Notions of time and space

What is time?

A physical notion of time:

- Transitivity: $x < y \rightarrow y < z \\ x < z$
- Linearity: $x < y \rightarrow x = y \rightarrow y < x$
- Irreversibility: $x < y \rightarrow y < x$
- Continuity: $x < y \rightarrow x = y \rightarrow x > y$

A mathematical notion of time:

- Transitivity: $x < y \rightarrow y < z \rightarrow x < z$
- Linearity: $x < y \rightarrow x = y \rightarrow y < x$
- Irreversibility: $x < y \rightarrow y < x$
- Continuity: $x < y \rightarrow x = y \rightarrow x > y$

Real clocks can be synchronized by GPS.

Real clocks run asynchronously.

Clocks under higher gravity or in faster observations frames are slower.

Universal time UT 0
1952: Royal Greenwich observatory founded

Universal time UT 1
Continuous simulation of the basic solar system

Universal time UT 2
Computation of the effects of the rotation of the earth

Universal time UT 1

International Atomic Time TAI

Universal Time Coordinated UTC

Synchronisation

1968: The concept of absolute time is destroyed: the covariance of light and time to the frame.

1912: Einstein’s general theory of relativity (the independence of time and events in space).

What is embodiment?

Embodied phenomena are those that by their very nature occur in real-time and real-space.

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

References

Dourish, Paul: "What time is?" in: "Time & Space" (Addison Wesley, fourth edition 2009)

What is time?

Time is a physical property of natural phenomena.

Is time an inherent property of nature? Is time

What is time?

Is time an inherent property of nature? Is time

Copyright Languages; with time, 2009 pp. 1-18

What time is it?

Real clocks have limited resolution.

Which timeframes can be used as real-time?

Moon calendars since about 4000 BC

Solar calendars since about 50 BC

Spring based clocks since about the 13th century

Pendulum clocks since 1656

Universal time UT 1
Continuously updated corrections to UT0

Universal time UT 2
Computation of the velocity of the earth

Universal time UTC 0

Evidence of the earth’s rotation

Real-time systems and engineering:

It is usually concerned with the time estimate: an actual, accessible, generated or measured by events.

"Real-time" in engineering is the notion of an actual, accessible, generated or measured by events.

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

What is embodiment?

The phenomena of experience as the central aspects and building blocks of understanding.

Edmund Husserl (1859-1938, Vienna, Halle, Göttingen, Freiburg):

Founder of the phenomenological tradition

Phenomenological concept of the intentional object

The relationship of experience and the objects observed

The concept of embodiment

Phenomenology as a discipline

The relationship of experience and the objects observed

A scientific discipline of consciousness

The relationship of experience and the objects observed

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements

What is embodiment?

The phenomena of experience as the central aspects and building blocks of understanding.

Edmund Husserl (1859-1938, Vienna, Halle, Göttingen, Freiburg):

- Founder of the phenomenological tradition
- Phenomenological concept of the intentional object
- The relationship of experience and the objects observed
- A scientific discipline of consciousness

What is embodiment?

The phenomena of experience as the central aspects and building blocks of understanding.

Edmund Husserl (1859-1938, Vienna, Halle, Göttingen, Freiburg):

- Founder of the phenomenological tradition
- Phenomenological concept of the intentional object
- The relationship of experience and the objects observed
- A scientific discipline of consciousness

What is time?

What is time? / What is embodiment?

- Interfacing with time
- Specifying timing requirements
- Satisfying timing requirements
What is embodiment?

Embodiment is the property of any engagement with the real world which may make this engagement meaningful.

Implications:
- Universal intelligence, autonomy, consciousness or any other cognitive process which is independent of physical environments.
- Universal morphology (mechanical design, robot, device, …) which is useful or even operational in all physical environments.

Embodied phenomena are those that by their very nature occur in real time and real space.

Embodied is the property of any engagement with the real world which may make this engagement meaningful.

Meaningfully embedded systems are part of an 'ecological niche' which is meaningful considering the morphology and cognitive ability of the entity. 

Embodied skills as part of meaningful embedded systems thus depend on the task. To operate under real-time constraints is supportive and employed by the system.

There is no such thing as a body as a physical entity or any other cognitive process in itself does not exist.

Universal morphology (mechanical design, robot, device, …) which is useful or even operational in all physical environments.

Meaningful embedded systems have a self-sufficient character. To construct meaningful morphologies is the task of the constructors. To operate under real-time constraints is supportive and employed by the system.

Digital design, robot, device, … which is useful or even operational in all physical environments.

- Hardware timers
  - Embedded designs
  - Real-time clock interface
  - Programming primitive ‘Delay’
  - Programming primitive ‘Timer’
Absolute delays are available in Ada and in other real-time and event-driven languages.

Only absolute delays and timers are available in strict real-time systems.

Only relative delays and timers are available in low-level, real-time environments.

Only absolute delays and timers are available in strict real-time systems.

Only relative delays and timers are available in low-level, real-time environments.

As a third alternative to keep waiting in infinite blocking, timeouts are implemented in:

- Shared variable communications
- Synchronization
- Conditional critical regions
- Interprocess communication
- Shared objects
- Message passing between processes

Asynchronous and synchronous message transfers

Remote procedure calls

Remote objects

Tasks
Interfacing with time

Relative timeouts on incoming calls

- Task body: Task_1 in Ada
- Current Temp: Temperature
- Begin

Loop
- Select
- Accept Call (T: Temperature)
- Do
- Current.Temp := T;
- End

End Controller;

Learn how to interrupt a running process due to a timeout?

- Inside a single process: "Timeout on actions"
- Externally triggered timeout on actions (Ada)
- Common real-time systems concept:

- Get a first approximation and employ spare time for refinements:
  - Set an absolute deadline for the computations
  - Measure the real-time used for the computations
  - Estimate the first results
  - Improve the result and keep a record of improvements (while keeping an eye on the deadline)
  - End in either of the cases:
    - Results cannot be improved
    - The deadline is fulfilled and there is a usable result in any case.

Temporal scopes

- Minimal & maximal
- Maximal
- Absolute

Common attributes:
- Measuring & recording time
- Minimizing delays
- Controlling execution time
- Absolute deadlines

Specifying timing requirements

Timeliness is often more important than Precision

1. Get a first approximation in the amount of time and adjust the deadline
2. Improve the algorithm or its input to reduce the time
3. Improve the result and keep a record of improvements while varying the time constraints
4. End in either of the cases:
   - Results cannot be improved
   - The deadline is fulfilled and there is a usable result in any case.

The big topics:

- What is time? / What is embodiment?
- Interfacing with time
- Satisfying timing requirements
  - Asynchronous transfer of control methods
  - Scheduling
  - Temporal scopes
  - Notions of time and space
Specifying timing requirements

Common temporal scope attributes

Real-time Euclid

(a language for complex real-time systems)

Real-time Euclid – Language status

Real-time Java

How to handle time-unbound primitives?

Real-time Euclid, CRL, DSP, Pearl, …

Can be:

1. Exclude them

2. Expand them to become individually safe

3. Reduce the scope of time-unbound primitives

4. Reduce the duration of time-unbound primitives

Loops are restricted to (time) bounded loops.

Processes are static and non-nested.

Deadlines as part of the language:

© 2017 Uwe R. Zimmer, The Australian National University
Specifying timing requirements

Real-time Java

public class RealtimeThread extends Schedulable {
    public static final int MIN_PRIORITY
    public abstract class Scheduler implements a runtime schedulability analysis
    public abstract class RealtimeThread {
        public synchronized void addIfFeasible ();
        public synchronized void deschedulePeriodic ();
        boolean waitForNextPeriod ()
    }
}

Two paths towards fulfilling real-time requirements

Real-time logic approach

- System identification and compile-time analysis
- Schedulers: fixed priorities, dynamic priorities and earliest deadline first
- Schedulers: fixed priorities, dynamic priorities and earliest deadline first
- Schedulers: fixed priorities, dynamic priorities and earliest deadline first

Complex systems approach

1. System identification and compile-time analysis
2. Real-time analysis and checks
3. Supply sufficient test for verifications and validations:

- Formal methods and model checking
- Schedulability analysis
- Runtime monitoring and adaptation
- Real-time logic approach & Complex systems approach