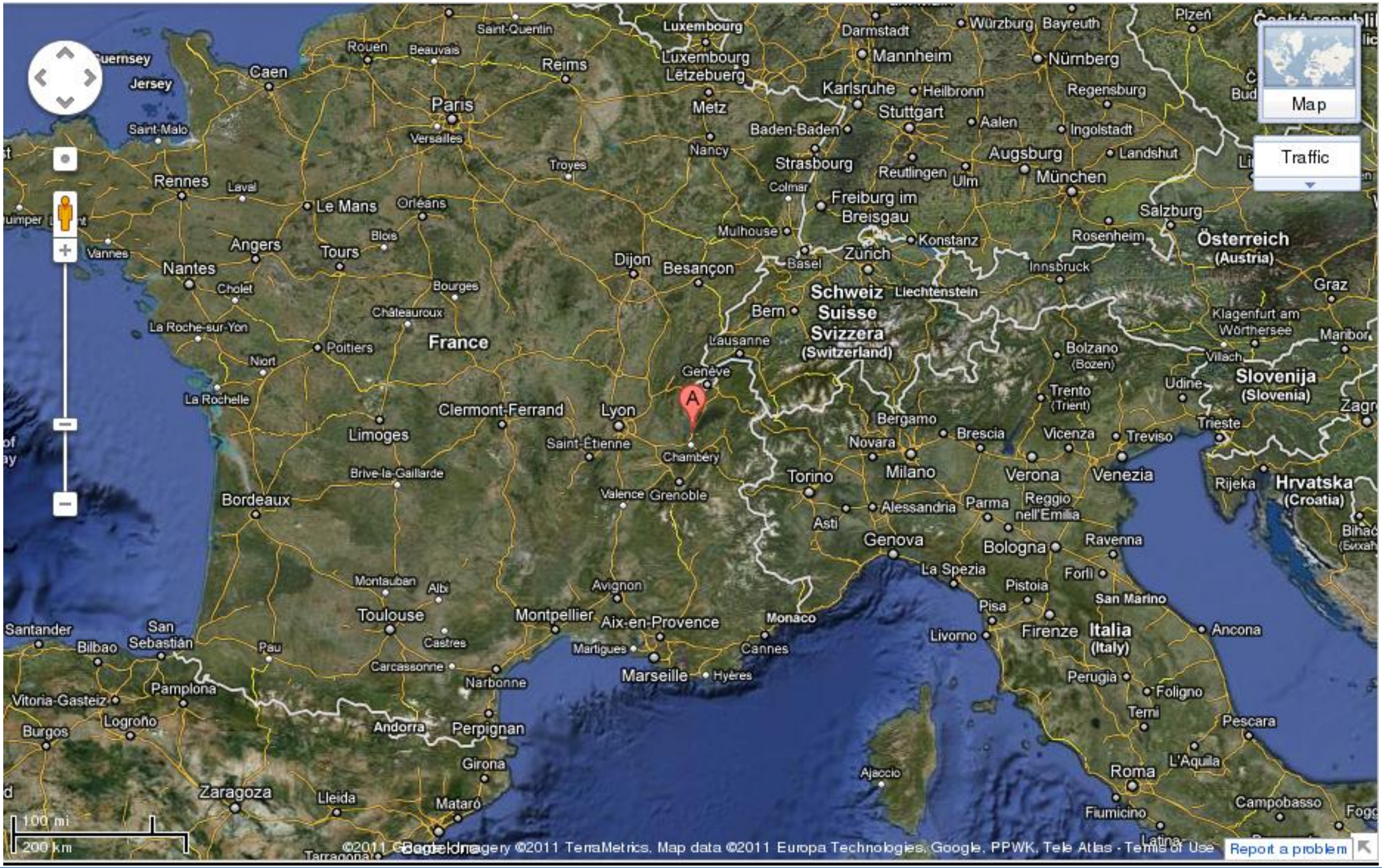
ARTIST Summer School Europe 2011

Mac Mollison

Real-Time Systems

at the University of North Carolina at Chapel Hill

Location


Paris
Versailles
Rouen
Beauvais
Saint-Quentin
Reims
Caen
Jersey
Saint-Malo
Rennes
Laval
Le Mans
Orléans
Troyes
Nancy
Metz
Luxembourg
Lëtzebuerg
Darmstadt
Mannheim
Würzburg
Bayreuth
Nürnberg
Karlsruhe
Heilbronn
Stuttgart
Aalen
Regensburg
Ingolstadt
Baden-Baden
Strasbourg
Colmar
Reutlingen
Ulm
Augsburg
Landshut
München
Freiburg im Breisgau
Mulhouse
Basel
Zürich
Konstanz
Salzburg
Rosenheim
Innsbruck
Österreich
(Austria)
Graz
Klagenfurt am Wörthersee
Maribor
Villach
Slovenija
(Slovenia)
Udine
Trieste
Rijeka
Hrvatska
(Croatia)
Angers
Tours
Blois
Vannes
Nantes
Cholet
La Roche-sur-Yon
Bourges
Châteauroux
Poitiers
Niort
La Rochelle
France
Dijon
Besançon
Schweiz
Suisse
Svizzera
(Switzerland)
Liechtenstein
Bern
Lausanne
Genève
Bolzano
(Bozen)
Trento
(Trient)
Clermont-Ferrand
Lyon
Saint-Étienne
Chambéry
Limoges
Brive-la-Gaillarde
Valence
Grenoble
Bergamo
Brescia
Vicenza
Treviso
Novara
Milano
Verona
Venezia
Torino
Asti
Alessandria
Parma
Reggio nell'Emilia
Genova
Bologna
Ravenna
La Spezia
Forlì
Pistoia
Pisa
San Marino
Livorno
Firenze
Italia
(Italy)
Ancona
Perugia
Foligno
Terni
Pescara
L'Aquila
Roma
Fiumicino
Campobasso
Bordeaux
Montauban
Albi
Toulouse
Castres
Carcassonne
Narbonne
Avignon
Montpellier
Aix-en-Provence
Monaco
Cannes
Martigues
Marseille
Hyères
Ajaccio
Santander
Bilbao
San Sebastián
Pau
Vitoria-Gasteiz
Pamplona
Logroño
Burgos
Andorra
Perpignan
Girona
Zaragoza
Lleida
Mataró
Map
Traffic
100 mi
200 km
©2011 Google - Imagery ©2011 TerraMetrics, Map data ©2011 Europa Technologies, Google, PPWK, Tele Atlas - Terms of Use
Report a problem

Pictures

Pictures

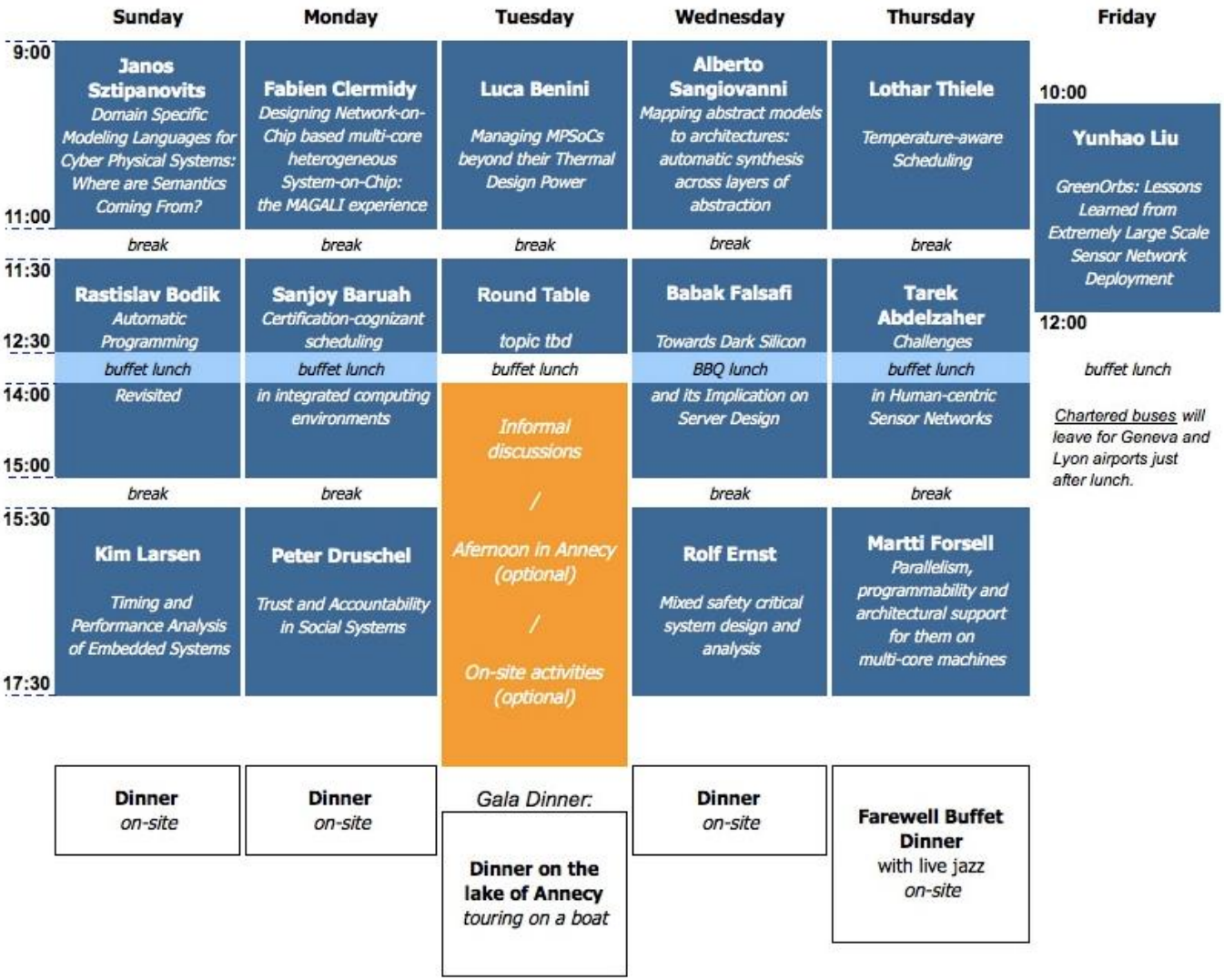
Pictures

Pictures

Organization

- Sponsored by *ArtistDesign European Network of Excellence on Embedded Systems Design* (“Artist”)
- Artist is part of the official EU scientific research program

Format


Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
9:00
11:00
11:30
12:30
14:00
15:00
15:30
17:30
Janos Sztipanovits
Domain Specific Modeling Languages for Cyber Physical Systems: Where are Semantics Coming From?
Fabien Clermidy
Designing Network-on-Chip based multi-core heterogeneous System-on-Chip: the MAGALI experience
Luca Benini
Managing MPSoCs beyond their Thermal Design Power
Alberto Sangiovanni
Mapping abstract models to architectures: automatic synthesis across layers of abstraction
Lothar Thiele
Temperature-aware Scheduling
10:00
Yunhao Liu
GreenOrbs: Lessons Learned from Extremely Large Scale Sensor Network Deployment
12:00
buffet lunch
Chartered buses will leave for Geneva and Lyon airports just after lunch.
break
Rastislav Bodik
Automatic Programming
Revisited
Sanjoy Baruah
Certification-cognizant scheduling
in integrated computing environments
Round Table
topic tbd
Babak Falsafi
Towards Dark Silicon
and its Implication on Server Design
Tarek Abdelzaher
Challenges
in Human-centric Sensor Networks
buffet lunch
BBQ lunch
Informal discussions
/
Afternoon in Annecy (optional)
/
On-site activities (optional)
Kim Larsen
Timing and Performance Analysis of Embedded Systems
Peter Druschel
Trust and Accountability in Social Systems
Rolf Ernst
Mixed safety critical system design and analysis
Martti Forsell
Parallelism, programmability and architectural support for them on multi-core machines
Dinner
on-site
Gala Dinner:
Dinner on the lake of Annecy
touring on a boat
Farewell Buffet Dinner
with live jazz
on-site

Domain-Specific Modelling Languages for CPS

- Janos Sztipanovits, Vander***bilt***
- Interesting historical explanation of CPS
  - Engineers used to just build complex systems using physical/mechanical components. Now, they *also* use computers and networks.

Continues on next slide

Domain-Specific Modelling Languages for CPS

- Uses domain-specific languages to model *structure* and *behavior* in a system
  - Domain examples: Electrical, thermal, mechanical, computational
- Uses meta-languages to compose the domain-specific languages and prove properties of the system

Automatic Programming Revisited

- Rastislav Bodik, UC Berkeley
- The "dream" of writing programs that write programs for us hasn't played out
- Let's do something less ambitious: programmer gives a specification with "holes" in it, and the computer fills them in

Continues on next slide

Automatic Programming Revisited

- SKETCH is a tool to do this

SKETCH: just two constructs

spec:

```
int foo (int x) {
    return x + x;
}
```

sketch:

```
int bar (int x) implements foo {
    return x << ??;
}
```

result:

```
int bar (int x) implements foo {
    return x << 1;
}
```

Timing and Performance Analysis of Embedded Systems

- Kim Larsen, University of Aalborg (Denmark)
- UPPAAL timed automata modelling tool
  - Check schedulability for task systems
    - (Up to a certain level of complexity)
  - Can also be used to compute WCET given a hardware model
  - Can also be used to model other properties of complex systems
- I was quite impressed by the capabilities of this tool!

Heterogeneous Network-on-Chips

- Fabien Clermidy, French Center for Atomic Energy
- This was my first in-depth exposure to the network on chip concept
  - Many cores (e.g. 64+) connected by a network (e.g. with routers) on a single chip

Continues on next slide

Heterogeneous Network-on-Chips

- Interesting claims
  - NoC will soon be common in the embedded market; Intel is already going this way, plus many others
  - NoCs will be highly heterogeneous
- His job is actually to design chips, so I'd take his claims very seriously.

Certification-cognizant scheduling

- Sanjoy Baruah, UNC
- OCBP for one-shot and recurring jobs
- New uniprocessor mixed criticality scheduling algorithm, EDF-MD
  - Improvement over OCBP, but for a more limited task model (the Liu & Layland sporadic model)
- Still many open problems (e.g. shared resources)

Trust and Accountability in Social Systems

- Peter Druschel, founder of MPI-SS
- This was a distributed/networked systems talk
- Replace *fault tolerance* with *accountability*
  - Especially makes sense for systems with humans in the loop
  - Library exists: "PeerReview"
    - Used in an accountable email system, and other things

Continues on next slide

Trust and Accountability in Social Systems

- Sybil (forged identity) detection/tolerance schemes
  - Using “credit network” formalism

Managing MPSoCs beyond their Thermal Design Power

- Luca Benini, University of Bologna
- "1,000-core NoC chips will be here in 3-4 years"
  - He designs these chips for industry

Continues on next slide

Managing MPSoCs beyond their Thermal Design Power

- Current/future chips will melt if allowed to run full throttle for more than a few seconds
- Two approaches: Thermal-aware scheduling and hardware control
  - He discusses only the latter
  - Idea: adjust volatage, frequency, and other "knobs" at runtime based on control theory and sophisticated models

Round Table

- Included Joseph Sifakis
  - shares a 2007 Turing Award for work in model checking
- Mainly just rehashed points brought up in the lectures.

Mapping models to architectures using automatic synthesis

- Alberto Sangiovanni-Vincentelli, UC Berkeley
- Interesting discussion of auto market
  - Growth is in electronics
  - Approaching 100 ECUs per car
    - Now moving from federated to integrated architecture (e.g. AUTOSAR)
  - "I predict that in 5 years you can buy a car that will drive autonomously (because I have seen one)."

Continues on next slide

Mapping models to architectures using automatic synthesis

- Automatic synthesis
  - Synthesis is kind of like compilation, but takes a more abstract domain model and produces code optimized for that domain
  - Example: "Electronic design automation" industry (integrated circuit design)
    - This speaker co-founded Cadence AND Synopsys!

Continues on next slide

Mapping models to architectures using automatic synthesis

- "There are 100 million LOC in a car today"
  - Most in entertainment system
  - "80% of code in an automobile is auto-generated"
- Wants to apply integrated circuit logic synthesis techniques to other areas (e.g. automobile software)
- Mapping synthesized software onto multicore platforms is going to be a big challenge.

Towards Dark Silicon for Servers

- Babak Falsafi, EPFL (Switzerland)
- "Dark silicon" = most of the platform is intended to be kept powered down most of the time, for energy/thermal reasons
- For server workloads, optimal caching policy is specific to the exact part of the workload under consideration

Continues on next slide

Towards Dark Silicon for Servers

- NoC caches should use separate infrastructure (for example, on-chip routers) for cache *requests* and cache *responses*
  - Because requests are small, responses are large

Mixed safety critical design and analysis

- Rolf Ernst, TU Braunschweig (Germany)
- Solving the same problems Sanjoy does, but by creating special (very predictable) hardware instead of special scheduling algorithms.
  - To start with, the hardware is done in FPGA
- Basic idea: Use a different core for each criticality, and use very specialized NoC to make sure they can't interfere

Temperature-aware Scheduling

- Lothar Thiele, ETH Zurich (Switzerland)
- The goal is to prevent system from going over some maximum temperature
- He presented two techniques
  - Both take a partitioned scheduling approach
  - I will gloss over them for time's sake

Continues on next slide

Temperature-aware Scheduling

- First approach
  - Add idle time to the beginning of each task
    - This allows the CPU to cool off
  - Uses real-time calculus to compute the amount of idle time needed for each task.

Continues on next slide

Temperature-aware Scheduling

- Second approach
  - Uses on online “shaper” to insert tiny bits of idle time into tasks dynamically (online)

Human-centric Sensor Networks

- Tarek Abdelzaher, UIUC (USA)
- Discusses GreenGPS system
  - Finds "shortest path" for *fuel consumption* of automobiles
  - Based on sensor data collected from people's engines and GPS

Continues on next slide

Human-centric Sensor Networks

- In traditional control applications, the goal is to control the "plant" (e.g., speed of the car for cruise control)
- Speaker views *people* and *society* as the plant for many future applications
  - GreenGPS is a good example
- Because this requires collecting lots of data from people, he uses innovative statistical techniques to ensure data remains anonymous.

Architectural support for parallel programming

- Martti Forsell, VTT (Finland)
- Most of this talk was just graduate parallel programming
  - e.g. explaining the PRAM (parallel RAM) abstraction for parallel programs

Continues on next slide

Architectural support for parallel programming

- His overall thesis
  - PRAM and similar abstractions are the right way to make parallel programming "easy"
  - However, there is a huge gap between this kind of model and the way computers *actually* work (architecturally)
  - He believes the solution is not to change our models or to abandon them, but to *design architectures to match the models*
    - That's what he's working on

Extremely Large Scale Sensor Network Deployment

- Yunhao Liu, Tsinghua University (China)
- Coal Mine Monitoring
  - Detect dangerous conditions (e.g., low oxygen, flooding) in a coal mine
- OceanSense
  - Sensors deployed on buoys (i.e., floats) to detect depth of sea
  - Helps authorities determine where to dredge (dig out) sea floor to make it passable for large ships

Continues on next slide

Extremely Large Scale Sensor Network Deployment

- GreenOrbs
  - Heterogeneous nodes (for long-range comm., or for sensing)
  - Largest number of hops is 28
  - Sensors detect fire, canopy coverage, and measure carbon dioxide emission of trees
  - Designed for long-term deployment (ideally 3-10 years)

Continues on next slide

Extremely Large Scale Sensor Network Deployment

- CitySee
  - WSN to monitor conditions in a city
  - 1,100 nodes measuring temperature, humidity, light, etc.
  - Future work: Deploy 4,000 sensors covering 100 square kilometers
- WSN security using techniques from topology (a branch of mathematics)

Overarching Themes

- 1,000-core embedded systems really are coming soon
- Thermal-aware techniques are mandatory to deal with the thermal wall
- Formalisms for dealing with complex and safe software are needed
  - Mixed criticality
  - Model-based design

The End

For slides (and, eventually, videos) of the lectures:

http://www.artist-embedded.org/artist/Overview,2278.html