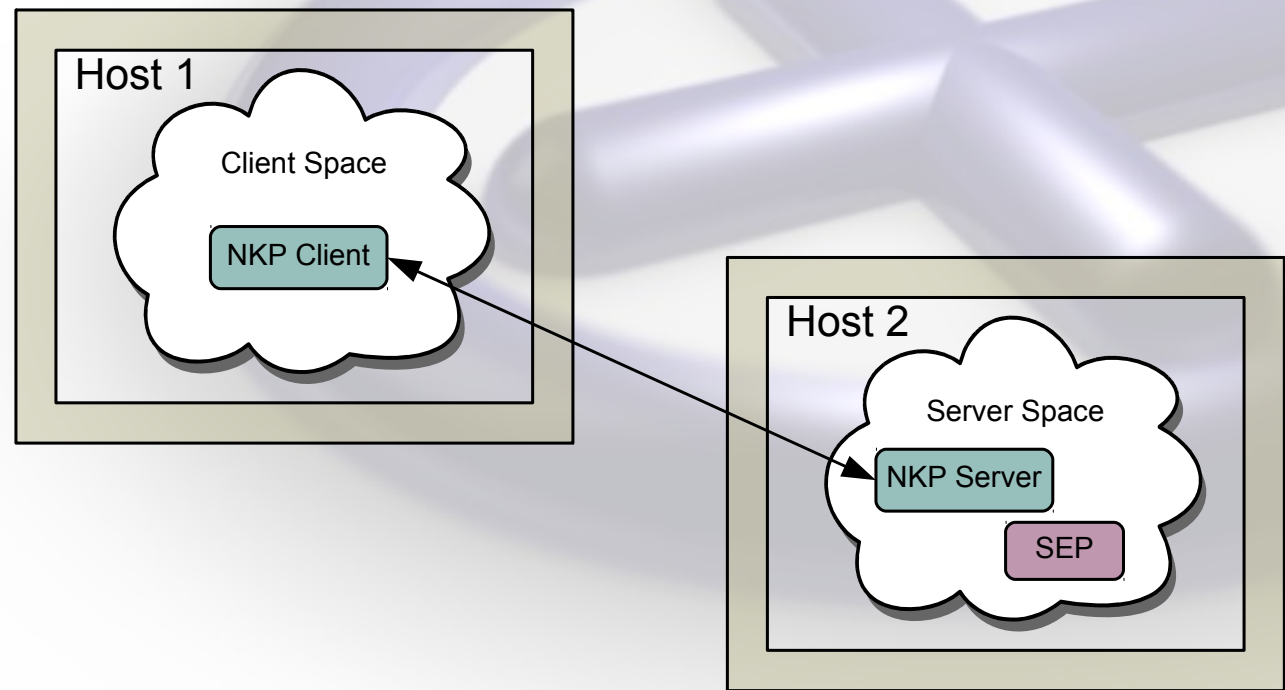
The screenshot displays the NetKernel Demo interface. On the left, a text input field contains the resource request `res:/hello.txt`. Below the input field is an `Issue Request` button. On the right, there are two yellow panels listing presets. The top panel, titled "Presets - Raw Identifiers", lists: `Hello`, `2+2`, `http://www.google.com`, `active:toUpper`, `fib(5)`, `Active URI`, `Active URI (2)`, `Functional URI`, `Abstracted res:/greenbox`, and `DPML RDF-Pipeline`. The bottom panel, titled "Presets - Declarative Requests", lists: `Hello`, `2+2`, `http://www.google.com`, `fib(5)`, `Active URI`, `Literal Arguments`, `Active URI (2)`, `Functional Requests`, `Active Groovy Literal`, and `DPML Literal RDF Pipeline`. At the bottom of the interface, there is a `<NetKernel>` logo with "powered by" text above it, and a copyright notice: `© 2010, 1060 Research Ltd.`

The Web *Inside*



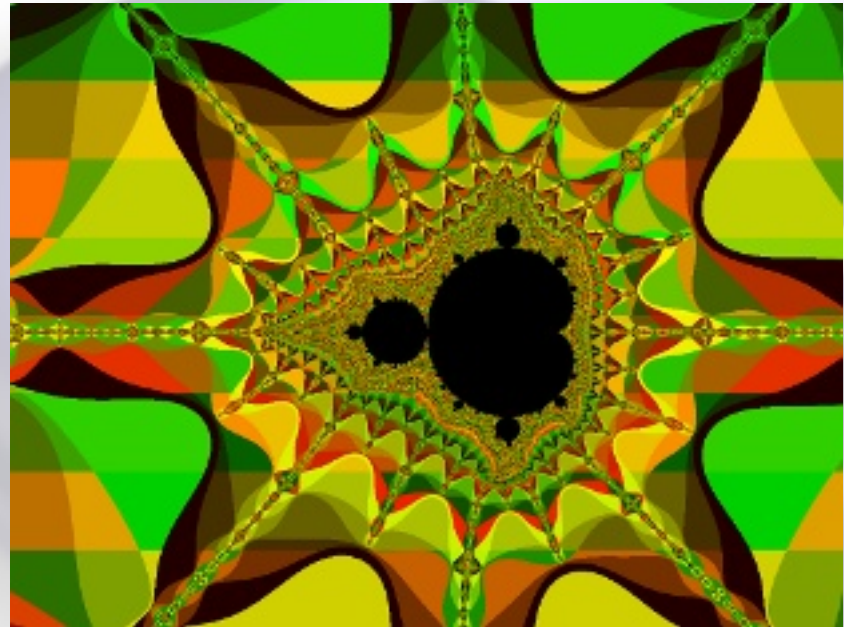
ROC *Outside*

- NetKernel Protocol (NKP)
- Enables NetKernel ROC abstraction to seamlessly span hosts



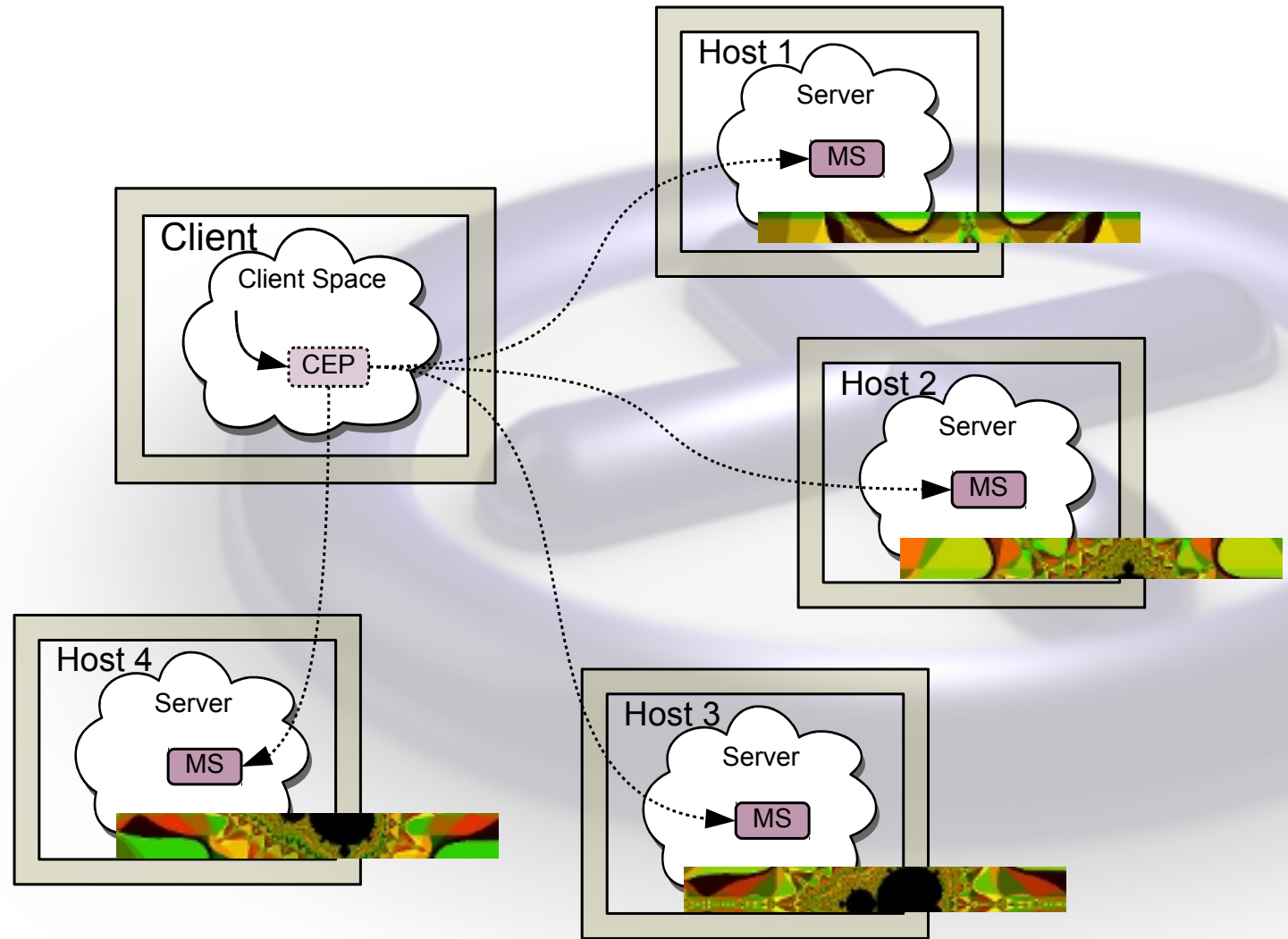
NKP Demo

- Resource Oriented Mandelbrot Set
- `active:mandelbrotStripe`

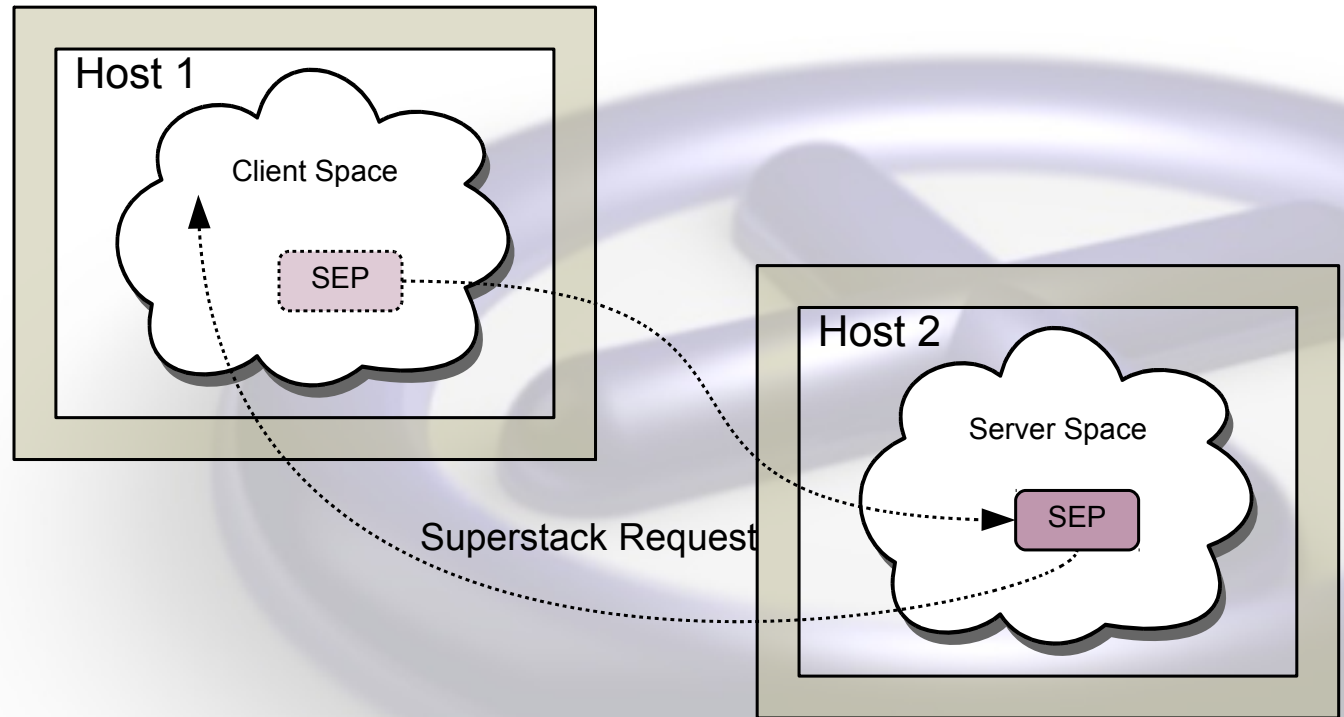


- Async fan-out of stripe computation
 - Scales Locally
 - Scales Remotely

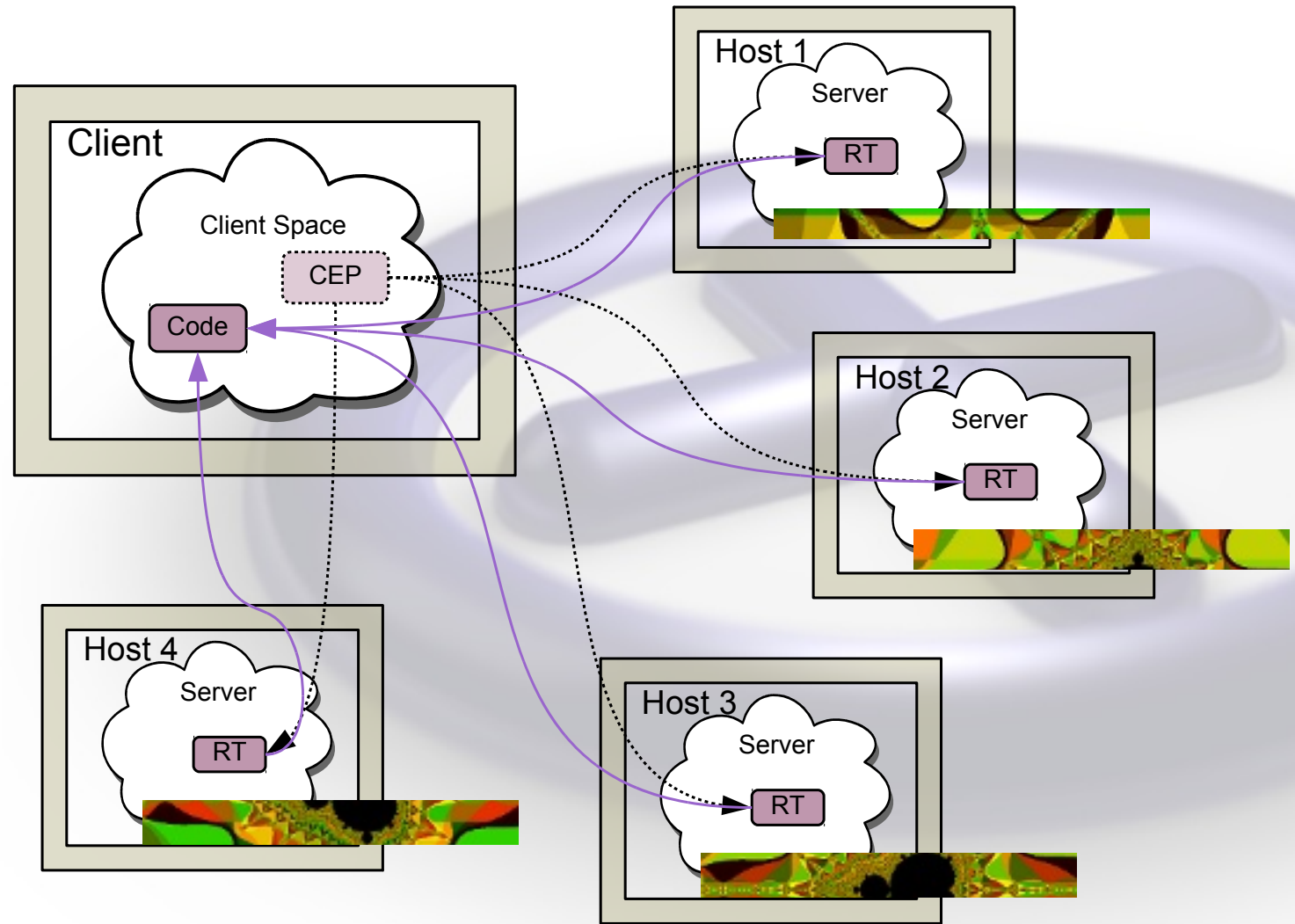
NKP - Balanced OneShot



Client-Server Inversion



NKP - Runtime Cluster



NetKernel Protocol Features

- Asynchronous / Threadless design on both Client and Server
- Throughput Management via Async Throttles
- Pluggable Load Balancing Algorithms
- Oneshot Mode: REST++
- Mount Mode: NFS++
- Dynamically Reconfigurable Client
- Symmetric or Asymmetric Spatial Scope (Superstack)
- Architectural Trust Boundaries
- Distributed Caching and Dependency Model

NetKernel



Why NetKernel?

True Cloud Architecture Platform
Composite Architecture
Application Server
Software for the 21st Century

NetKernel

Resource Oriented
Computing Abstraction

Composable

Flexible

Scaleable

Unbelievable

