DE LA RECHERCHE À L'INDUSTRIE

OPEN-SOURCING SOFTWARE CONFIDENCE

OPEN WORLD FORUM

François | Bobot Software Security Laboratory

www.cea.fr

How do you trust (your) open-source systems?

PROCESS- AND PRODUCT-BASED ASSURANCE

What are the means to build trust?

- Process-based assurance
  - Based on testing, V&V tools designed in the 1980s
  - Familiar, but **expensive** to scale up to large, participative process

Inapplicable to COTS software components

- Product-based assurance
  - Using formal methods to provide strong guarantees regarding:
    - Compliance with software safety requirements
    - Absence of software security vulnerabilities
  - Disruptive, but help meet mandatory requirements **at reduced costs**

Next-generation verification tools are reaching maturity in terms of cost effectiveness and industrial integration

Tools to understand software properties

- Properties are formalized using **unequivocal specifications**

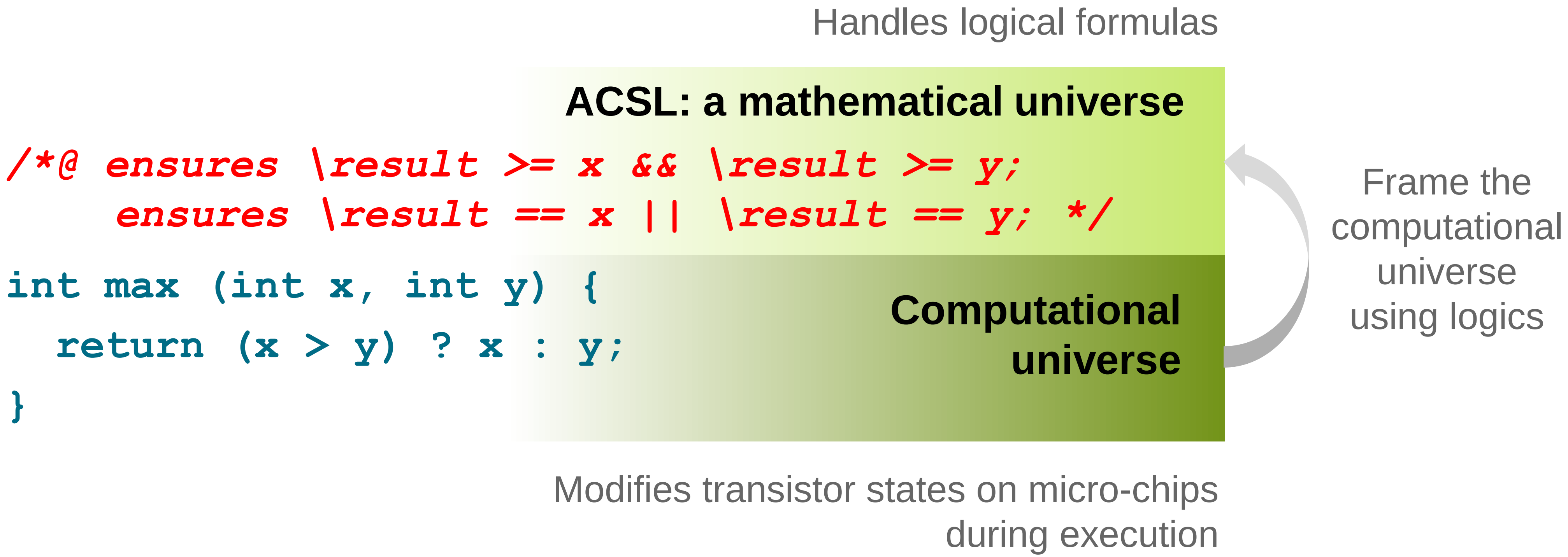
```
\forall a, i ; \valid(a+(0..N-1)) ==> 0 <= i < N ==> a[i] <= C
```

- Software systems are analyzed as **sets of rules**
  - Transforming the system state
  - Satisfying certain properties

On a given perimeter

- Formal methods are used to **prove** that some software properties hold…
- … or to **provide insight** on why other properties do not.

A FORMALISM FOR SOFTWARE VALIDATION



Handles logical formulas

ACSL: a mathematical universe

```
/*@ ensures \result >= x && \result >= y;
     ensures \result == x || \result == y; */
int max (int x, int y) {
   return (x > y) ? x : y;
}
```

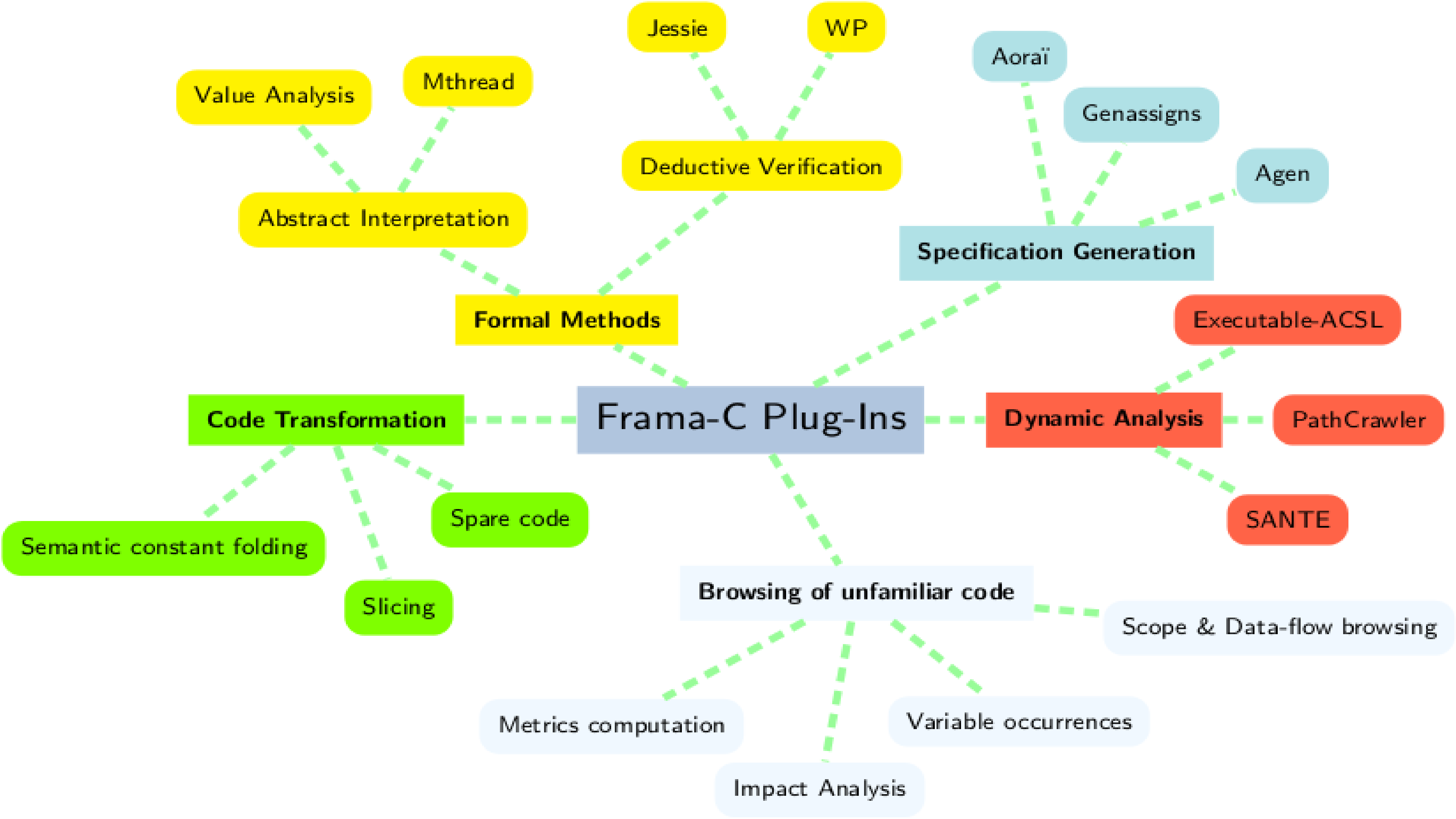
Computational universe

Frame the computational universe using logics

Modifies transistor states on micro-chips during execution

```
unsigned int M ;
/*@
requires \valid (p) && \valid (q);
ensures  M == (*p + *q) / 2;
*/
void mean (  unsigned int* p,
             unsigned int* q ) {
if (* p >= * q )
  M = (* p - * q ) / 2 + * q ;
else
  M = (* q - * p ) / 2 + * p ;
}
```

- **Caller-callee contract**
- **Callee requires some pre-conditions from the caller**
- **Callee ensures some post-conditions hold when it returns**



Frama-C Plug-Ins

Formal Methods

Abstract Interpretation

Value Analysis

Mthread

Deductive Verification

Jessie

WP

Specification Generation

Aoraï

Genassigns

Agen

Dynamic Analysis

Executable-ACSL

PathCrawler

SANTE

Browsing of unfamiliar code

Scope & Data-flow browsing

Metrics computation

Impact Analysis

Variable occurrences

Code Transformation

Semantic constant folding

Slicing

Spare code

ENFORCING CODING STANDARDS WITH FRAMA-C

- Result Airbus and Atos have designed the Taster plugin on top of Frama-C to enforce coding standards.

- Conclusion Frama-C yields productivity gains and ensures code quality.

Benefits :

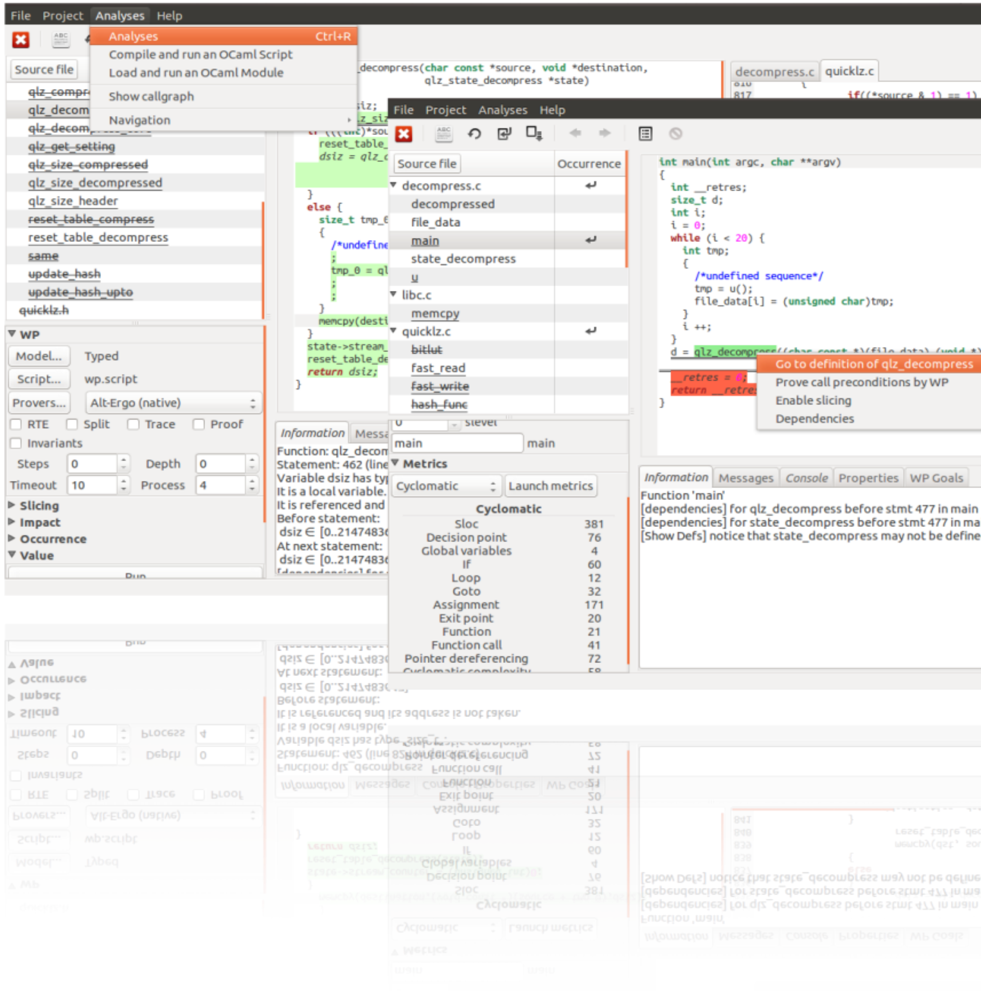
✓ Eases code review on syntactic or typing rules.

✓ Validation of semantic rules:

- dataflow related rules on variables,
- runtime errors requiring a value analysis.

SEMANTICAL ANALYSIS

- Automated process
- Integral & pointer ranges
- Some ACSL verifications
- Runtime-errors threats
- Side-effects & dependency analysis
- Program structure & transformations



Model... Typed

Script... wp.script

Provers... Alt-Ergo (native)

Steps 0 Depth 0

Timeout 10 Process 4

slevel 0

main main

Cyclomatic

```
_decompress(char const *source, void *destination,
            qlz_state_decompress *state)
```

```
817            if((*source & 1) == 1)
```

```
int main(int argc, char **argv)
{
  int __retres;
  size_t d;
  int i;
  i = 0;
  while (i < 20) {
    int tmp;
    {
      /*undefined sequence*/
      tmp = u();
      file_data[i] = (unsigned char)tmp;
    }
    i ++;
  }
  d = qlz_decompress((char const *)(file_data), (void *)
  __retres = 0;
  return __retres
}
```

Function: qlz_decom
Statement: 462 (line
Variable dsiz has typ
It is a local variable.
It is referenced and
Before statement:
dsiz ∈ [0..2147483
At next statement:
dsiz ∈ [0..2147483

Cyclomatic	
Sloc	381
Decision point	76
Global variables	4
If	60
Loop	12
Goto	32
Assignment	171
Exit point	20
Function	21
Function call	41
Pointer dereferencing	72

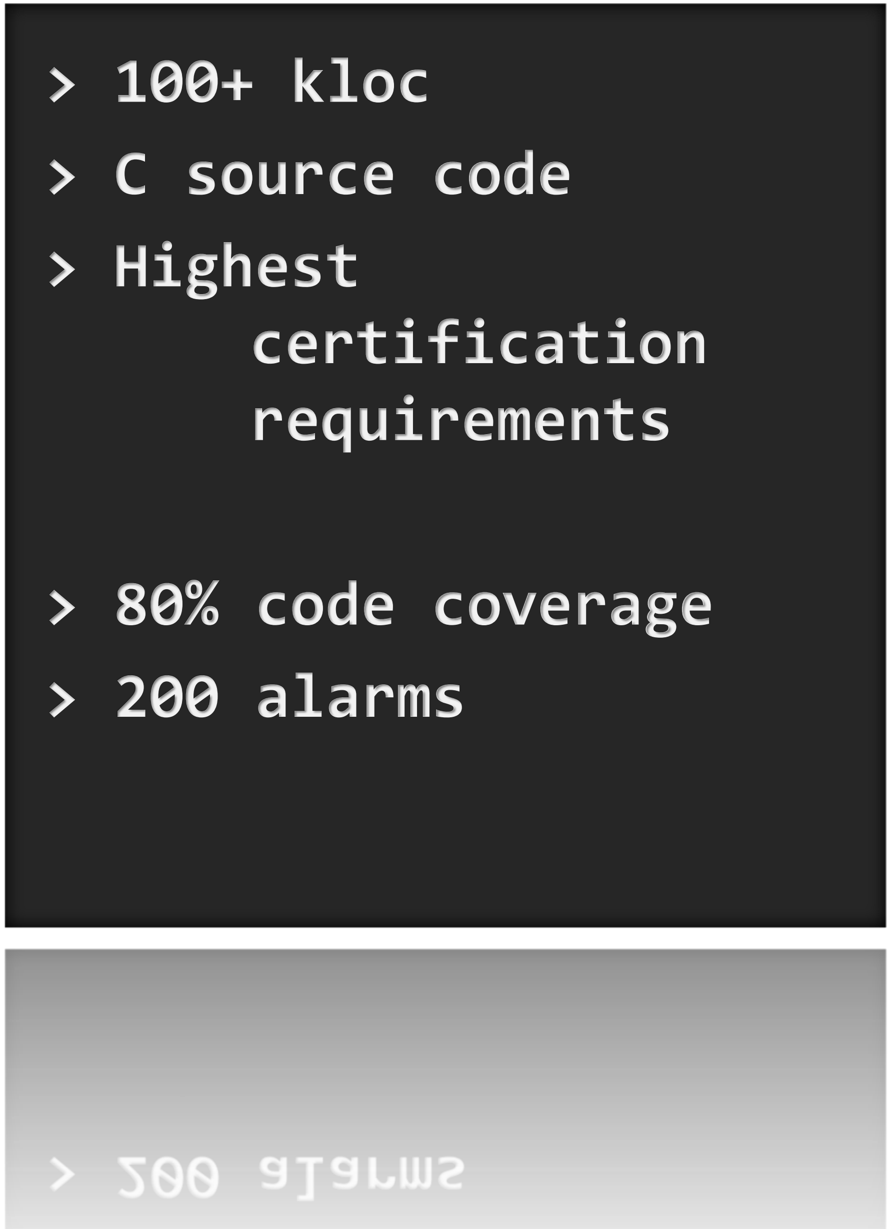
Function 'main'
[dependencies] for qlz_decompress before stmt 477 in main
[dependencies] for state_decompress before stmt 477 in ma
[Show Defs] notice that state_decompress may not be define

CHECKING INTRINSIC FAULTS IN SCADA SYSTEMS

- Result Researchers have demonstrated the absence of multiple fault families in safety-critical software.

- In addition derived analyses cover structural properties on memory separation and cyclic behaviors.

- Conclusion Frama-C enables highly-automated verification runs.



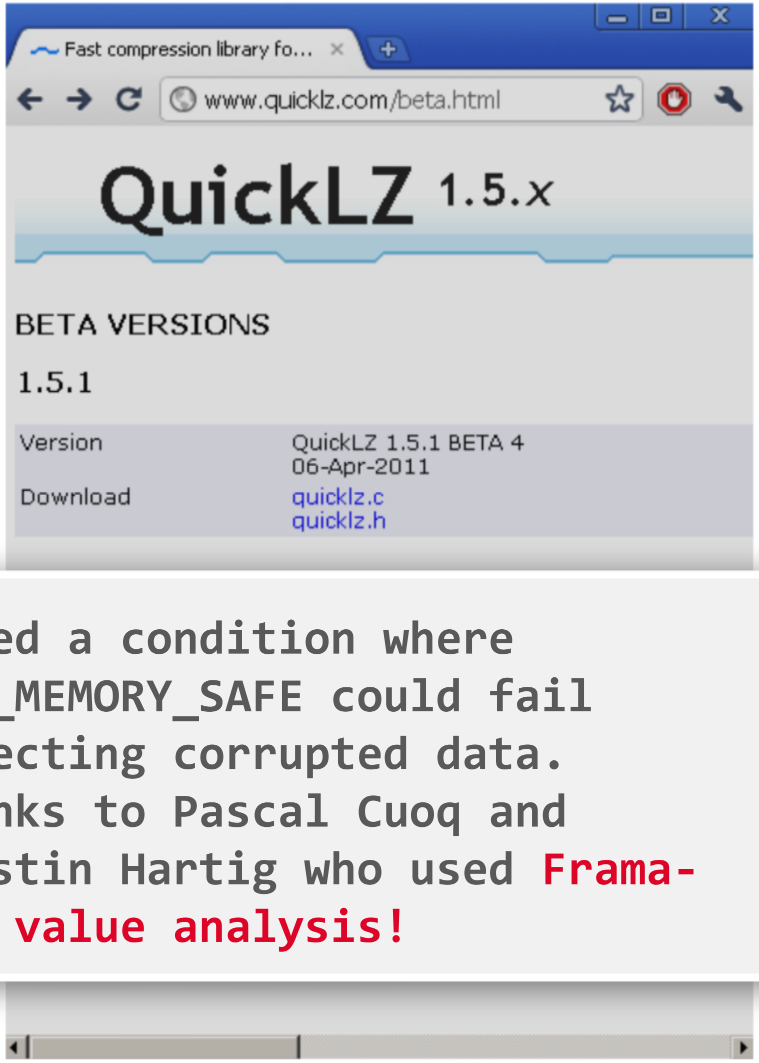
```
> 100+ kloc
> C source code
> Highest
        certification
        requirements

> 80% code coverage
> 200 alarms
```

DETECTION OF A SECURITY FLAW IN A COTS COMPRESSION LIBRARY

- Result **CEA researchers identified a bug in the QuickLZ library. This bug was acknowledged by the designer and corrected in version beta 1.5.1.**

- Conclusion **Software analysis can be applied to general-purpose COTS, enabling their use in security-critical systems.**

EMC² where information lives® facebook AGFA Agfa



QuickLZ 1.5.x

BETA VERSIONS

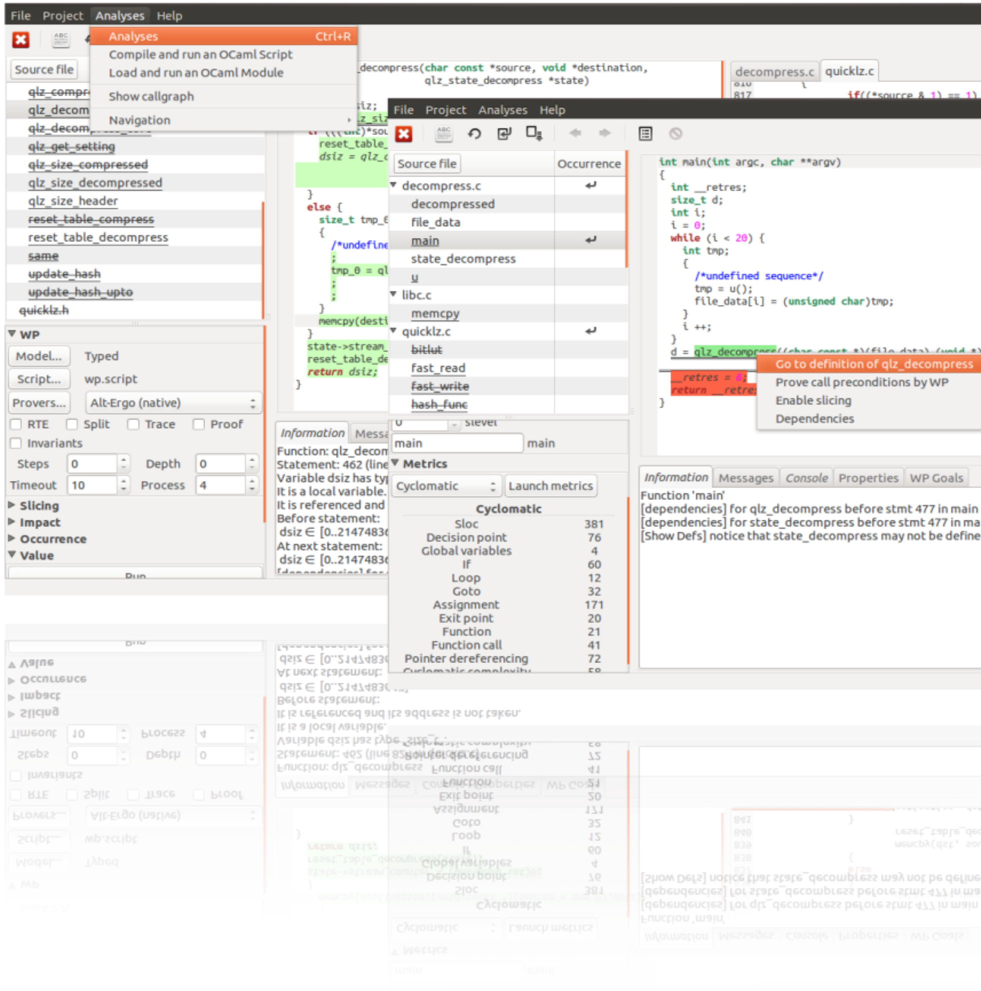
1.5.1

Version	QuickLZ 1.5.1 BETA 4 06-Apr-2011
Download	quicklz.c quicklz.h

```
Fixed a condition where
QLZ_MEMORY_SAFE could fail
detecting corrupted data.
Thanks to Pascal Cuoq and
Kerstin Hartig who used Frama-
C's value analysis!
```

DEDUCTIVE VERIFICATION

- Assisted process
- Full range of ACSL specifications
- Proof of code conformity
- Use of external solvers
- Function call sequences



```
int main(int argc, char **argv)
{
  int __retres;
  size_t d;
  int i;
  i = 0;
  while (i < 20) {
    int tmp;
    {
      /*undefined sequence*/
      tmp = u();
      file_data[i] = (unsigned char)tmp;
    }
    i ++;
  }
  d = qlz_decompress((char const *)(file_data), (void *)
  __retres = 0;
  return __retres;
}
```

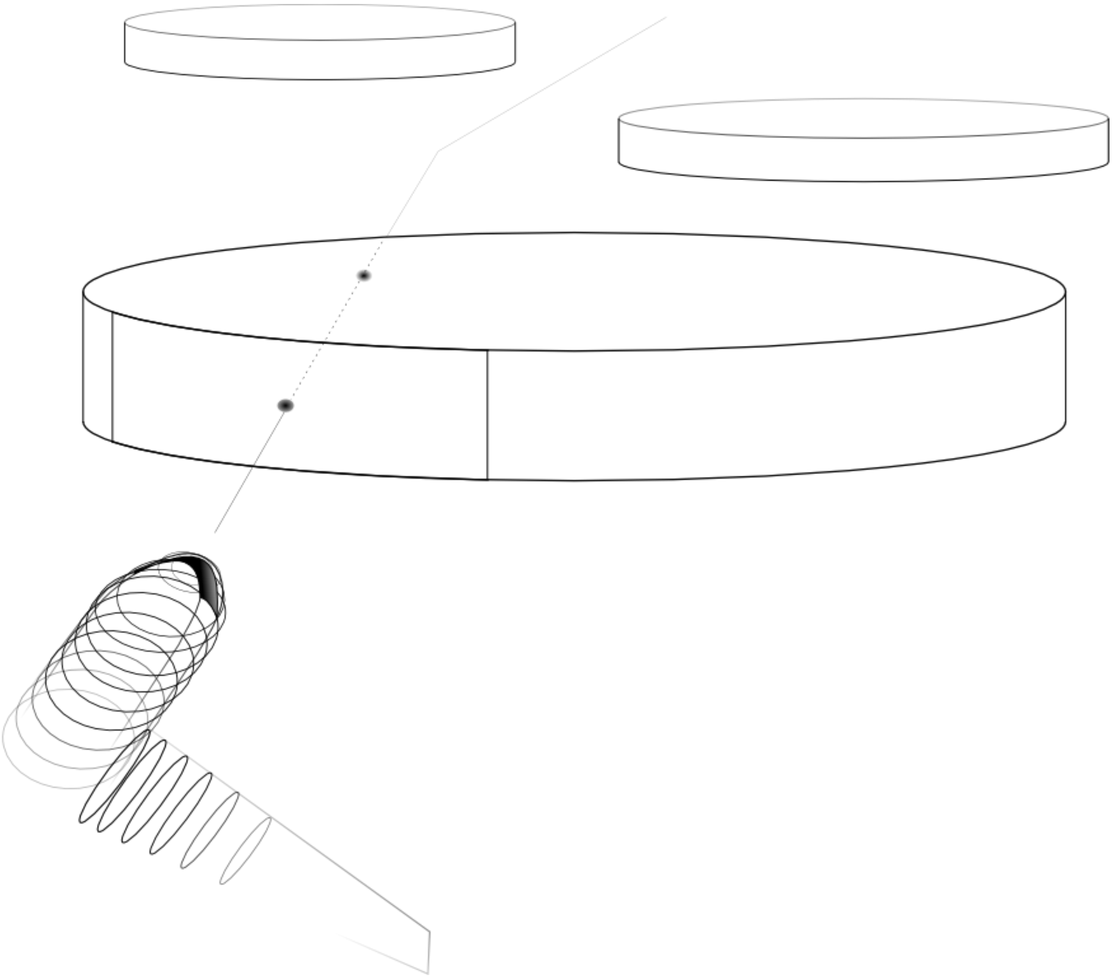
```
Function 'main'
[dependencies] for qlz_decompress before stmt 477 in main
[dependencies] for state_decompress before stmt 477 in ma
[Show Defs] notice that state_decompress may not be define
```

```
Function: qlz_decom
Statement: 462 (line
Variable dsiz has typ
It is a local variable.
It is referenced and
Before statement:
 dsiz ∈ [0..21474836
At next statement:
 dsiz ∈ [0..21474836
```

Cyclomatic	
Sloc	381
Decision point	76
Global variables	4
If	60
Loop	12
Goto	32
Assignment	171
Exit point	20
Function	21
Function call	41
Pointer dereferencing	72

FORMAL ALGORITHMIC CONFORMANCE PROOF

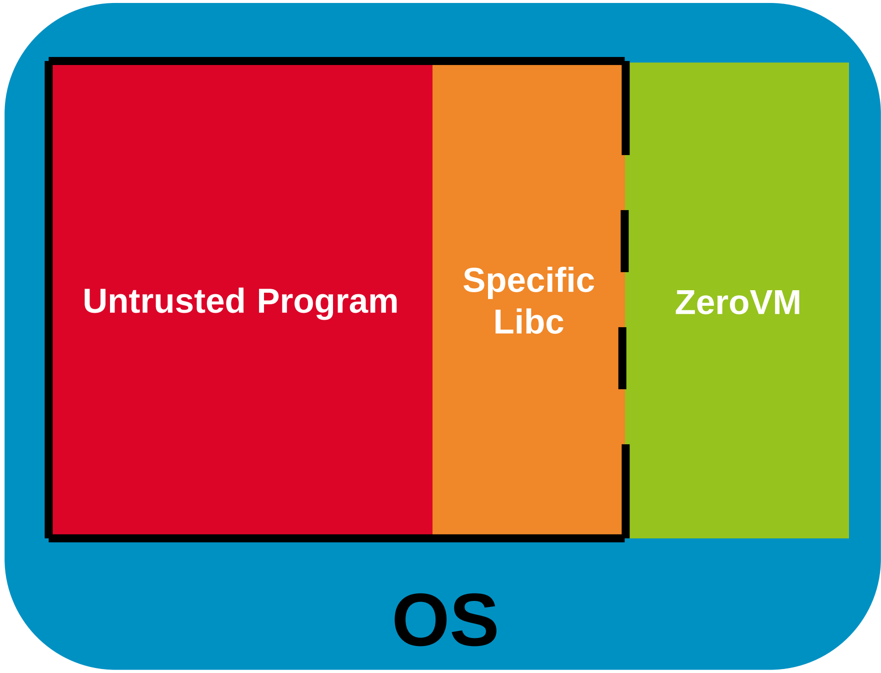
```
% Conflict during interval [B,T]
conflict_2D?(s,v) : bool =
  EXISTS (t: Lookahead): sqv(s+t*v) < sq(D)

% 2-D Conflict Detection (cd2d)
cd2d?(s,v) : bool =
  horizontal_los?(s+B*v) OR omega_vv(s)(v) < 0

% THEOREM: cd2d is correct and complete
cd2d : THEOREM
  conflict_2D?(s,v)
  IFF
  cd2d?(s,v)
```



ZEROVM: HYPERVISOR FOR THE CLOUD

- Isolation done by technique similar to Chrome NaCl:
  - Untrusted program compiled by a custom compiler
  - A validator check the binary before running it

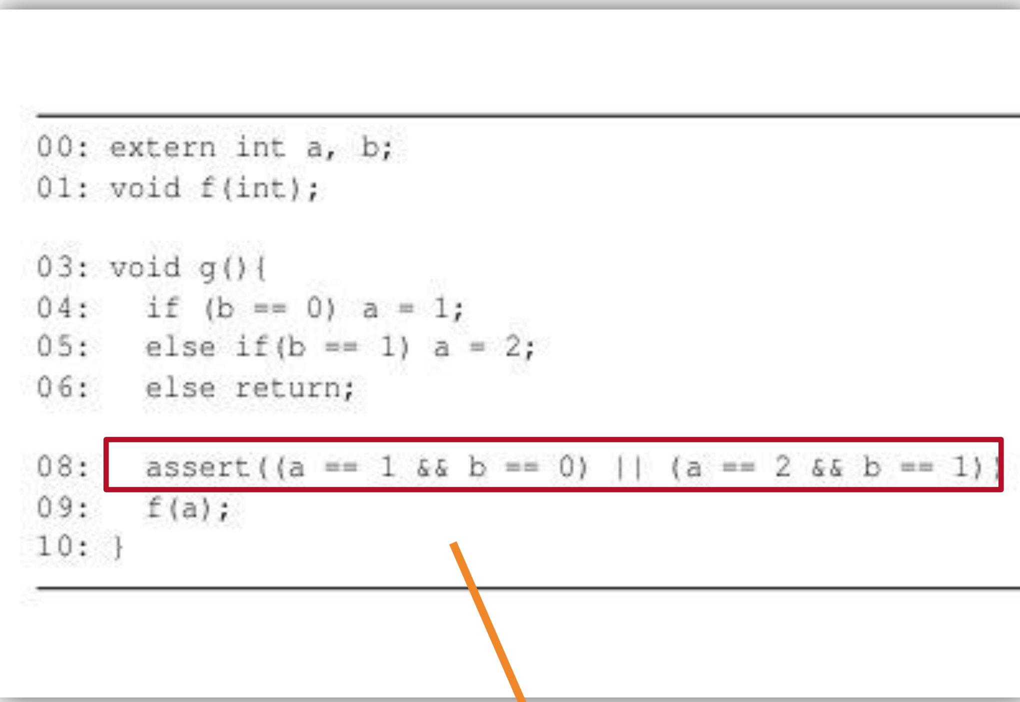
- ZeroVM allows the untrusted application to use OS syscalls only as authorized by the manifest
  - Ex: Restrict which file/pipe can be read/write for how much
  - Verification using the Frama-C WP plugin of this property



RUNTIME MONITORING AND VERIFICATION

- Result Use program analysis and transformations to synthesize:
  - **security monitors**
  - **fault injectors**

- Conclusion Runtime code can be added to harden legacy software through hardware-enabled runtime verification.



```
extern int a, b;
void f(int);

void g(){
   if (b == 0) a = 1;
   else if(b == 1) a = 2;
   else return;

   assert((a == 1 && b == 0) || (a == 2 && b == 1)
   f(a);
}
```

SPECIFIC OPEN-SOURCE PROGRAM VERIFICATION

Great for:

- Give examples of real programs to verify
- If you find a bug, you can propose a patch

Problems:

- The code writer can’t be reached anymore
- Question too hard for the current maintainer
- Can’t ask benevolent developper to right formal specification

FRAMA-C
AN OPEN-SOURCE FRAMEWORK

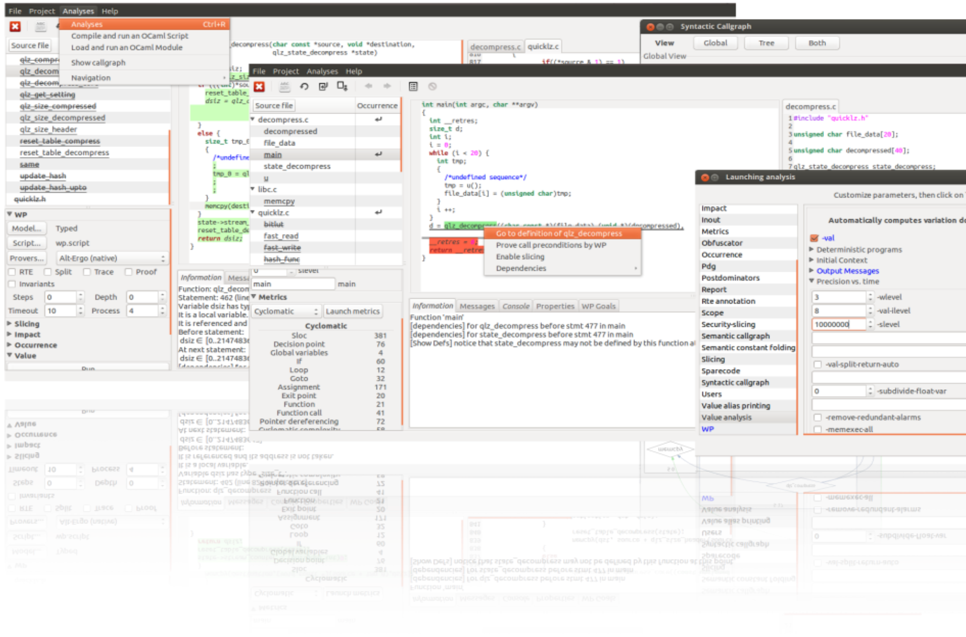
FRAMA-C: LGPL 2.1

- A major release about every six months
- 9000+ downloads
- 3000+ messages on public mailing-list
- BTS, wiki, blogs publics
- Presents on Stackoverflow
- Used for teaching and research in France, England, Germany, Portugal, Russia, USA, …

OPEN-SOURCE AND INNOVATION

- Industries appreciate:
  - the plugin system
  - possibility to look into the kernel
  - other open-source plugins for ideas

- Creation of a start-up in 2013:
  - packaging et dedicated analysis
  - support et industrial licences
  - composants validation kits



```
int main(int argc, char **argv)
{
  int __retres;
  size_t d;
  int i;
  i = 0;
  while (i < 20) {
    int tmp;
    {
      /*undefined sequence*/
      tmp = u();
      file_data[i] = (unsigned char)tmp;
    }
    i ++;
  }
  d = qlz_decompress((char const *)(file_data),(void *)(decompressed),
```

```
Function 'main'
[dependencies] for qlz_decompress before stmt 477 in main
[dependencies] for state_decompress before stmt 477 in main
[Show Defs] notice that state_decompress may not be defined by this function a
```

Cyclomatic	
Sloc	381
Decision point	76
Global variables	4
If	60
Loop	12
Goto	32
Assignment	171
Exit point	20
Function	21
Function call	41
Pointer dereferencing	72

Automatically computes variation d
☑ -val
3 -wlevel
8 -val-ilevel
10000000 -slevel
□ -val-split-return-auto
0 -subdivide-float-var
□ -remove-redundant-alarms
□ -memexec-all

TRUST IN SOFT

Merci! Question?