**System Overview**

**Camelot program**: Describes bounds for space consumption and other kinds of resource.

**Type**: Bytecode logic of resources within the Isabelle theorem prover.

**Grail bytecode** & **Proof of resource bound**: Bytecode logic of resources within the Isabelle theorem prover.

**Compiler**

**Certificate generator**

High level functional programming language:

Low level bytecode executes on the Java Virtual Machine:

**Resource inference**
MRG – Mobile Resource Guarantees

GRAIL

Camelot

Compile

=}

Imperative

Functional

method static int fib (int n) =
  let val a = 0
  val b = 1
  fun loop (int a, int b, int n) =
    let val b = add a b
    val a = sub b a
    val n = sub n 1
    in
      test(n,a,b)
    end
  fun test (int n, int a, int b) =
    if n<=1 then b else loop(a,b,n)
  in
    test(n,a,b)
  end

Java classfile

Guaranteed Resource Allocation Intermediate Language

Project IST-2001-33149 funded by the Global Computing proactive initiative in the Future and Emerging Technologies arm of the IST Programme
MRG – Mobile Resource Guarantees

Proof-Carrying Code

Code producer

Camelot program

Certificate

Grail

Java classfile

Compiler

Code consumer

Resource policy

Proof checker

Certificate

Grail

Java classfile

Proof

Carrying Code

Code producer

Camelot program

Certificate

Grail

Java classfile

Compiler

Code consumer

Resource policy

Proof checker

Certificate

Grail

Java classfile

Proof

Carrying Code

Project IST-2001-33149 funded by the Global Computing proactive initiative in the Future and Emerging Technologies arm of the IST Programme