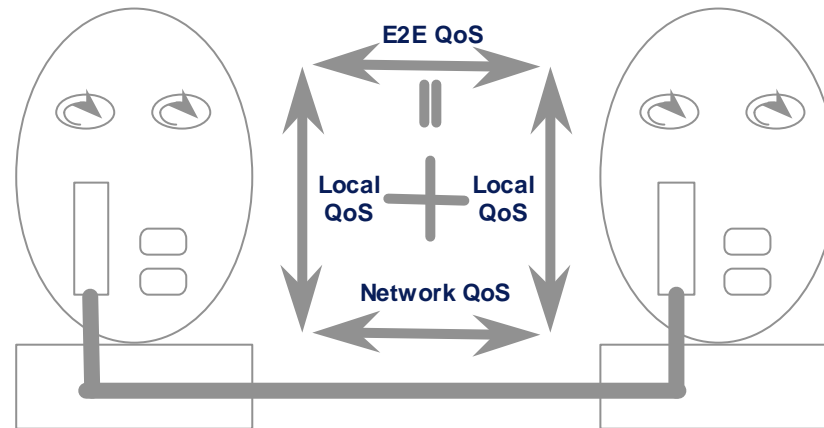


DIMMA Final Report

October 1997



Simon Waterhouse

simon.waterhouse@ansa.co.uk

14 October 1997



Goals

- DIMMA 2.01:
 - Improve performance
 - Analyse (profile) current implementation
 - Eliminate bottlenecks
 - Measure a range of QoS configurations
 - Allow protocol independence
 - for QoS aware RPC applications
- Complete the DIMMA documentation
 - overall design & implementation



Performance Improvements

- Replace iostream marshalling with custom
 - eliminate low level mutex locking
 - run time decisions => compile time (use of templates)
- IIOP optimisation
 - replace marshall/unmarshall of IIOP headers at different protocol layers
- TCP read-ahead
 - minimise 'recv' system call overhead



Other Enhancements

- Re-structure IIOP client session management
 - eliminate context switch for high performance QoS configurations
- Add support for EngineeringQoS to IIOP protocol
 - allow applications to specify QoS requirements in protocol independent fashion
 - defer protocol choice to runtime

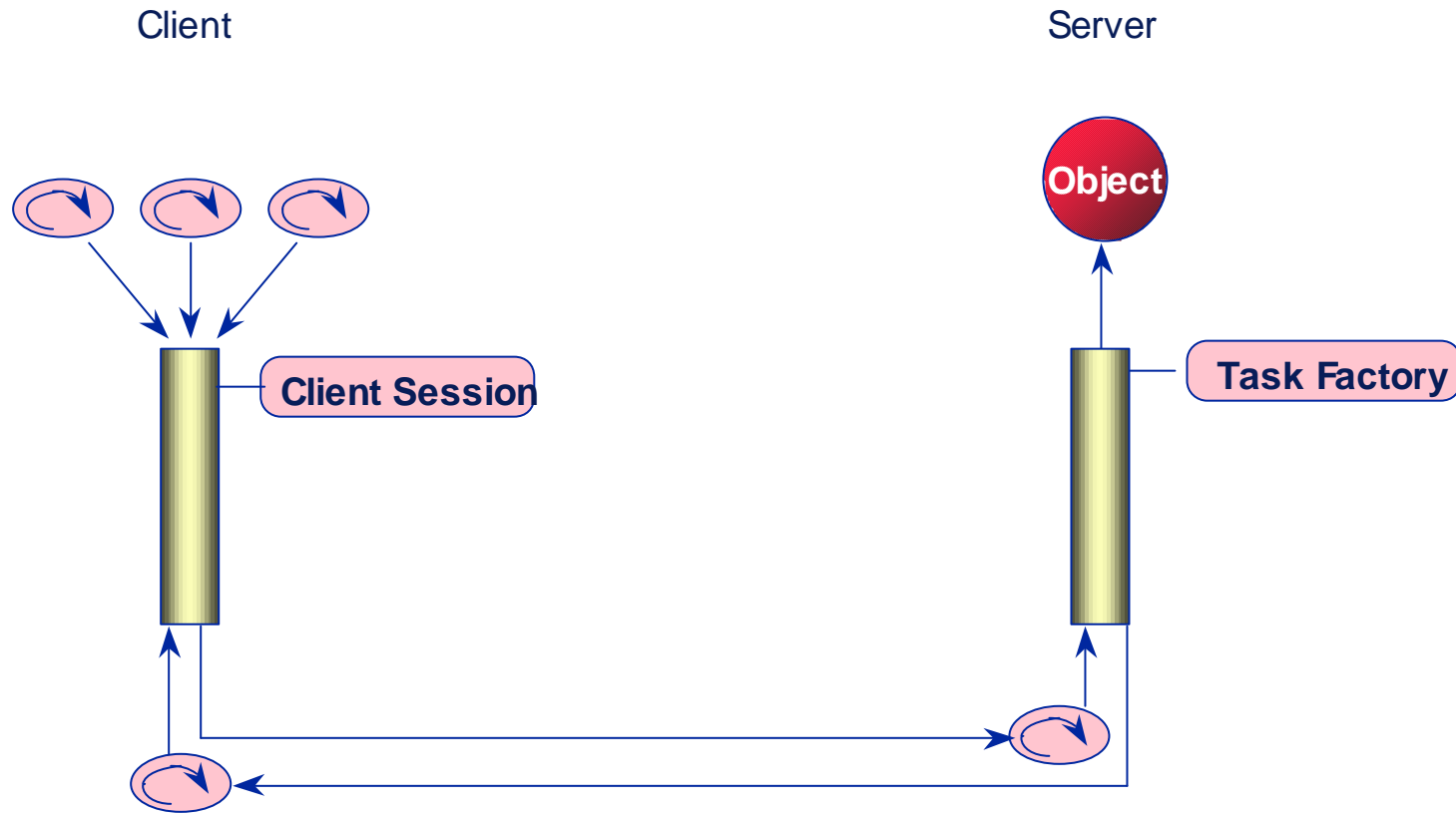


QoS Configurations

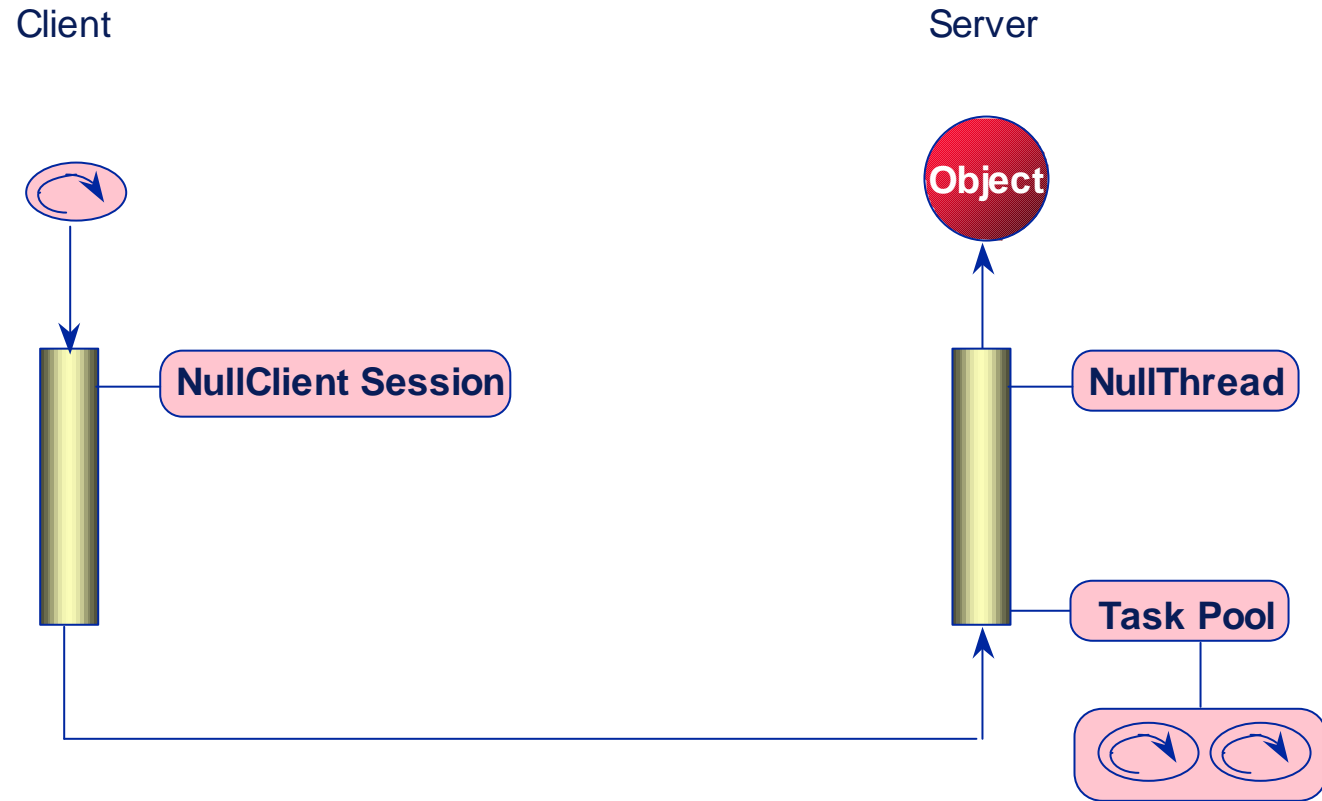
- Measure
 - Default configuration
 - full client & server multiplexing
 - resource factories (resource allocation on demand)
 - “High performance” configuration
 - dedicated threading (minimise context switch)
 - dedicated resource pools (pre-allocate resources)
 - see ‘examples/Time’ for performance test code



QoS - Default Configuration

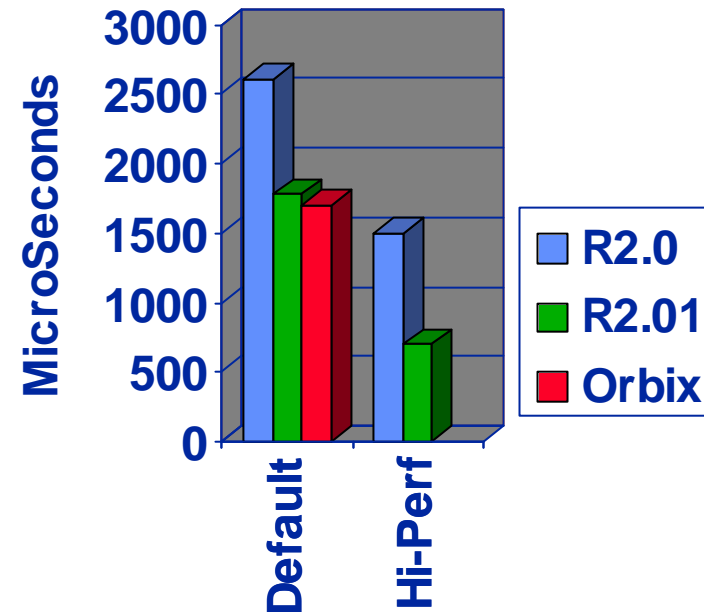


QoS - High performance



Performance Measurements

- Large performance gains in DIMMA R2.01
- Default QoS
 - 50% faster than R2.0
 - comparable with commercial ORBs
- High-performance QoS
 - 2-3 times faster than commercial ORBs
- see APM.2046



Documentation

- Available now from <ftp.ansa.co.uk>
 - Introduction- APM.1995
 - Design and Implementation - APM.2063
 - Performance Analysis APM.2046
 - Build and Installation - APM.2036
 - Writing an application - APM.2037
 - Tracing - APM.1980

