Panel on
FUTURE COMPUTING/BUSTECH/COMPUTATION TOOLS

Tools and Theory: Drivers Spectrum for Future Computing

Panel on FUTURE COMPUTING/BUSTECH/COMPUTATION TOOLS

Tools and Theory: Drivers Spectrum for Future Computing

Panelists:

- **Rudolf Berrendorf**, Bonn-Rhein-Sieg University, Germany (Moderator)
  Computer Simulations

- **Lorenzo Bettini**, University Florence, Italy
  Statically vs. dynamically typed languages in relation to IDE tooling

- **Kendall Nygard**, North Dakota State University, USA
  Security issues, especially for mobile platforms

Panel Discussion

- First, panelists introduce themselves,
- then they present their points of view,

- and then, the panel takes this up for a fruitful discussion.

- Everybody in this room is invited to participate in the discussion

- Rudolf Berrendorf
- Bonn-Rhein-Sieg University,
  Sankt Augustin, Germany
- Full Professor in Computer Science
- Head of Scientific Computing Platform
- Research Interests:
  - parallel program development
  - (parallel) program optimizations
  - parallel data structures and algorithms

Tools and Theory:
Drivers Spectrum for Future Computing

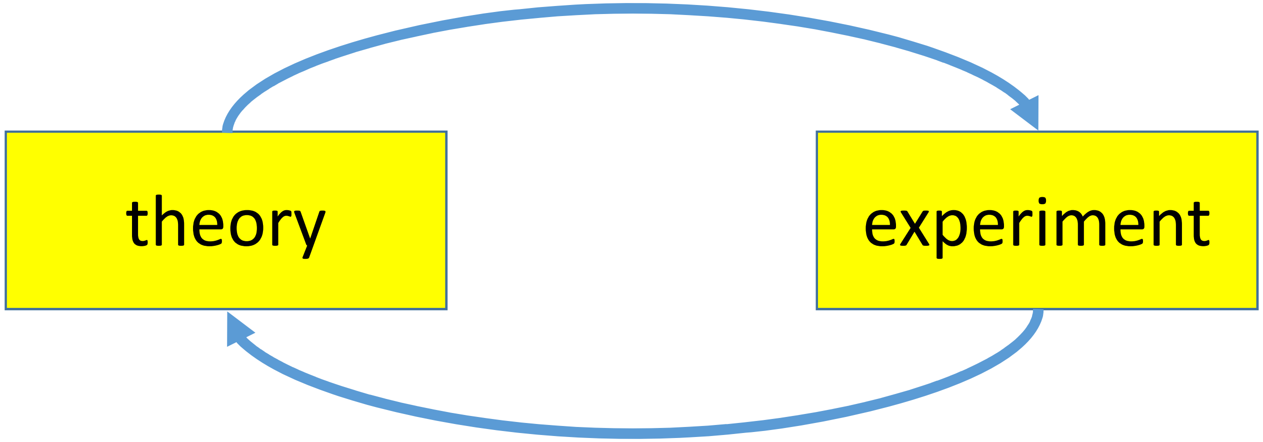
- What will computing be used for in the future?

- How can computing help us to do things we could not do before?

- What is the relation to theory?

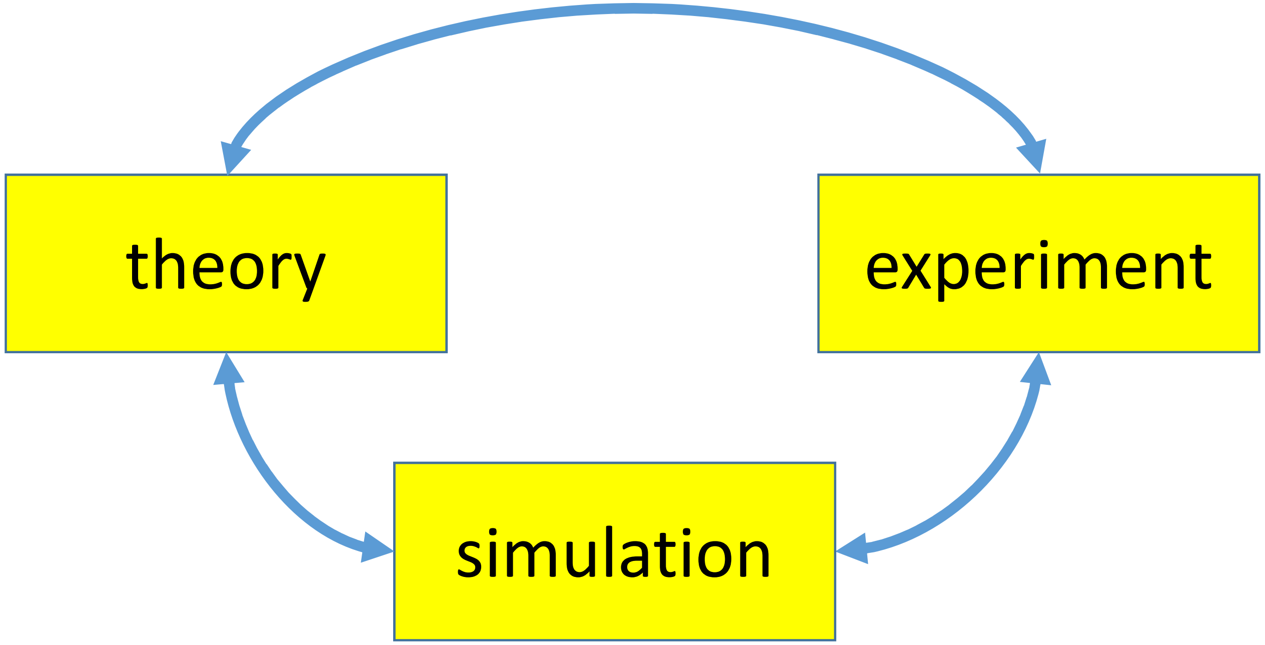
Theory – Experiment

- Traditionally,
  - an experiment was used to verify a theory
  - a theory was developed to abstract from experiment data
- Both supplement each other



Theory – Experiment - Simulation

- A third scientific method in use since several decades is the simulation on (large) computers.



Computer Simulations

- Simulations of 1D / 2D / 3D geometrical regions, e.g., oil reservoirs, earthquake, wings of an aircraft, crash simulations, acoustic absorption in cars

- Simulations not geometry related, e.g., equities / stock trading, bio sequencing, brain simulation

- Most common to all simulations of practical relevance: you can either have accurate answers or fast answers, both not both together
- Therefore, simulations at any time are often restricted by the available computational power that is available and time constraints

Example Car Manufacturer

- To develop one model over 4 years, 250 different simulation tools were used
- During the development time, in average 1000 simulations per week were run
- The simulations took up to 1 week per run

Future Simulations Enable New Insights

- Future computers will have much more computational power
- This allows us in our simulations
  - to be more accurate (~ higher resolution) for better quality
  - to have more complex models (instead of an aircraft wing simulate the whole aircraft)
  - to process larger amounts of data to find relations (Big Data)

- Some computational methods currently in use do not scale with that, which asks for new methods (theory!)

Simulation ↔ Theory

- New methods must
  - have a low computational complexity (otherwise they often will have no practical relevance)
  - work parallel (computational demands and future architectures require that)
  - be scalable with more parallelism (future degrees of available parallelism will be higher than today)
  - be scalable with more data (e.g., iterative linear solvers for $N=10^{12}$)

Summary

- Much more raw computational power will be available in the future
- Computer simulations get faster (reduction in development time)
  and/or more accurate (higher data quality)
- But not all currently available methods are suited for that

Statically vs Dynamically typed languages (and their IDE tooling)

Lorenzo Bettini
Professor in Computer Science
at Dipartimento di Statistica, Informatica, Applicazioni –
Università di Firenze

Research Interests

- Design, Theory and Implementation of programming languages
- Type Theory
- OO language extensions
- Java-like languages
- And their IDE (with Xtext)

Static Type Systems

- Define the *Static Semantics* of programs
- Based on well-known logic theories
- Provide some static guarantees
  - Necessary
  - Not Sufficient

Type Soundness & TDD

- The dynamic semantics respects the static semantics
- A well-typed program cannot “go wrong” w.r.t. to types

- **Test Driven Development**
  - Complementary to the type systems
  - Again, not sufficient but kind of necessary

IDE support

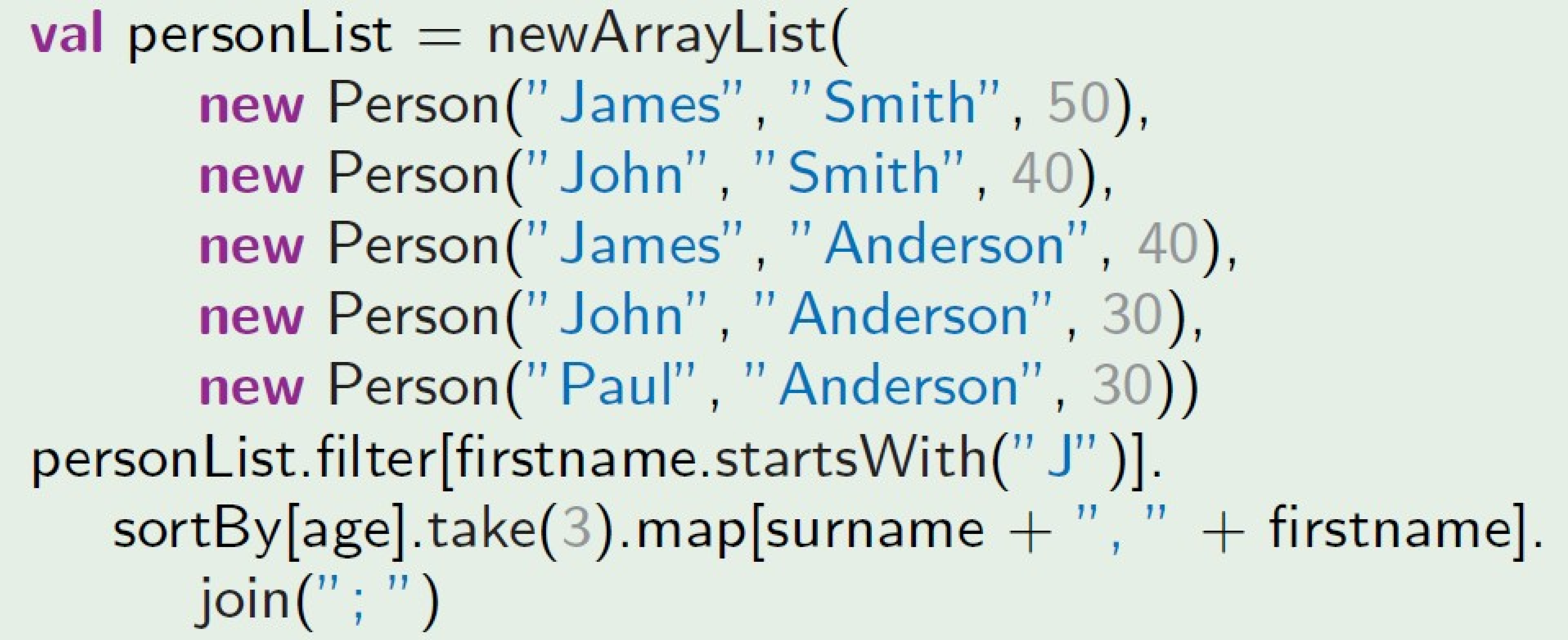
- If a language is statically typed:
  - Easier “navigation” to definitions
  - Very useful code completion proposals
  - Sensible refactoring

Worried about verbosity?

- Implement type inference
  - No need to write types
  - And still enjoy static guarantees

Example: Xtend

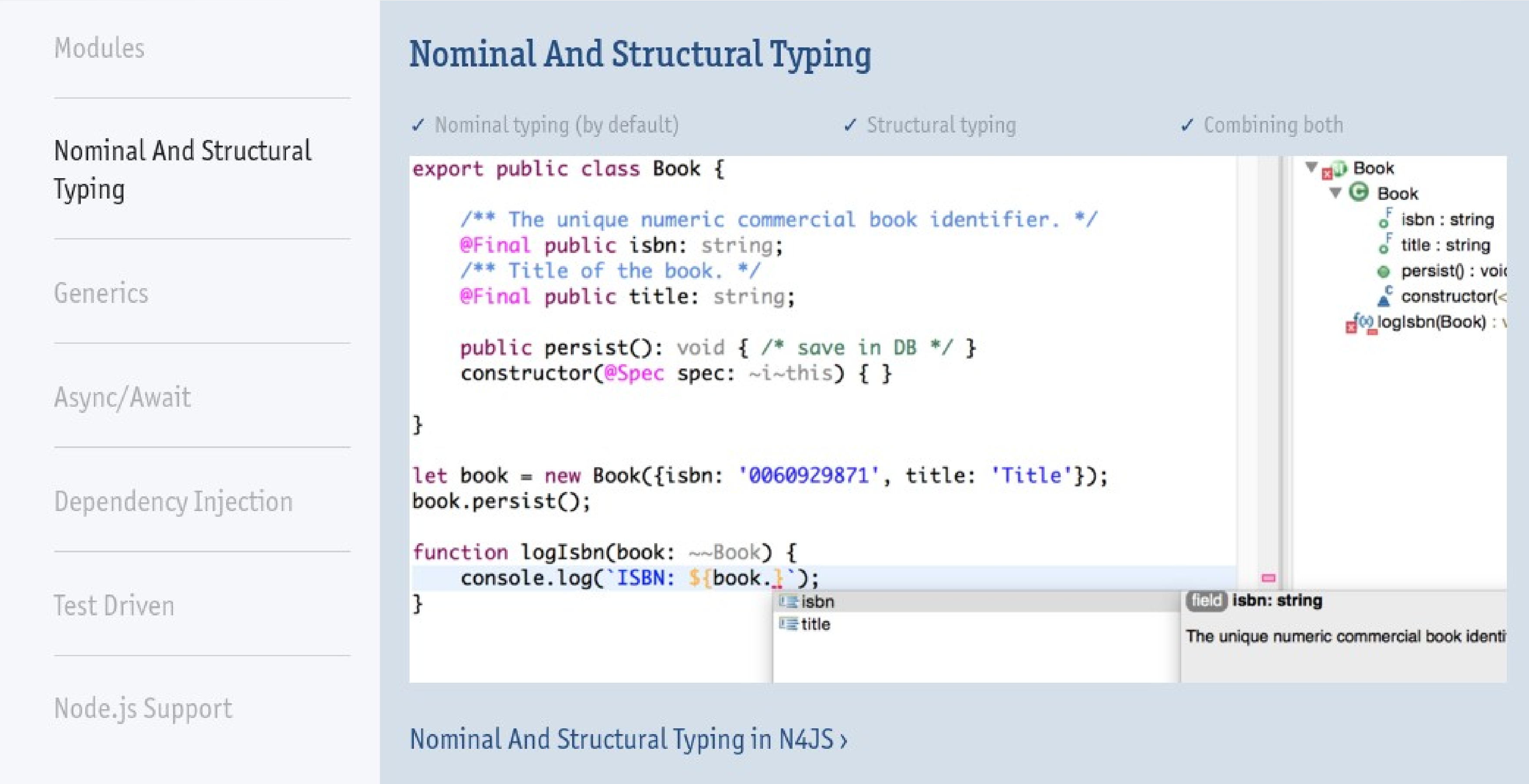
- Types are inferred, statically



```
val personList = newArrayList(
        new Person(”James”, ”Smith”, 50),
        new Person(”John”, ”Smith”, 40),
        new Person(”James”, ”Anderson”, 40),
        new Person(”John”, ”Anderson”, 30),
        new Person(”Paul”, ”Anderson”, 30))
personList.filter[firstname.startsWith(”J”)].
    sortBy[age].take(3).map[surname + ”, ” + firstname].
        join(”; ”)
```

Static type systems are being adopted in dynamic languages

- N4JS: a statically typed Javascript dialect
- http://numberfour.github.io/n4js/



Modules

Nominal And Structural Typing

Generics

Async/Await

Dependency Injection

Test Driven

Node.js Support

Nominal And Structural Typing

✓ Nominal typing (by default) ✓ Structural typing ✓ Combining both

```
export public class Book {

    /** The unique numeric commercial book identifier. */
    @Final public isbn: string;
    /** Title of the book. */
    @Final public title: string;

    public persist(): void { /* save in DB */ }
    constructor(@Spec spec: ~i~this) { }

}

let book = new Book({isbn: '0060929871', title: 'Title'});
book.persist();

function logIsbn(book: ~~Book) {
    console.log(`ISBN: ${book.}`);
}
```

isbn: string

The unique numeric commercial book identi

Nominal And Structural Typing in N4JS ›

About N4JS

- Implemented with Xtext
- The type system implemented in Xsemantics
- Shown on Wednesday in my talk

15:45 - 17:30 COMP TOOLS 1

Implementing the Type System for a
Typed Javascript and its IDE

Lorenzo Bettini, Jens von Pilgrim, Mark-Oliver Reiser

Social Media, Crime, Cyber Security: Some Look Ahead Thoughts

Kendall E. Nygard

North Dakota State University

Kendall E. Nygard, Research Areas

- Encryption, Cyber Security
- Smart Grid,
- Sensor Networks
- Unmanned Air Vehicles, especially mission planning
- Social networks

Famous Data Breaches

- Federal Reserve Bank of New York, 2016 – Bangladesh loses $100M
- Target Stores, 2013, 110M records
- Sony Playstation, 102.6M accounts, cleanup = $171M
- Home Depot, 2014, 56M payment cards
- U. S. Office of Personnel Management, 20M records

U. S. State of North Dakota Takes Action in Cyber Security

- Upwards of 40 million attacks monthly on state government alone
- Huge need for Cyber Security professionals
- Governor Task Force established
- Chancellor of Higher Education System establishes 11 campus consortium in Cyber Security education

Cyber Security for Internet of Things and Cyber Physical Systems

66% of major companies rate them selves as underprepared for security in such systems

Sources: Wordstream, Raytheon

Cyber Crime Major 3-Year Trends (Raytheon)

Category	Percentage Change Forecast
Nation-state attackers	+37%
Cyber Warfare or Cyber Terrorism	+24%
High-value data Breaches	+15%
Sophistication of Attackers	+14%

Cyber Security Need for Adaptation

- "…new and different kinds of threats …some of these kinds of threats will get in … are we really ready …?" Vice Adm. Jan Tighe, commander of U.S. Fleet Cyber Command
- **Software Defined Perimeter (SDP) and BeyondCorp**. Mover away from single perminer firewalls. For example, cloud SCP security alliance Integrates device authentication, identity-based access and dynamically provisioned connectivity to redefine perimeter. Effectively stops most forms of network attacks, such as denial of service, man in the middle, etc.

Digital Natives, born 1982 or later

- Have always been connected, are tech savvy
- One-third of the adult population by 2020 and 75 percent of the workforce by 2025
- Tend to believe that the world should be open and not hierarchical
- Very social, always connected
- Skeptical of motives and will challenge authority
- Believe they already know a great deal

Source: 123RF, Shutterstock

Source: Brookings Institution

Digital Immigrants, born before 1982

- Adapted to technology as adults
- Less knowledgeable and comfortable with technologies
- Tend to obey rules and respect authority
- Tend to value security very highly
- Learning styles tend to be traditional

Source: Brookings Institution

Sources: Georgia Tech, MHealthWatch

Social Media Analytics

- **Reactive analyses**. e.g., law enforcement investigating San Bernadino crime
- **Proactive analyses.** e.g., U. S. Department of State gaining information on the perceptions and attitudes of people based on Twitter feeds.
- **Tools.** Natural language processing across multiple languages, data mining, human network analytics, ontology management, sentiment analyses