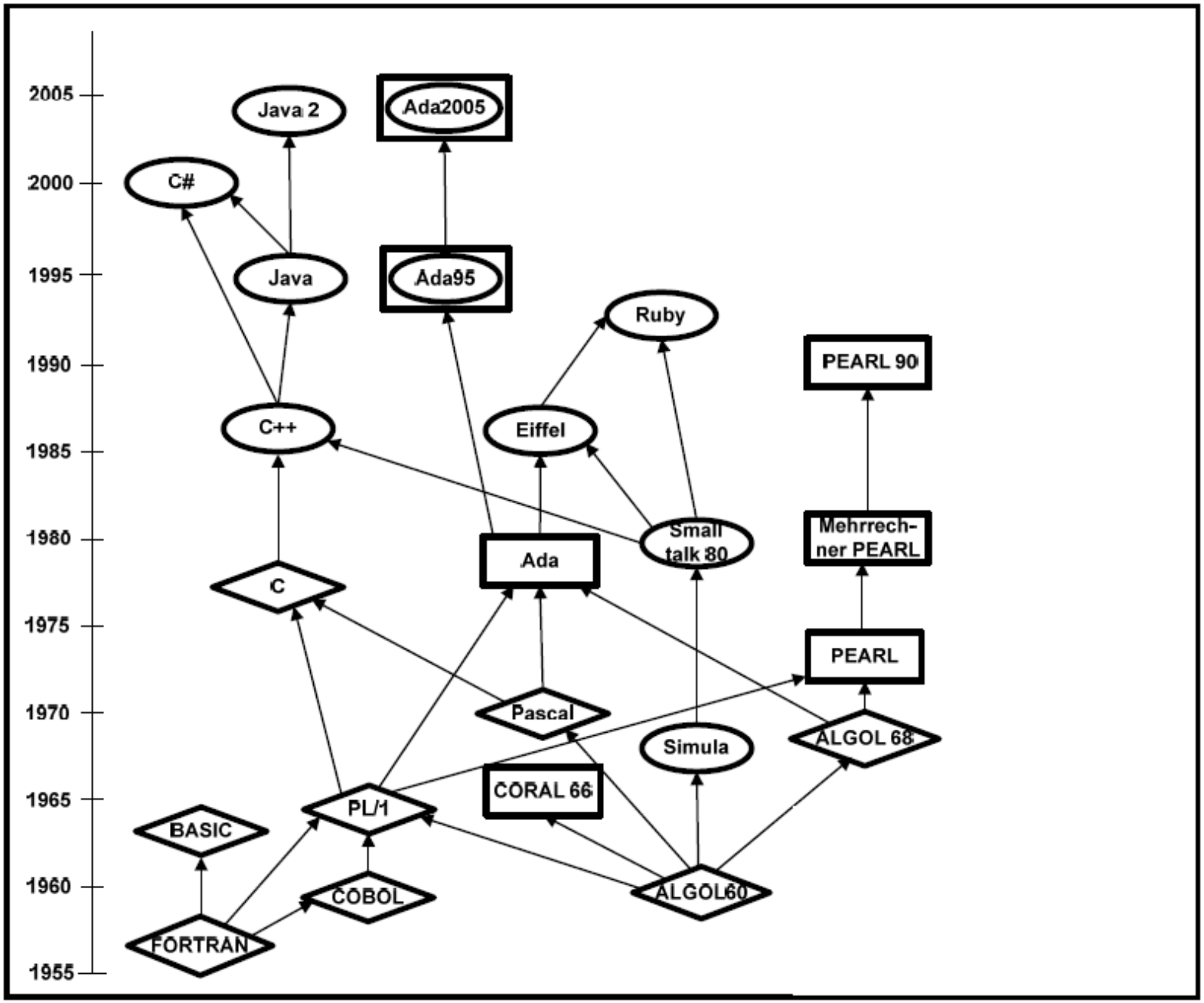
Real Time Systems – SS2016

Prof. Dr. Karsten Weronek
Faculty 2
Computer Science and Engineering

Real-time Programming Language

Networks for Realtime

Languages for RT



C and C++

Efficient machine-oriented system programming language

Flexible comparable to Assembler

Control flow like in higher programming languages

Universal applicability

Concurrent C: Extension of C with concepts for Real-Time execution.

C and C++ have no real-time constructs!

A real-time operating system is mandatory!

Calling of OS-functions from C-program:

Real-time library is needed!

Eligibility of C and C++ for real-time programming:

FRANKFURT UNIVERSITY OF APPLIED SCIENCES

- Very popular language for Real-Time systems.
- Very good support with professional compilers and tools
- Good programming environments.
- Compiler for most micro-controllers
- Support for Real-Time-Operating-Systems, such as
  - QNX, OS9, RTS, VxWorks, etc.

- Attention when using object-oriented constructs:
  - non-deterministic runtime behaviour
  - Bad memory usage

- Therefore we use C in the exercises, however
  - this doesn't exclude C++ for RTS

Field of application:

- rapid prototyping especially for client/server-architectures
- Multimedia applications (video, sound, animation)
- Intra-net-applications

Problems:

- Memory management (garbage collection)
- High memory usage (especially RAM)
- Bad runtime behaviour

Extensions for Java to support Real-Time behaviour:

1. Scheduling
   Real-Time scheduler with guarantied deadlines
2. Memory management
   Extension of the storage model
   to support deterministic behavior
3. Synchronization
   Specification of scheduling algorithms, which support
   "priority inheritance\ and "priority ceiling\-methods
4. Asynchronous event handling
   Warranty, that the programs support massive parallel events
   (up to several 10000).

If there are only few critical sections in a program

Java (with RT-extension) has more a advantages than disadvantages.

Some authors are convinced that Java-RT is very useful for RTS.

At the end of the day:

It depends on the specific requirement.

Process and Experiment Automation Real-Time Language

All basic data types and language constructs of other procedural programming languages are available in PEARL.
PEARL oers comfortable language constructs for realizing multitasking- and Real-Time duties.

Special Data types:

- CLOCK, DURATION
  data types for describing time events and durations.
- INTERRUPT
  connection to hardware interrupts.
- SEMA
  Semaphore-Variables are used for synchronization of different tasks.
- BOLT
  Bolt variables support concurrent read access on data, which is impossible with Semaphore variables.

Block structure, availability of objects:

- Object declarations inside BEGIN-END-Blocks
- Procedure- and function- wide objects
- Module- wide objects
- Access of objects from other modules using global
- decelerations of data
- Announcement of objects for other modules using
- global decelerations

- Multitasking - instructions
  - ACTIVATE Taskname
    Starts a task with the name Taskname immediately.
  - TERMINATE Taskname
    Task with the name Taskname is terminated.
  - SUSPEND Taskname
    Task with the name Taskname is suspended.
  - CONTINUE Taskname
    Suspended task with the name Taskname is continued.
  - PREVENT Taskname
    Task with the name Taskname, which is activated by an event will be descheduled.
    The task will not be activated by the event.

Planing of events and point in time

The activation and continuation of tasks may also be scheduled by external events or point in times. Its also possible to schedule a xed point of time

or interrupts.

Examples:

ALL 0.00005 SEC ACTIVATE Highspeedcontrol;
cyclic activation of controls with a frequency of 20 kHz
AT 12:00 ALL 4 SEC UNTIL 12:30 ACTIVATE LB PRIO 1;
cyclic schedule, between 12:00 and 12:30 with high priority every 4 seconds.
WHEN fire ACTIVATE firefighters;
activation of task firefighters if the interrupt re is raised.

Task synchronization to avoid dead locks

- A synchronization of tasks is always need, if data is used by several tasks.
- Using semaphor- and bolt-variables tasks can be synchronized.
  If Task A wants to access data,
  which is used by Task B , Task A is blocked until
  Task B will release data.
- For semaphores additionally a non-blocking test is possible (TRY).

Ada

Ada is a structured, statically typed, imperative, wide-spectrum, and object-oriented high-level computer programming language, extended from Pascal and other languages.

It has built-in language support for design-by-contract, extremely strong typing, explicit concurrency, offering tasks, synchronous message passing, protected objects, and non-determinism.

Ada improves code safety and maintainability by using the compiler to find errors in favor of runtime errors.

Ada is an international standard;
the current version (known as Ada 2012)
is defined by ISO/IEC 8652:2012.

Owing to time constraint RT-Networks and RT-buses are introduced in detail.

However:

What you need to know from the guest lecture is:

When using Ethernet for RT purposes you have to make shure the you don't get retries on the network. This can be achieved by isolating the network nodes in single segments. The term is: Microsegmentation!