power and tools that allow the developer to create and compose abstractions while making use of the full power of concurrency and distribution.

1. Abstractions for data naming, sharing and storage (consistency and versioning), computation (processes), communication (invocation).
2. These abstractions can be instantiated and controlled by a separate control plane, enabling elastic configuration of the number and placement of computation and data entities, transparently to the program text.
3. A composition mechanism based on scoping enables information hiding.
4. Communication supports asynchronous (concurrent) and synchronous (consensus-based) concurrency and distribution.
operation invocation, with transactional and causal consistency guarantees.
5. Communication channels are based on a publish-subscribe/data flow model, with forward and backward paths, and carry any mixture of state, delta, or operation.
6. Transparent storage and communication of metadata, such as timestamps, provenance, security labels, or accounting information.
7. Specifying preconditions, effects and invariants, in order to enable verification.

At the same time, our approach helps avoid many of the opportunities for error, by focusing on the essential properties of application correctness. It is often the invariants required over application data that dictates the protocol for accessing the data; this is an intuition that programmers commonly apply. Hence, we aim to apply leverage language and verification tools, to aid the programmer in choosing the best consistency level and in synthesising a program that respects its specification.

Principales activités
The research has both a fundamental and an applied aspect and aims for practical results.

Compétences

Requirements and application

Candidates to this position should hold a Master's in Computer Science/Informatics or a related field. They must be excited by research in systems, distributed systems, databases, and/or language and verification, and should have an excellent academic record in one of these areas. He or she will be developing and experimenting software at large scale. Teamwork and communication skills, industrial experience, and good knowledge of Erlang and/or node.js is a plus. This is offered as part of Inria’s annual PhD competition. To apply, please provide the following information:

- A resume or Curriculum Vitæ.
- A list of courses and grades of the last two years of study (an informal transcript is OK).
- Names and contact details of at least two references (people who can recommend you), whom we will contact directly.
- If relevant, a link to your publications and/or open-source developments.

Avantages sociaux

- Subsidised catering service
- Partially-reimbursed public transport

Rémunération

Gross Salary per month: 1 982 € the first 2 years and 2 085 € the last year