Real-time Distributed Computer Systems 423 (623.423)

1st semester 2003 (previously RTDS 408)

UNIT OVERVIEW

Lecturer & Tutor:

Dr Gary A Bundell, Room 4.12, email: bundell@ee.uwa.edu.au,
Ph: 9380 3815

Text Book:

Gomaa, H. *Designing Concurrent, Distributed and Real-Time Applications with UML*: Addison-Wesley 2000

Recommended Reading:

Burns, A. and Wellings, A. *Real-Time Systems and Programming Languages 2nd ed*: Addison-Wesley 1997
Douglass, B.P. *Real-Time UML*: Addison-Wesley 1998
Gomaa, H. *Software Design Methods for Concurrent and Real-Time Systems*: Addison-Wesley 1993
Laplante, P.A. *Real-Time Systems Design and Analysis - An Engineer's Handbook 2nd ed*: IEEE Press 1997
Levi, S.T. and Agrawala, A.K. *Real-Time System Design*: McGraw-Hill 1990
Naiditch, D. J. *Rendezvous with Ada 95*: Wiley 1995
Gosling, A. *The Java Programming Language*: Addison-Wesley 1996
Orfali, R & Harkey, D. *Client/Server Programming with Java and Corba*: Wiley 1997
Shaw, A.C. *Real-Time Systems & Software*: Wiley 2001

Unit Objectives:

This unit will give you a clear understanding of the *software architectures* of real-time distributed systems, and the principles and practice of their *modelling* and *design*.

Prerequisites:

Advanced Computer Architectures and Computer Operating Systems or equivalents.

Topics:

1. Introduction - terminology and historical perspective of real-time and distributed systems with some examples.

2. Real-time programming languages: requirements and relevant features of Ada and Java.

3. Distributed systems software environments - REC, RPC, NFS, PVM, HTTP/CGI, RMI, CORBA, DCOM, SAP, SOAP approaches.

4. Real-time distributed system design methodologies – RTSAD, DARTS, RT-UML, COMET - case studies.

5. Real-time distributed operating systems: taxonomy, performance issues (benchmarking), real-time kernels, interrupt and exception handling, scheduling & dispatching, synchronisation, exclusion, a typical industrial RTOS (LynxOS), and POSIX real-time extensions and threads.

6. Real-time distributed system performance modelling - real-time scheduling theory and event sequence analysis. Application to task structuring, task clustering and task priority assignment.

7. Time handling, clock algorithms, real-time objects - formal definitions, object-oriented real-time architecture, exceptions, deadlines.

8. Temporal object relations - convex and non-convex time intervals, calendars, constraint propagation.

9. Formal methods in real-time information system modelling and structured software design – Petri nets, timed Petri nets, stochastic Petri nets and simulation; Real-Time Logic.

Tutorials:

Some problems (distributed before tutorial), some design case studies and testing issues discussion, and some video material.

Laboratories:

The laboratories are initially focussed on technology familiarization with CORBA and doing some performance comparisons with related distributed object technologies. This is followed by the development of a typical multiple client-server distributed application which includes a mobile information appliance development component.

A major part of the project is decomposition of the task to group members, systematic development, and then integration of those components into the final application. A test specification is to be constructed to meet the requirements and application of the COMET RTDS design methodology will be required.

Eight × 3 hour sessions are scheduled for laboratories. These are performed in groups of 3/4 students per group. A group lab report is to be produced and the software demonstrated to the test specification.

Assessment:

1. Examination:
   - 3 hours @ 60% of unit mark
2. Assignment:
   - Mobile real-time distributed systems technology trend assessment @ 10% of unit mark
   - Issued 4th week of semester, due 8th week of semester
3. Laboratories:
   - Group lab report @ 30% of unit mark
   - Assessment is moderated with a group assessed contribution weight and an individual report component
   - The laboratories are scheduled for the 5-12th weeks of semester, report due 13th week of semester

Lecture Schedule (N.B. advisory only):

Topic	Lect #	Notes	Text
Unit Overview and Introduction	1,2	in, de	*
Real-Time languages – features (e.g. Ada/Java)	3,4	rl1	**
Distributed Processing Software Environments	5,6,7	dp1-2	4,*
RTDS Design Methodologies: RTSAD, DARTS, OOAD, RT-UML	8,9,10	rm1	*
RTDS Detailed Design Methodologies: COMET	11,12, 13,14	rm2-4	10-14, 16
RTDS Performance Modelling & Analysis	15,16	pm	17,*
Real-time Operating Systems: LynxOS, POSIX, Benchmarking	17,18	rt	4,*
Time Handling	19	th	***
Temporal Object Relations, Time Constraint Projection & Propagation	20	tr	***
Axiomatic Formal Methods in RT Design – Real-Time Logic	21	rf	***
Petri Net Modelling & Analysis	22	pn	*
Performance Modelling with Time-augmented Petri Nets	23	pt	***
Stochastic Petri Nets	24	sp	***
Unit Review & Exam Preparation	25	-	

Notes:

1. Lecture notes letters/numbers are section identifiers, text numbers are chapters in Gomaa, 2000.
2. * From various other texts, papers and vendor literature.
3. ** From Naiditch, D. J. *Rendezvous with Ada 95* and Gosling, A. *The Java Programming Language*.
4. *** From Levi, S.T. and Agrawala, A.K. *Real-Time System Design* plus various papers.
5. Lecture notes are available from the Department in a bound booklet format.
6. All course information, lecture notes, tutorial questions and solutions, some past exam papers, etc, will be available on WWW: *www.ee.uwa.edu.au/~rdcs423*.