



Software Analyzers

# Ivette

## An amazing new GUI for Frama-C

---

L. Correnson

CEA-List, Paris Saclay, Software Safety & Security Lab



**list**

```
long m1[8][8], long m2[8][8]) { long i, j, k, tmp1[8][8], tmp2[8][8]; /* Loops indexes and temporary matrices. */ double ftmp1, ftmp2; static int init = 1; static long mc1[8][8], mc2[8][8];
for(i = 0; i < 8; i++) { ftmp1 = ((j == 0) ? 0.5 / sqrt (2.0) : 0.5) * cos ((2.0 * i + 1.0) * j * TH); ftmp2 = ftmp1; /* The well known formula. The max absolute value for ftmp1 and ftmp2 is 0.5. */ ftmp1 *= (1 << NBC2); /* Multiply the cosine coefficient by 2^NBC2. The max absolute value for * ftmp2 is 2^(NBC2-1). */ if (ftmp2 < 0) ftmp2 -= 0.5; else ftmp2 += 0.5; /* For symmetrical rounding. */ mc2[i][j] = ftmp2; } init = 0; }
for(i = 0; i < 8; i++) for(j = 0; j < 8; j++) { for(k = 0, tmp1[i][j] = 0; k < 8; k++) tmp1[i][j] += mc1[i][k] * m1[k][j]; /* The [i,j] coefficient of the matrix product MC1*M1. */ tmp1[i][j] >>= (NBC1 + 10 - NBI); tmp1[i][j] += 1; /* For rounding. */ }
for(i = 0; i < 8; i++) for(j = 0; j < 8; j++) { for(k = 0, tmp1[i][j] = 0; k < 8; k++) tmp1[i][j] += mc2[i][k] * tmp2[k][j]; /* The [i,j] coefficient of the matrix product MC2*M2. */ tmp1[i][j] >>= (NBC1 + 10 - NBI); tmp1[i][j] += 1; /* For rounding. */ }
typedef struct { long pmse[8][8]; long pme[8][8]; } IEEE_1180_1990_stat_set; long IEEE_1180_1990_rand(long L, long H) { static long randx = 1; void IEEE_1180_1990_stat_set(IEEE_1180_1990_stat_set *s);
/* Loops indexes and temporary matrices. */ double ftmp1, ftmp2; static int init = 1; static long mc1[8][8], mc2[8][8]; /* hard-coded cosines matrices. */if (init) { for (i = 0; i < 8; i++) for(j = 0; j < 8; j++) { for(k = 0, tmp1[i][j] = 0; k < 8; k++) tmp1[i][j] += mc1[i][k] * m1[k][j]; /* The [i,j] coefficient of the matrix product MC1*M1. */ tmp1[i][j] >>= (NBC1 + 10 - NBI); tmp1[i][j] += 1; /* For rounding. */ } }
for(i = 0; i < 8; i++) for(j = 0; j < 8; j++) { for(k = 0, tmp1[i][j] = 0; k < 8; k++) tmp1[i][j] += mc2[i][k] * tmp2[k][j]; /* The [i,j] coefficient of the matrix product MC2*M2. */ tmp1[i][j] >>= (NBC1 + 10 - NBI); tmp1[i][j] += 1; /* For rounding. */ }
return s; }
int main() { IEEE_1180_1990_stat_set s; IEEE_1180_1990_stat_set(s); }
/* End of file */
```

The screenshot shows the Frama-C GUI interface. On the left, a 'Source file' list includes files like 'abs', 'encode', 'filter', 'filterz', 'logsch', 'logsc', 'main', 'quantl', 'scalel', 'uppol1', 'uppol2', and 'upzero'. The 'Value' section shows a 'Run' button, a '0' value, and a 'main' callstack entry. The main editor displays C code for 'logsch' and 'scalel' functions. The bottom panel shows a 'Values' tab with a table of variable values across different callstack frames.

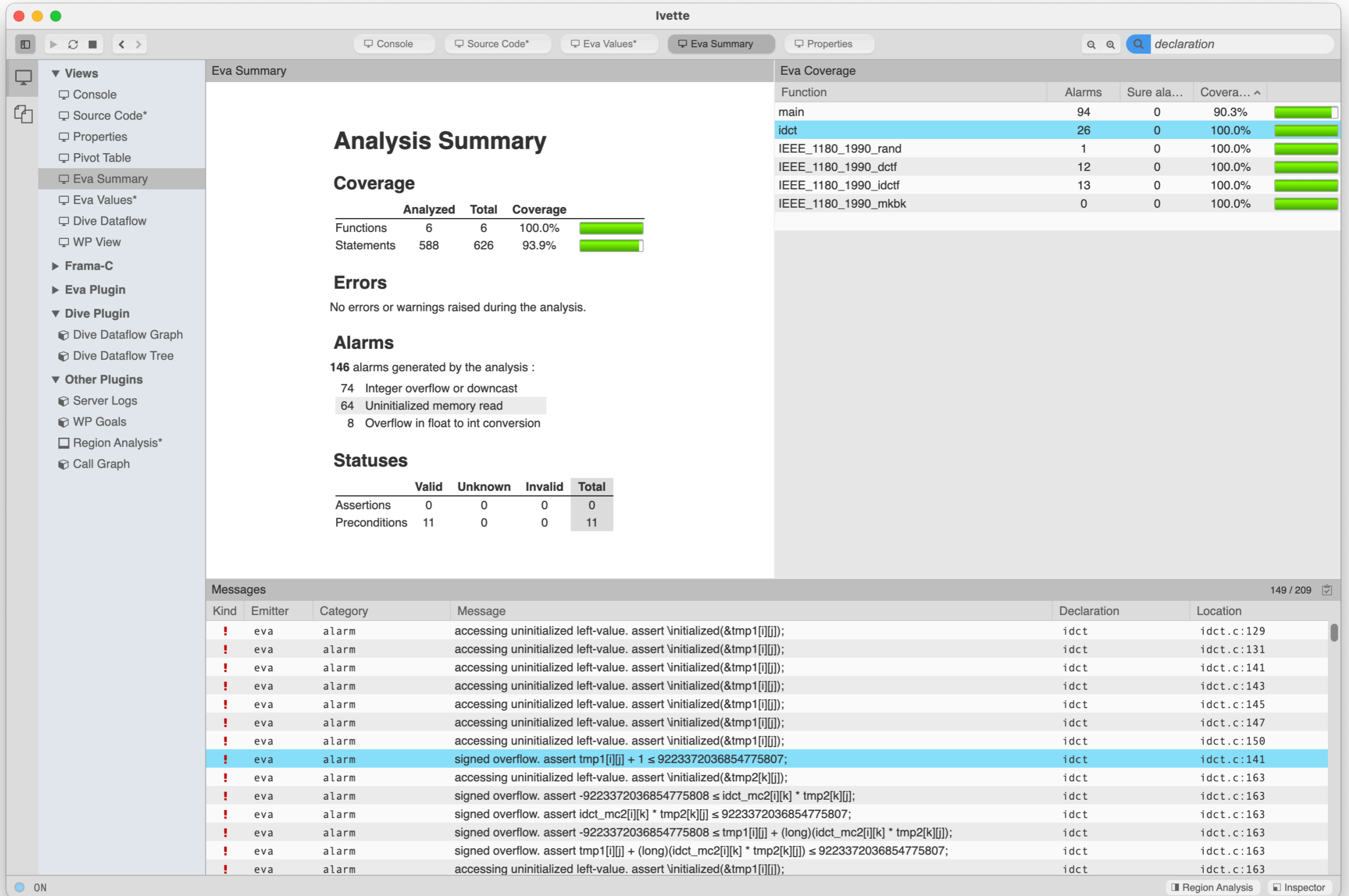
```

int logsch(int ih_0, int nbh_0)
{
  int wd;
  wd = (int)((long)nbh_0 * 127L >> 7L);
  nbh_0 = wd + wh_code_table[ih_0];
  if (nbh_0 < 0) {
    nbh_0 = 0;
  }
  if (nbh_0 > 22528) {
    nbh_0 = 22528;
  }
  return nbh_0;
}
    
```

```

tests/test/adpcm.c
...
488 int scalel(int nbl, int shift_constant)
489 {
490     int wd1, wd2, wd3;
491     wd1 = (nbl >> 6) & 31;
492     wd2 = nbl >> 11;
493     wd3 = ilb_table[wd1] >> (shift_constant + 1 - wd2);
494     return(wd3 << 3);
495 }
496
497 /* upzero - inputs: dlt, dlti[0-5], bli[0-5], outputs: updated bli[0-5] */
498 /* also implements delay of bli and update of dlti from dlt */
499
500 void upzero(int dlt, int *dlti, int *bli)
501 {
502     int i, wd2, wd3;
503     /*if dlt is zero, then no sum into bli */
504     if(dlt == 0) { /* CONDITION 711 */
505     /*@ loop pragma UNROLL 7; */
506         for(i = 0; i < 6; i++) {
507             bli[i] = (int)((255L*bli[i]) >> 8L); /* leak factor of 255/256 */
508         }
509     }
510     else {
511     /*@ loop pragma UNROLL 7; */
512         for(i = 0; i < 6; i++) {
513             if((long)dlt*dlti[i] >= 0) wd2 = 128; else wd2 = -128; /* CONDITION 718 : 2exp6 p...
514             wd3 = (int)((255L*bli[i]) >> 8L); /* leak factor of 255/256 */
515             bli[i] = wd2 + wd3;
516         }
517     }
518     /* implement delay line for dlt */
519     dlti[5] = dlti[4];
    
```

Callstack	wd	ih_0	ih_0 ≤ 2
all	{0; 791}	{1; 2; 3}	unknown
encode ← main	{791}	{3}	invalid
encode ← main	{0}	{2}	valid
encode ← main	{0}	{1}	valid
encode ← main	{0}	{3}	invalid
encode ← main	{0}	{3}	invalid



The screenshot displays the Ivette software interface with the following components:

- Views Panel (Left):** A sidebar containing navigation options: Console, Source Code\*, Properties, Pivot Table, Eva Summary (selected), Eva Values\*, Dive Dataflow, WP View, Frama-C, Eva Plugin, Dive Plugin (with sub-options: Dive Dataflow Graph, Dive Dataflow Tree), and Other Plugins (with sub-options: Server Logs, WP Goals, Region Analysis\*, Call Graph).
- Navigation Bar (Top):** Includes tabs for Console, Source Code\*, Eva Values\*, Eva Summary (active), and Properties. A search bar on the right contains the text 'declaration'.
- Analysis Summary (Center):**
  - Analysis Summary:** A large heading.
  - Coverage:** A table showing analyzed and total counts for functions and statements.
 

	Analyzed	Total	Coverage
Functions	6	6	100.0%
Statements	588	626	93.9%
  - Errors:** A section stating 'No errors or warnings raised during the analysis.'
  - Alarms:** A section titled '146 alarms generated by the analysis:' with a list:
    - 74 Integer overflow or downcast
    - 64 Uninitialized memory read
    - 8 Overflow in float to int conversion
  - Statuses:** A table showing the count of valid, unknown, and invalid statuses.
 

	Valid	Unknown	Invalid	Total
Assertions	0	0	0	0
Preconditions	11	0	0	11
- Eva Coverage (Right):** A table listing functions and their coverage metrics.
 

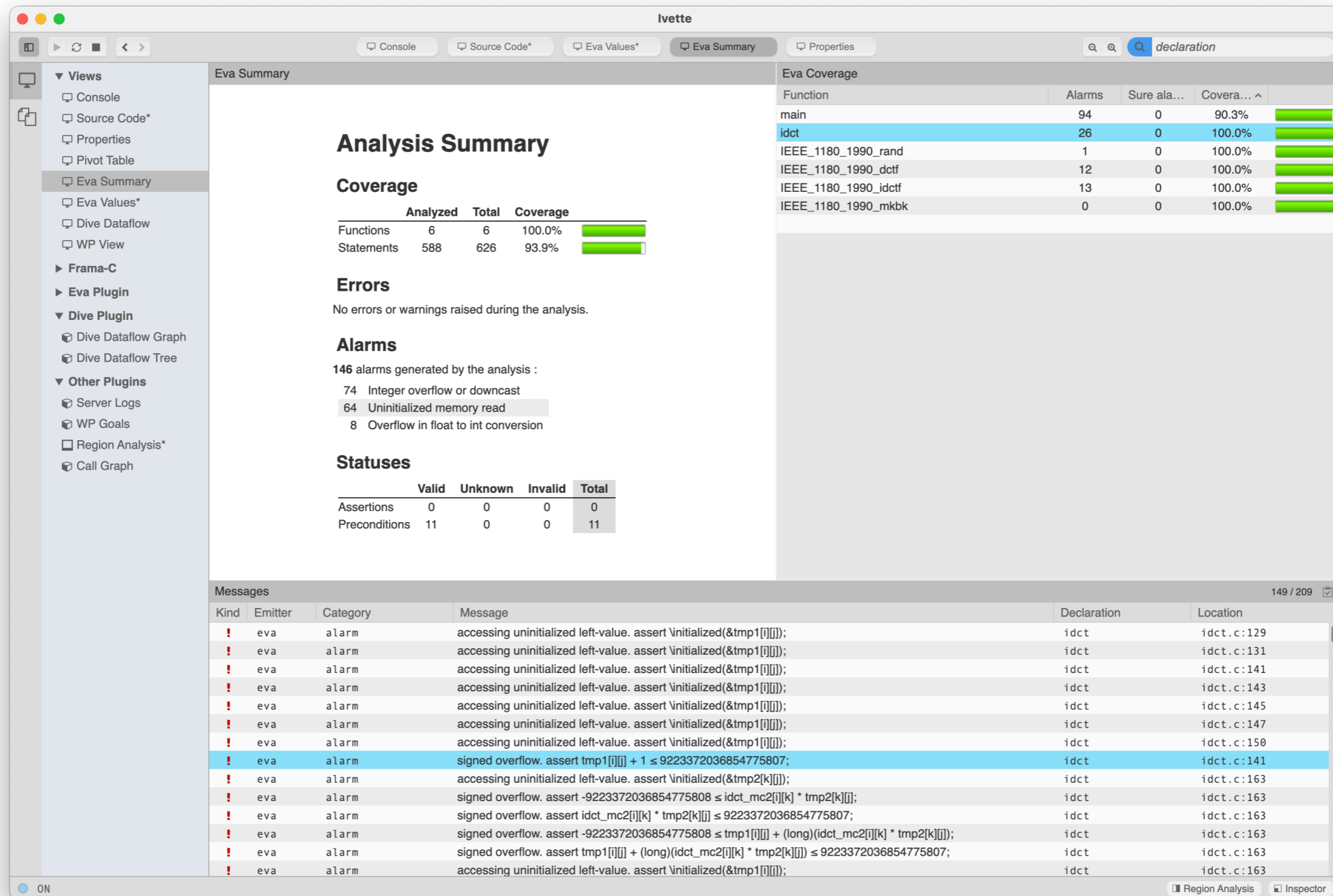
Function	Alarms	Sure ala...	Covera... ^
main	94	0	90.3%
idct	26	0	100.0%
IEEE_1180_1990_rand	1	0	100.0%
IEEE_1180_1990_dctf	12	0	100.0%
IEEE_1180_1990_idctf	13	0	100.0%
IEEE_1180_1990_mkbk	0	0	100.0%
- Messages (Bottom):** A table listing 149 messages. The selected message is:
 

Kind	Emitter	Category	Message	Declaration	Location
!	eva	alarm	signed overflow. assert tmp1[i][j] + 1 ≤ 9223372036854775807;	idct	idct.c:141

# What's New ?

---

Organization



The screenshot shows the 'Eva Summary' window for a project named 'Ivette'. The interface includes a sidebar with various views, a main content area with an 'Analysis Summary', and a 'Messages' table at the bottom.

### Analysis Summary

#### Coverage

	Analyzed	Total	Coverage
Functions	6	6	100.0%
Statements	588	626	93.9%

#### Errors

No errors or warnings raised during the analysis.

#### Alarms

146 alarms generated by the analysis :

- 74 Integer overflow or downcast
- 64 Uninitialized memory read
- 8 Overflow in float to int conversion

#### Statuses

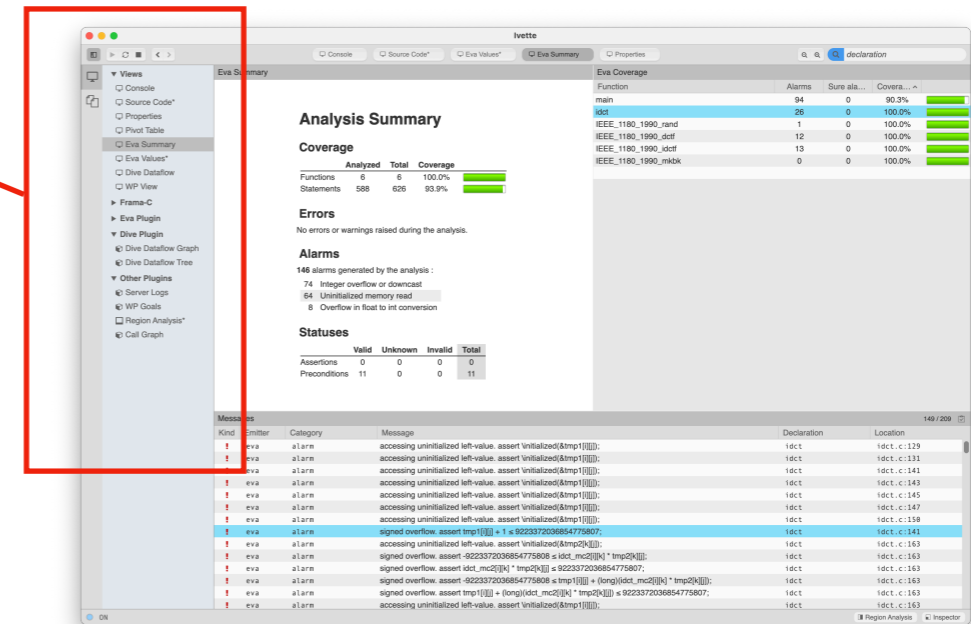
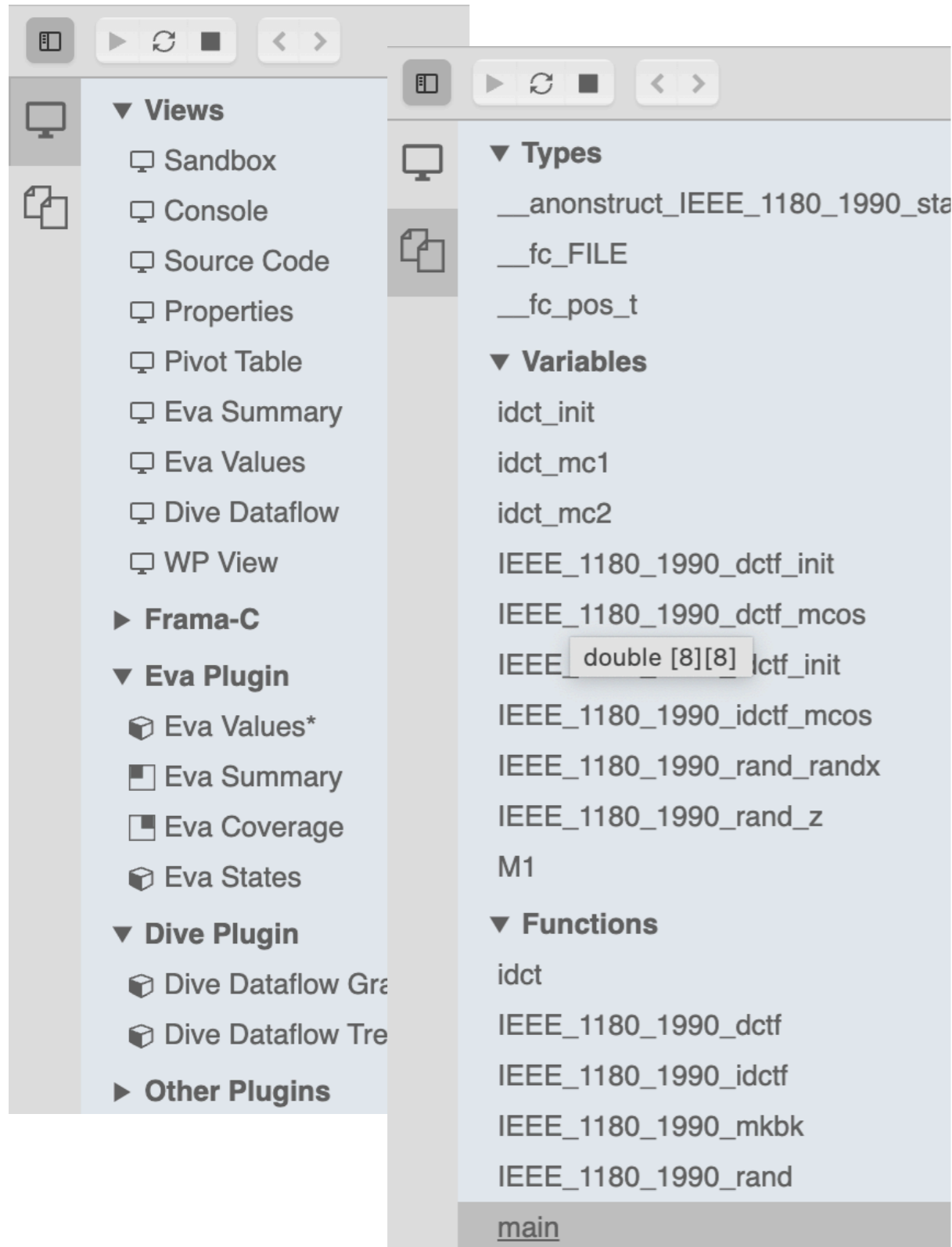
	Valid	Unknown	Invalid	Total
Assertions	0	0	0	0
Preconditions	11	0	0	11

#### Eva Coverage

Function	Alarms	Sure ala...	Covera... ^
main	94	0	90.3%
idct	26	0	100.0%
IEEE_1180_1990_rand	1	0	100.0%
IEEE_1180_1990_dctf	12	0	100.0%
IEEE_1180_1990_idctf	13	0	100.0%
IEEE_1180_1990_mkbk	0	0	100.0%

#### Messages

Kind	Emitter	Category	Message	Declaration	Location
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:129
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:131
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:141
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:143
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:145
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:147
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:150
!	eva	alarm	signed overflow. assert tmp1[i] + 1 ≤ 9223372036854775807;	idct	idct.c:141
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp2[k]);	idct	idct.c:163
!	eva	alarm	signed overflow. assert -9223372036854775808 ≤ idct_mc2[i][k] * tmp2[k];	idct	idct.c:163
!	eva	alarm	signed overflow. assert idct_mc2[i][k] * tmp2[k] ≤ 9223372036854775807;	idct	idct.c:163
!	eva	alarm	signed overflow. assert -9223372036854775808 ≤ tmp1[i] + (long)(idct_mc2[i][k] * tmp2[k]);	idct	idct.c:163
!	eva	alarm	signed overflow. assert tmp1[i] + (long)(idct_mc2[i][k] * tmp2[k]) ≤ 9223372036854775807;	idct	idct.c:163
!	eva	alarm	accessing uninitialized left-value. assert \initialized(&tmp1[i]);	idct	idct.c:163



Q-Splitters

Reset Clear H=0.26 V=0.46 A B C D

**Analysis Summary**

**Coverage**

Functions	Analyzed	Total	Coverage
6	6	6	100.0%
Statements	588	626	93.9%

**Errors**

No errors or warnings raised during the analysis.

**Alarms**

146 alarms generated by the analysis:

- 74 Integer overflow or downcast
- 64 Uninitialized memory read
- 8 Overflow in float to int conversion

**Statuses**

	Valid	Unknown	Invalid	Total
Assertions	0	0	0	0
Preconditions	11	0	0	11

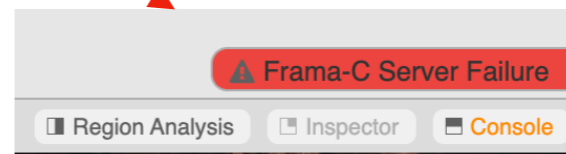
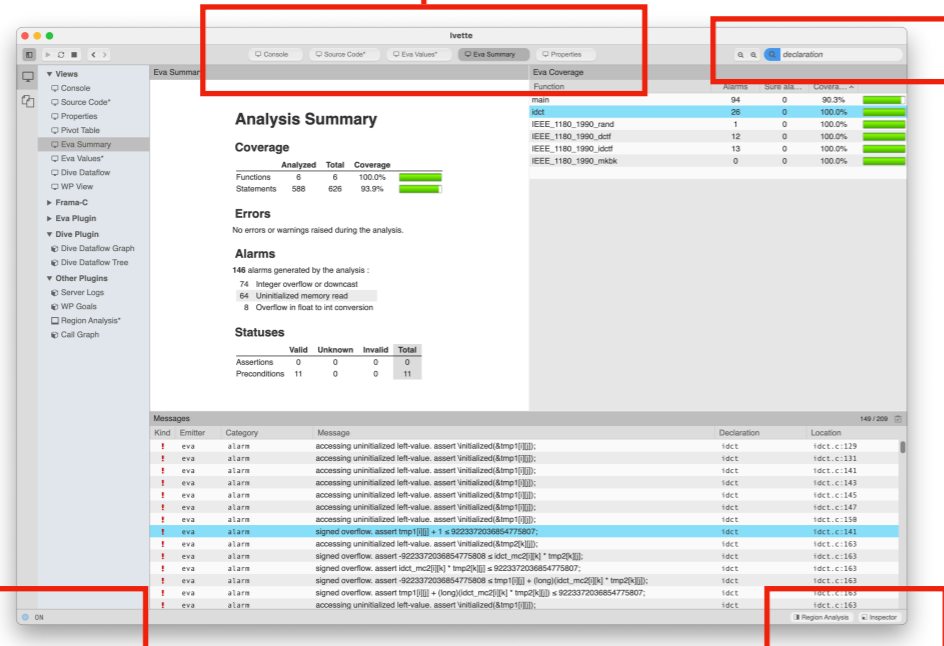
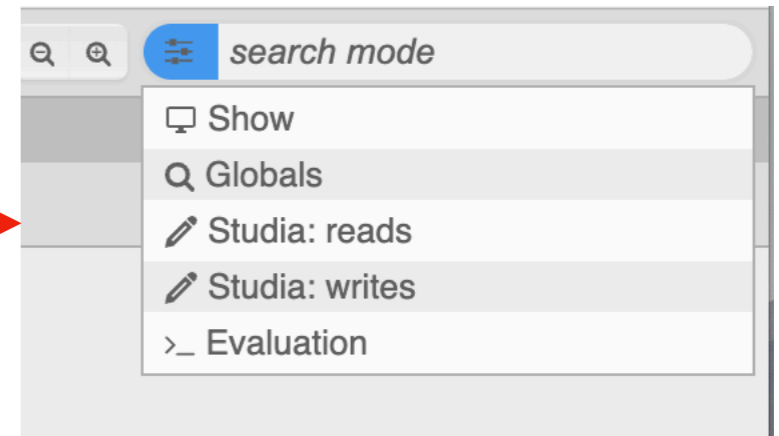
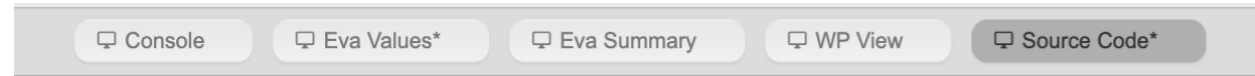
**Eva Coverage**

Function	Alarms	Sure al...	Covera...
main	94	0	90.3%
IEEE_1180_1990_rand	26	0	100.0%
IEEE_1180_1990_dctff	12	0	100.0%
IEEE_1180_1990_idctf	13	0	100.0%
IEEE_1180_1990_mdbk	0	0	100.0%

**Messages**

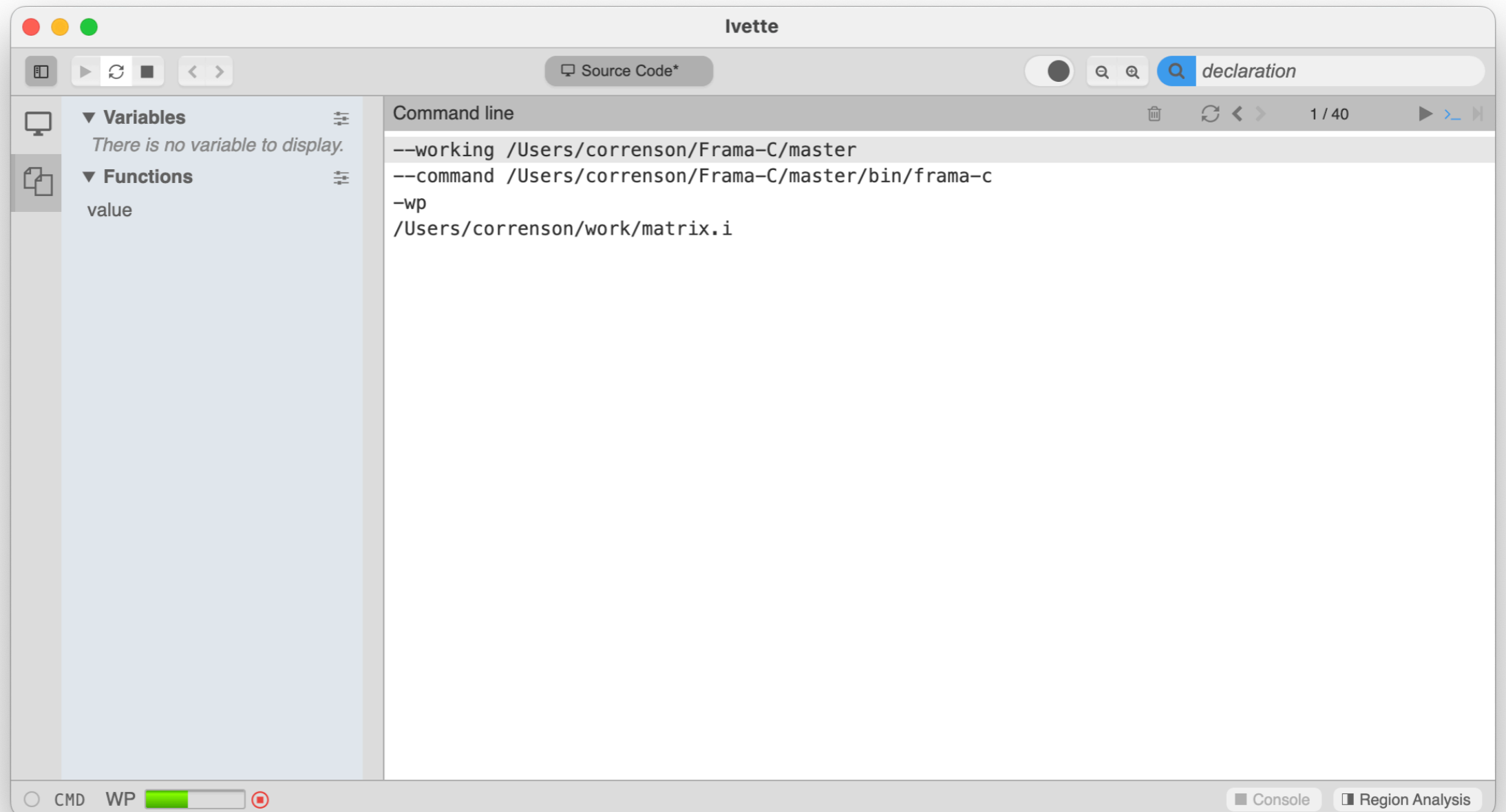
Kind	Emitter	Category	Message	Declaration	Location
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:129
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:131
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:141
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:143
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:145
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:147
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:159
I	eva	alarm	signed overflow, assert tmp1[0] + 1 < 9223372036854775807;	ldct	ldct.c:141
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp2[0]);	ldct	ldct.c:163
I	eva	alarm	signed overflow, assert -9223372036854775808 < idct_mc2[0] * tmp2[0];	ldct	ldct.c:163
I	eva	alarm	signed overflow, assert idct_mc2[0] * tmp2[0] < 9223372036854775807;	ldct	ldct.c:163
I	eva	alarm	signed overflow, assert -9223372036854775808 < tmp1[0] + (long)idct_mc2[0] * tmp2[0];	ldct	ldct.c:163
I	eva	alarm	signed overflow, assert tmp1[0] + (long)idct_mc2[0] * tmp2[0] < 9223372036854775807;	ldct	ldct.c:163
I	eva	alarm	accessing uninitialized left-value, assert uninitialized(&tmp1[0]);	ldct	ldct.c:163

# View Bar – Search Bar – Status Bar – Dock





```
[ ~/work ]  
$ ivette -wp matrix.i
```



## Ivette Documentation - v29.0.0

- M** Ivette Documentation - v29.0.0
- > **M** dome
- > frama-c
- ▼ **M** **ivette**

- I** ComponentProps
- I** ContentProps
- I** Hint
- I** ItemProps
- I** SearchProps
- I** SidebarProps
- I** TitleBarProps
- I** ToolProps
- I** ViewLayoutProps
- T** Layout
- T** Layout1
- T** Layout2
- T** Layout3
- T** Layout4
- T** LayoutPosition
- T** compld
- F** TitleBar
- F** focusSearchMode
- F** registerComponent
- F** registerGroup
- F** registerSandbox
- F** registerSearchMode
- F** registerSidebar

Ivette Documentation / ivette /

# Module ivette

## INDEX

### Interfaces

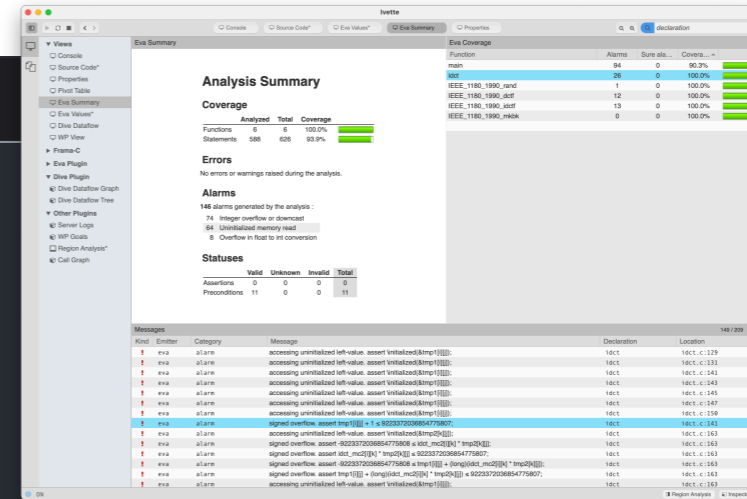
- I** ComponentProps
- I** ContentProps
- I** SearchProps
- I** TitleBarProps
- I** ToolProps
- I** Hint
- I** SidebarProps
- I** ViewLayoutProps

### Type Aliases

- T** Layout
- T** Layout1
- T** Layout3
- T** Layout4
- T** compld
- T** Layout2
- T** LayoutPosition

### Functions

- F** TitleBar
- F** focusSearchMode
- F** registerGroup
- F** registerSandbox
- F** registerSidebar
- F** registerStatusbar
- F** registerView
- F** removeSearchMode
- F** updateSearchMode
- F** useSearchMode
- F** registerComponent
- F** registerSearchMode
- F** registerToolbar
- F** selectSearchMode

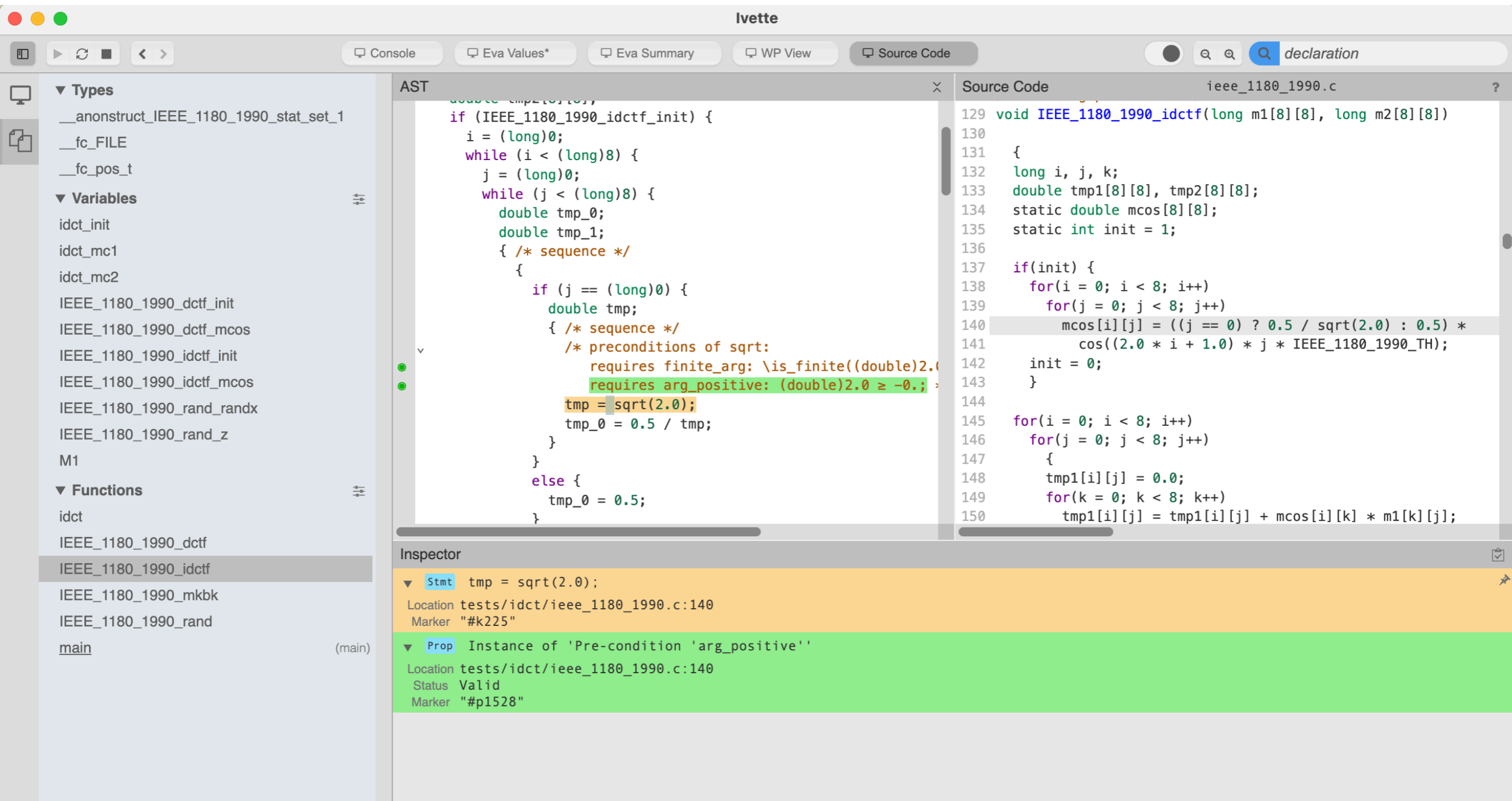


## What's In ?

---

« Old » components... but smarter !

And new ones !



The screenshot shows the Ivette IDE interface with the following components:

- Top Bar:** Includes navigation icons and tabs for Console, Eva Values\*, Eva Summary, WP View, and Source Code. A search bar on the right contains the text "declaration".
- Left Panel:** A tree view showing project structure under "Types" and "Variables". The "Variables" section is expanded, listing various identifiers like `idct_init`, `IEEE_1180_1990_dctf_init`, and `IEEE_1180_1990_idctf_mcos`. The "Functions" section is also visible, listing `idct` and `IEEE_1180_1990_dctf`.
- AST View:** Displays the abstract syntax tree for the selected code. It shows nested `while` loops and an `if` statement. A specific assignment `tmp = sqrt(2.0);` is highlighted in orange, and a precondition `requires arg_positive: (double)2.0 ≥ -0.;` is highlighted in green.
- Source Code View:** Shows the original C code from `ieee_1180_1990.c`. Lines 140-141 are highlighted in grey, corresponding to the AST view. Line 140 contains the assignment `tmp = sqrt(2.0);` and line 141 contains the `requires` precondition.
- Inspector:** Located at the bottom, it provides details for the selected AST node. It shows:
  - Stmt:** `tmp = sqrt(2.0);` at location `tests/idct/ieee_1180_1990.c:140` with marker `"#k225"`.
  - Prop:** Instance of 'Pre-condition 'arg\_positive'' at location `tests/idct/ieee_1180_1990.c:140`, with status `Valid` and marker `"#p1528"`.

ivette

Console Source Code\* Eva Values\* Eva Summary Properties declaration

Views  
 Console  
 Source Code\*  
 Properties  
 Pivot Table  
 Eva Summary  
 Eva Values\*  
 Dive Dataflow  
 WP View  
 Frama-C  
 Eva Plugin  
 Dive Plugin  
 Dive Dataflow Graph  
 Dive Dataflow Tree

### Eva Summary

#### Analysis Summary

##### Coverage

	Analyzed	Total	Coverage
Functions	6	6	100.0%
Statements	588	626	93.9%

##### Errors

No errors or warnings raised during the analysis.

##### Alarms

146 alarms generated by the analysis :

#### Eva Coverage

Function	Alarms	Sure ala...	Covera...
main	94	0	90.3%
idct	26	0	100.0%
IEEE_1180_1990_rand	1	0	100.0%
IEEE_1180_1990_dctf	12	0	100.0%
IEEE_1180_1990_idctf	13	0	100.0%
IEEE_1180_1990_mkbk	0	0	100.0%

### AST

```

j ++;
}
if (omse > (long)12800) {
  succ = (long)0;
}
if (ome < (long)0) {
  /*@ assert Eva: signed_overflow: -ome ≤ 9223372036854775807; */
  tmp_6 = - ome;
}
else {
  tmp_6 = ome;
}
if (tmp_6 > (long)960) {
  succ = (long)0;
}
}
i ++;

```

### Dive Dataflow Graph

main

res[i].pme[j][k] → ome

(long)0 → ome

ome → tmp\_6

Node shape: memory  
 constant  
 scalar type  
 aggregate type  
 set of addresses  
 analysis alarm  
 Node color: value cardinality  
 unique value  
 small range of values  
 large range of values  
 extreme range of values  
 Node outline color: taint analysis  
 directly tainted  
 indirectly tainted

### Eva Values

main 1 callstack

#	Before	After	Before	After
1	9223372036854775808..-1]	[-9223372036854775807..-1]	{0} or UNINITIALIZED	

The screenshot shows the Ivette IDE interface. The top bar includes navigation buttons and tabs for Console, Eva Values\*, Eva Summary, and WP View. A search bar on the right contains the text "declaration".

The left sidebar contains a "Views" panel with options like Sandbox, Console, Source Code, Properties, Pivot Table, Eva Summary, Eva Values\*, Dive Dataflow, and WP View. Below it are sections for "Frama-C", "Eva Plugin", "Dive Plugin", and "Other Plugins".

The main workspace is divided into three panels:

- AST:** Displays the abstract syntax tree for a C function:
 

```

      /*@ terminates \true;
      exits \false;
      ensures \false;
      assigns \nothing; */
      void get(int i, int j, int k)
      {
        A: ;
        if (i) {
          i ++;
        }
      }
      
```
- Inspector:** Shows details for a selected "Prop Post-condition":
 

```

      Prop Post-condition
      Location /Users/correnson/work/matrix.c:5
      Status Unknown
      Marker "#p1"
      
```
- WP - TIP:** Displays the proof transformer output for the function:
 

```

      typed_get_ensures Timeout (Qed 56ms) (Alt-Ergo) (Cached)
      Script (?)
      Have: (i@L15) = i@C:.
      Stmt { L15: i = i_1; }
      }
      Else { Have: i@B: = i@C:. Stmt { i = i_1; j = j_0; } }
      Stmt { C:: }
      If k@L1 != 0
      Then {
        Stmt { i = 1 + i@C;; }
        Have: (i@L18) = i@L5.
        Stmt { L18: i = i_0; }
      }
      Else { Have: i@C: = i@L5. Stmt { i = i_0; k = k_0; } }
      Stmt { L5: }
      Probe A = 0.
      Probe B = i@B: - i@L1.
      Probe C = i@C: - i@L1.
      Probe Offset = ((4 * k@L1) + (20 * j@L1) + (80 * i@L5)) / 4.
      }
      Prove: false.
      
```

At the bottom, there are controls for "Range (0-20)", "Enumerate lower, range 0-20 and upper.", and status indicators for "Configured", "Inf 0", and "Sup 20".

The screenshot displays the Ivette IDE interface with the following components:

- AST View:** Shows the source code of the `job` function:
 

```

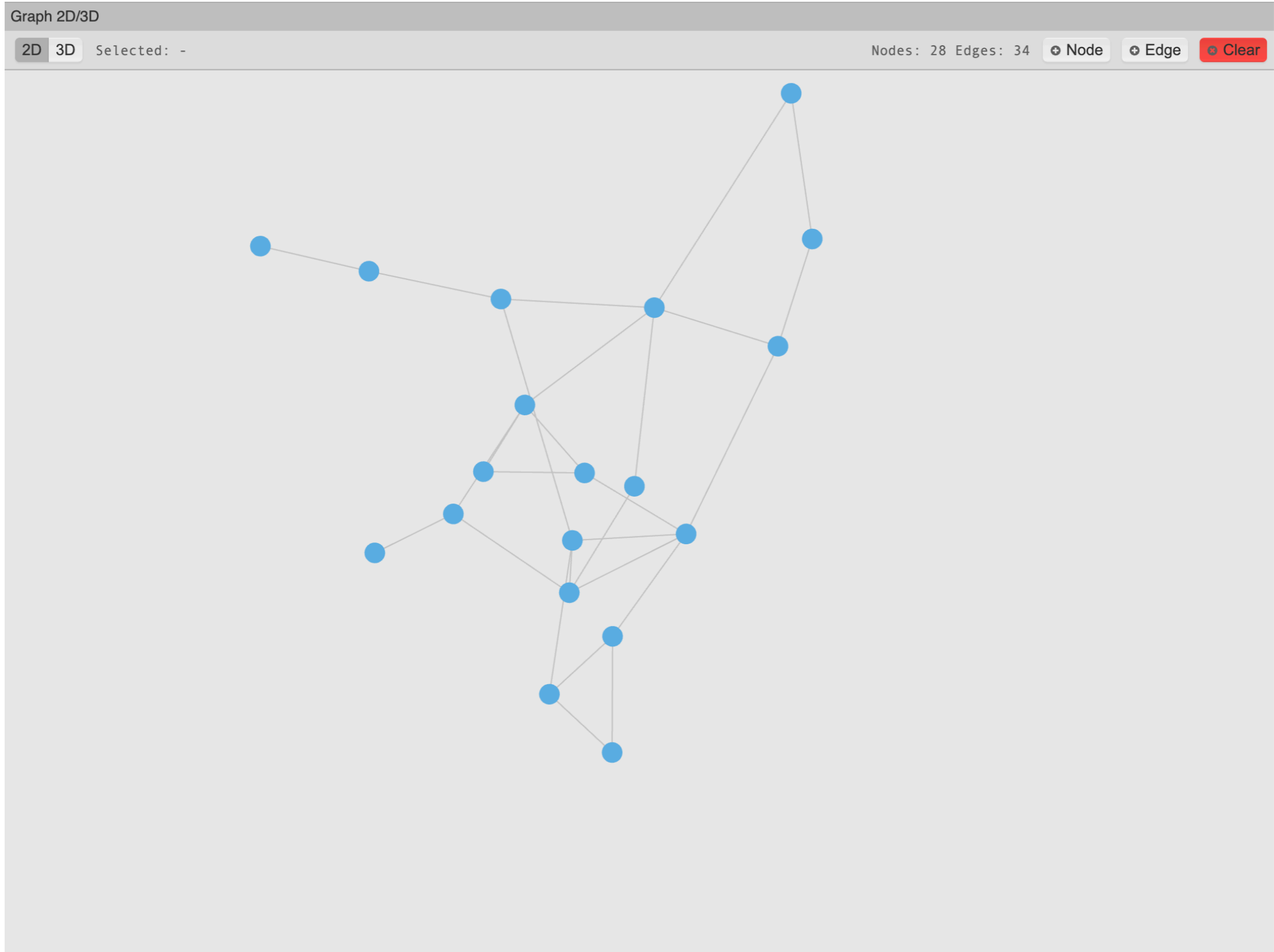
void job(FB *fb)
{
    SN *inp = & fb->inp1;
    SN *out = & fb->out1;
    SL *idx = & fb->idx1;
    {
        int i = 0;
        while (i < 3) {
            {
                (*(out + i))->v = (*(inp + i))->v + (fb->prm)->v;
                (*(out + i))->s = 0;
                (*(idx + i))->v = (*(inp + i))->s;
                (*(idx + i))->s = 0;
            }
            i++;
        }
        (fb->sum)->v = ((fb->out1)->v + (fb->out2)->v) + (fb->out3)->v;
        (fb->sum)->s = 0;
        return;
    }
}
      
```
- Inspector:** Shows details for the function `void job(FB *fb)`, including its location and type.
- Region Analysis:** A graph visualization showing the flow of memory regions.
  - Inputs:** `i` (green `RW(i)`), `inp` (orange `RW*`), `out` (orange `RW*`), `fb` (yellow `R*`), and `idx` (orange `RW*`).
  - Intermediate Regions:** Multiple yellow `R*` nodes, some with associated memory ranges (e.g., `0..63 [1]`, `64..95 [1]`, `96..127 ##`).
  - Outputs:** Various regions including `R(d)`, `R(i)`, `RW(d)` (green), `W(i)` (pink), `W(d)` (pink), `W(i)` (pink), and `RW(x)` (red).

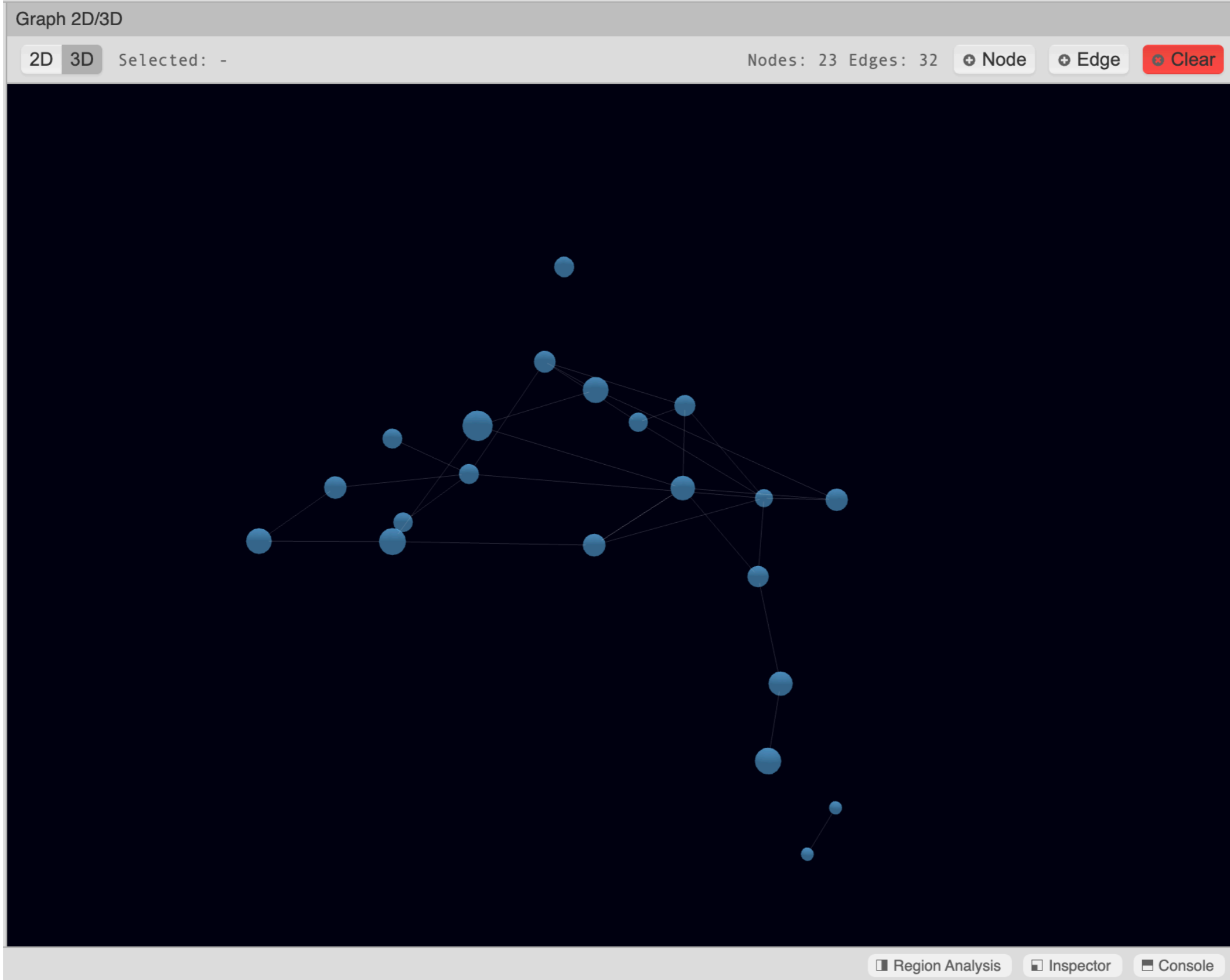
## What's Available ?

---

Rich Toolkit...



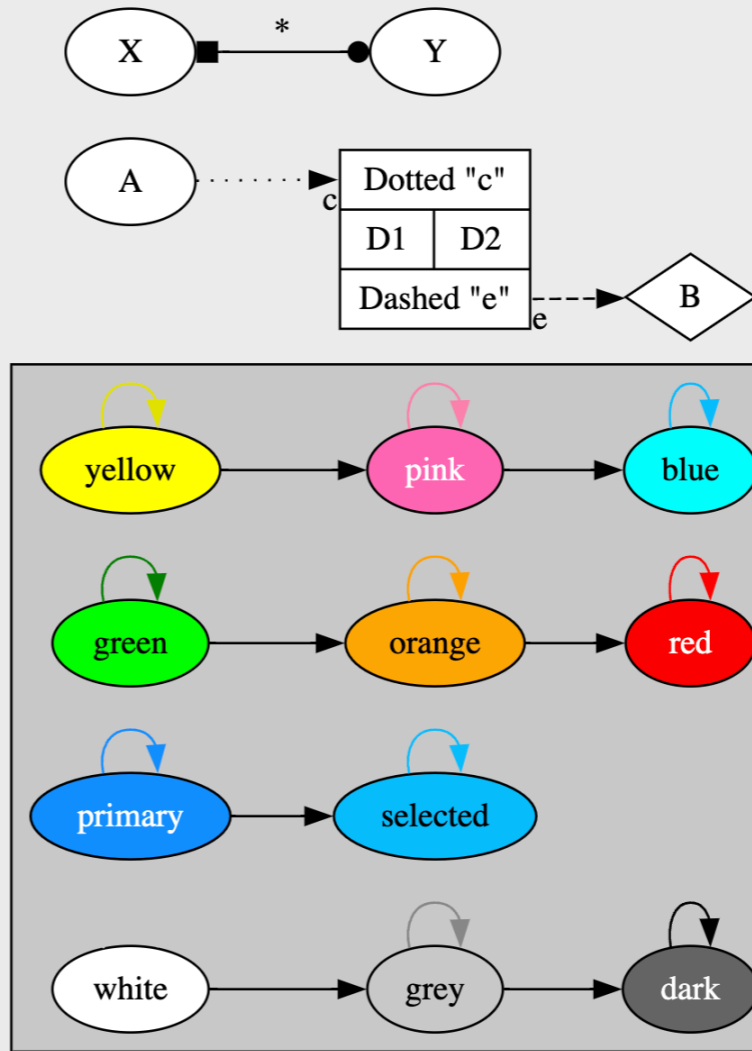




```

Diagram
Selected:
digraph {
  rankdir="LR"; bgcolor="none"; width=0.5;node [ style="filled"
  subgraph cluster_n0 {
    style="filled"; tooltip="Background Cluster"; fontcolor="bla
    n1; n2; n3; n4; n5; n6; n7; n8; n9; n10; n11;
  }
  n12 [ id="A"; label="A"; tooltip="A"; fontcolor="black"; fillc
  n13 [ id="B"; label="B"; shape="diamond"; tooltip="B"; fontcol
  n14 [ id="R"; shape="record"; label="<n15> Dotted \"c\"|{D1|D2
  n1 [ id="white"; label="white"; tooltip="white"; fontcolor="bl
  n2 [ id="grey"; label="grey"; tooltip="grey"; fontcolor="black
  n3 [ id="dark"; label="dark"; tooltip="dark"; fontcolor="white
  n4 [ id="primary"; label="primary"; tooltip="primary"; fontcol
  n5 [ id="selected"; label="selected"; tooltip="selected"; font
  n6 [ id="green"; label="green"; tooltip="green"; fontcolor="bl
  n7 [ id="orange"; label="orange"; tooltip="orange"; fontcolor=
  n8 [ id="red"; label="red"; tooltip="red"; fontcolor="white";
  n9 [ id="yellow"; label="yellow"; tooltip="yellow"; fontcolor=
  n10 [ id="blue"; label="blue"; tooltip="blue"; fontcolor="blac
  n11 [ id="pink"; label="pink"; tooltip="pink"; fontcolor="whit
  n17 [ id="X"; label="X"; tooltip="X"; fontcolor="black"; fillc
  n18 [ id="Y"; label="Y"; tooltip="Y"; fontcolor="black"; fillc
  n12 -> n14:n15 [ headlabel="c"; headtooltip="A -> R"; tooltip=
  n14:n16 -> n13 [ taillabel="e"; tailtooltip="R -> B"; tooltip=
  n4 -> n5 [ tooltip="primary -> selected"; arrowtail="none"];
  n1 -> n2 [ tooltip="white -> grey"; arrowtail="none"];
  n2 -> n3 [ tooltip="grey -> dark"; arrowtail="none"];
  n6 -> n7 [ tooltip="green -> orange"; arrowtail="none"];
  n7 -> n8 [ tooltip="orange -> red"; arrowtail="none"];
  n9 -> n11 [ tooltip="yellow -> pink"; arrowtail="none"];
  n11 -> n10 [ tooltip="pink -> blue"; arrowtail="none"];
  n1 -> n1 [ tooltip="white -> white"; color="#ccc"; arrowtail="n
  n2 -> n2 [ tooltip="grey -> grey"; color="#888"; arrowtail="n
  n3 -> n3 [ tooltip="dark -> dark"; color="black"; arrowtail="r
  n4 -> n4 [ tooltip="primary -> primary"; color="dodgerblue"; a
  n5 -> n5 [ tooltip="selected -> selected"; color="deepskyblue"
  n6 -> n6 [ tooltip="green -> green"; color="green"; arrowtail=
  n7 -> n7 [ tooltip="orange -> orange"; color="orange"; arrowta
  n8 -> n8 [ tooltip="red -> red"; color="red"; arrowtail="none"
  n9 -> n9 [ tooltip="yellow -> yellow"; color="#e5e100"; arrowt
  n10 -> n10 [ tooltip="blue -> blue"; color="deepskyblue"; arrc
  n11 -> n11 [ tooltip="pink -> pink"; color="palevioletred1"; a
  n17 -> n18 [ label="*"; labeltooltip="box to dot"; tooltip="bc
}

```

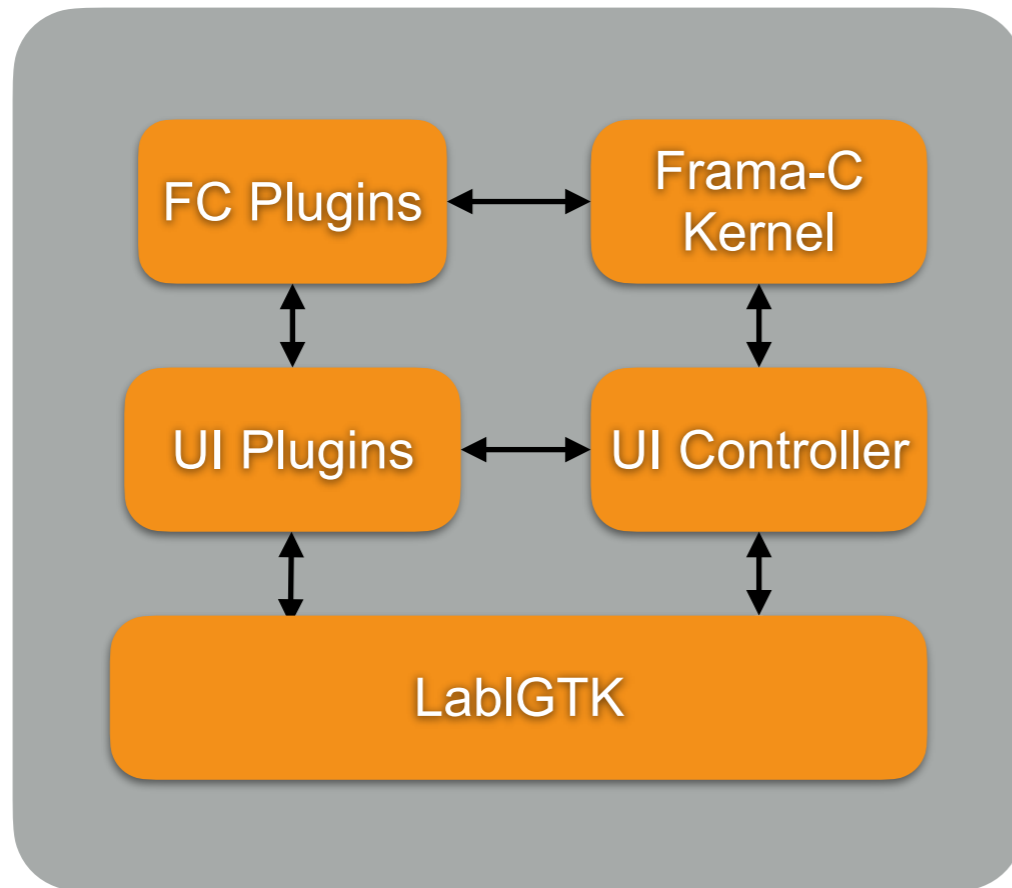


## What's Inside ?

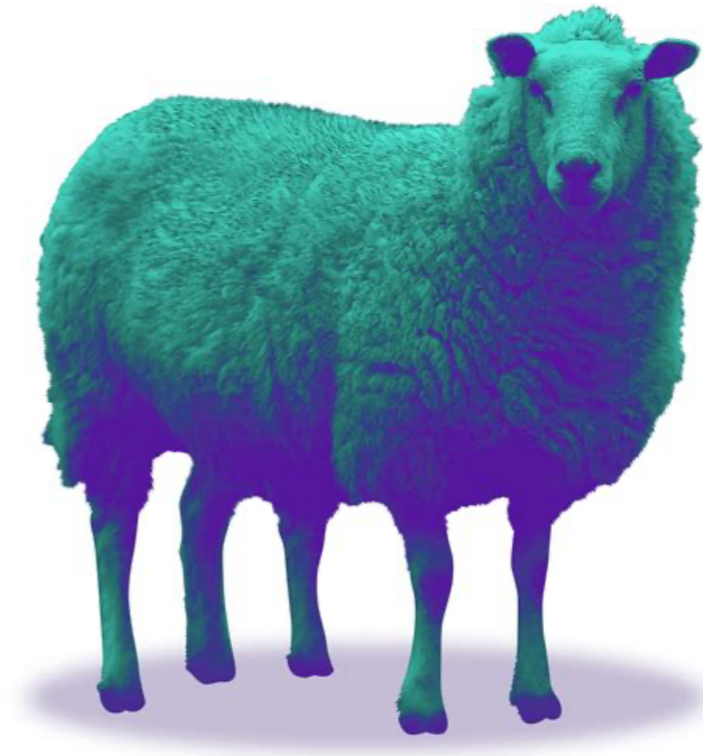
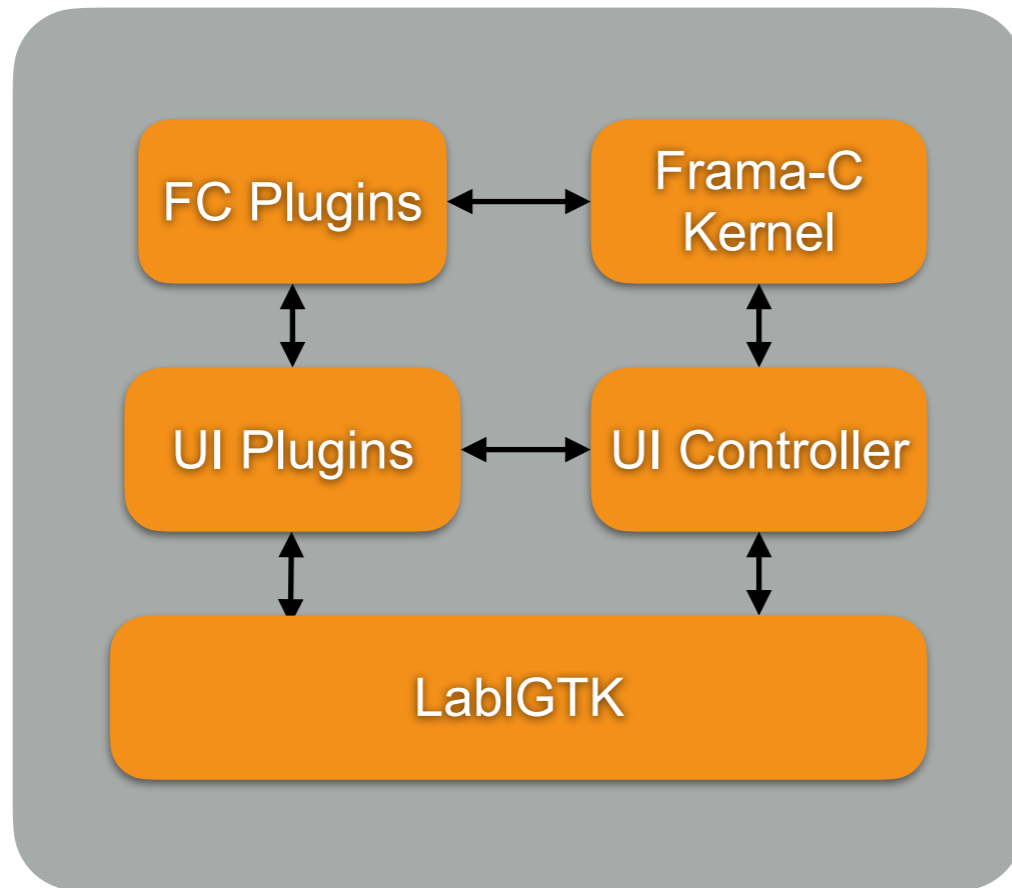
---

Not just a relooking...

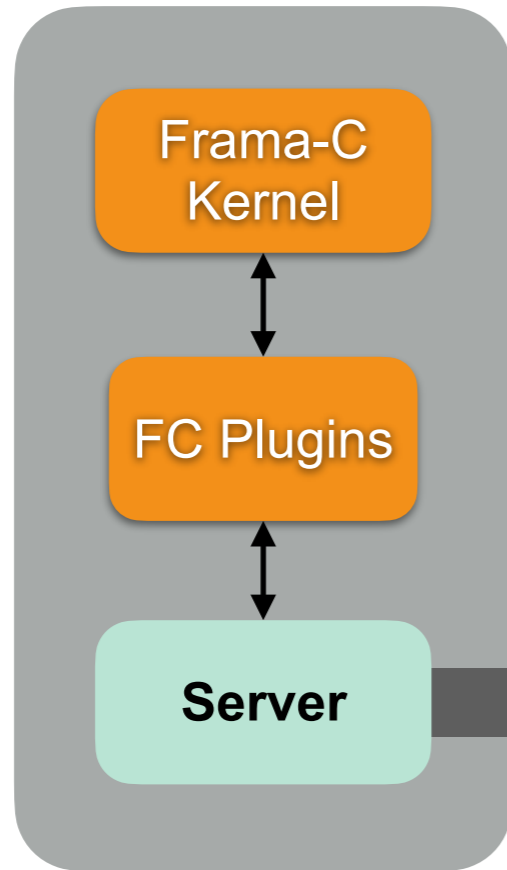
## frama-c-gui



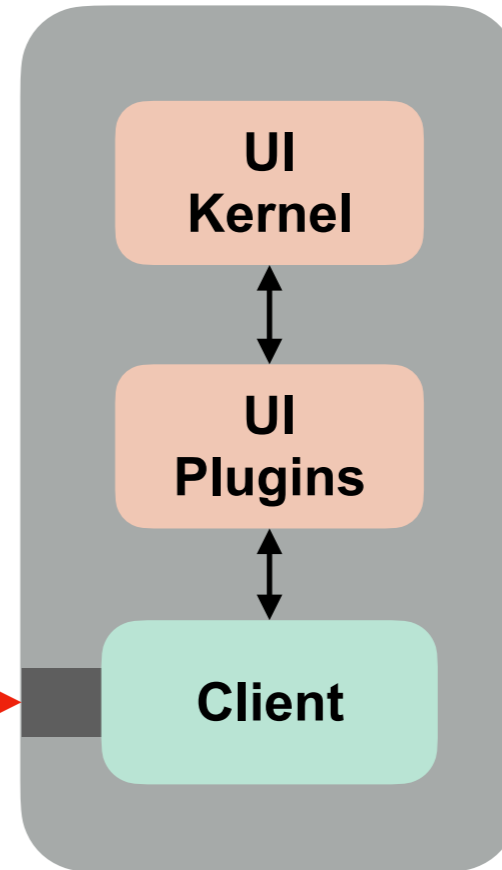
## frama-c-gui



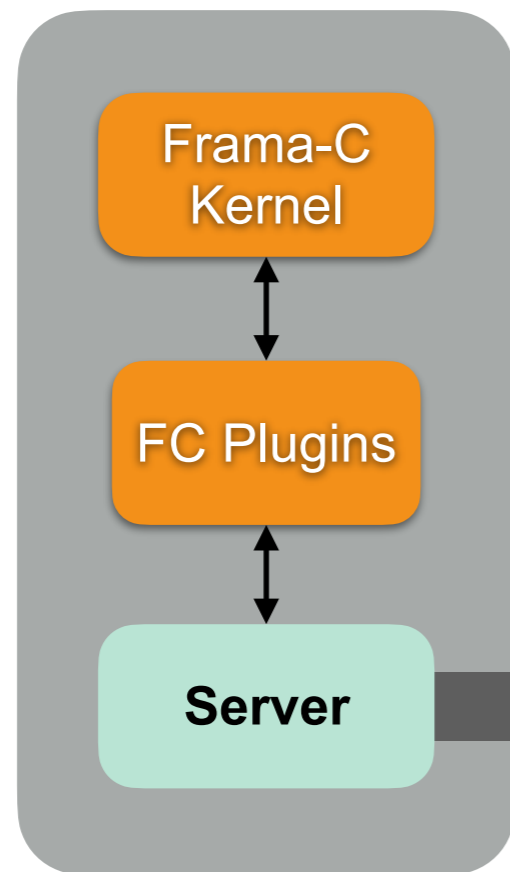
## frama-c



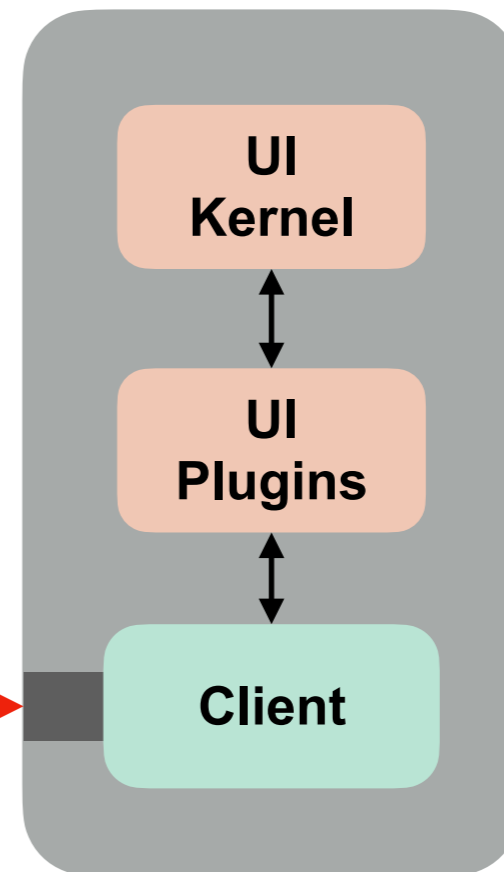
## Ivette



## frama-c



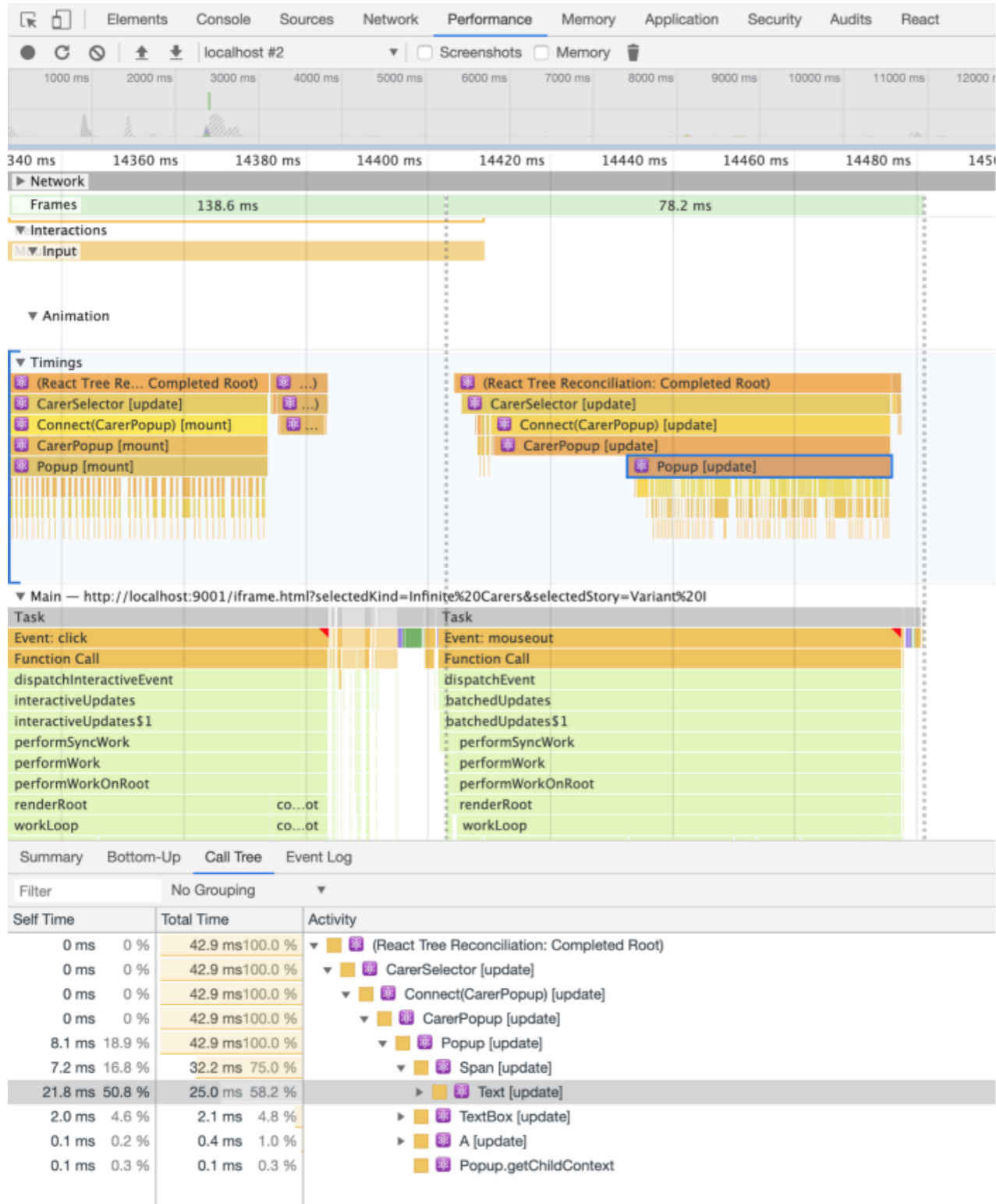
## Ivette



- ★ Command Driven (batch)
- ★ Synchronous
- ★ Intensive

- ★ Interactive
- ★ Asynchronous (threads)
- ★ Slow & Sparse





**< 1 ms**

few DOM Sync

**~10 ms**

sorting an array of 100,000 elements

**~100 ms**

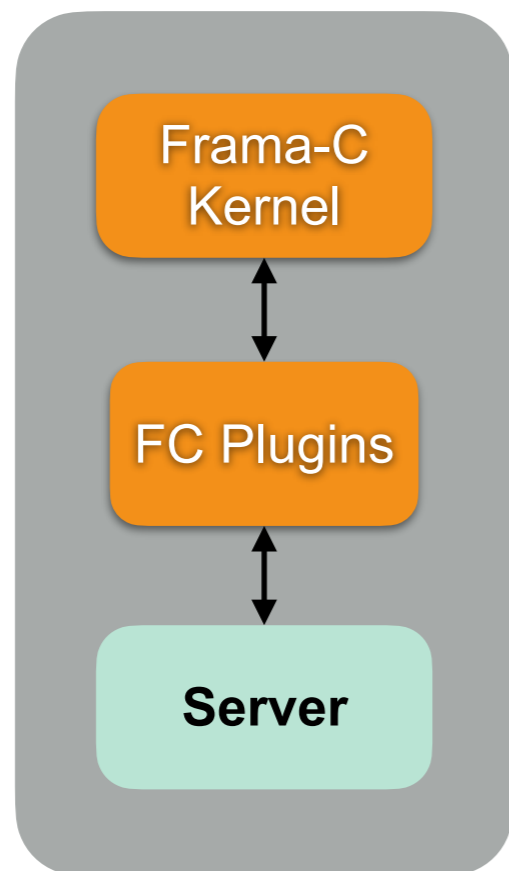
running SMT solver on a Frama-C/WP proof obligation

**~300 ms**

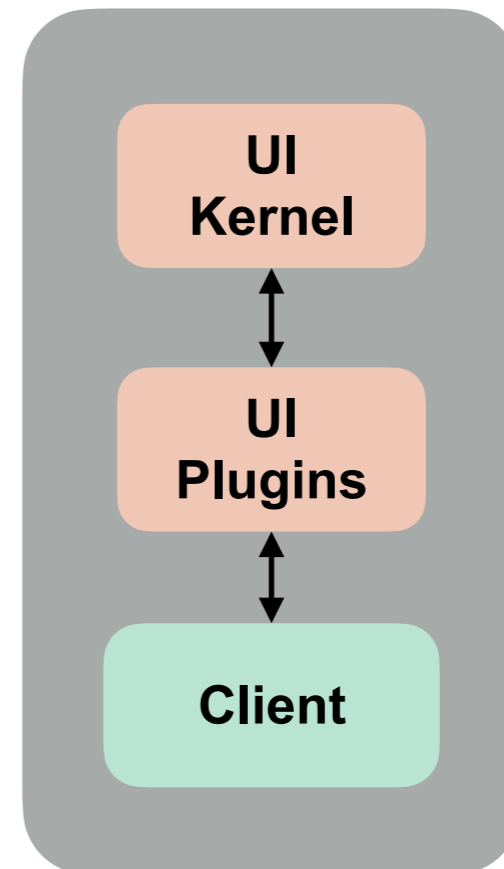
human expects UI feedback after a mouse click

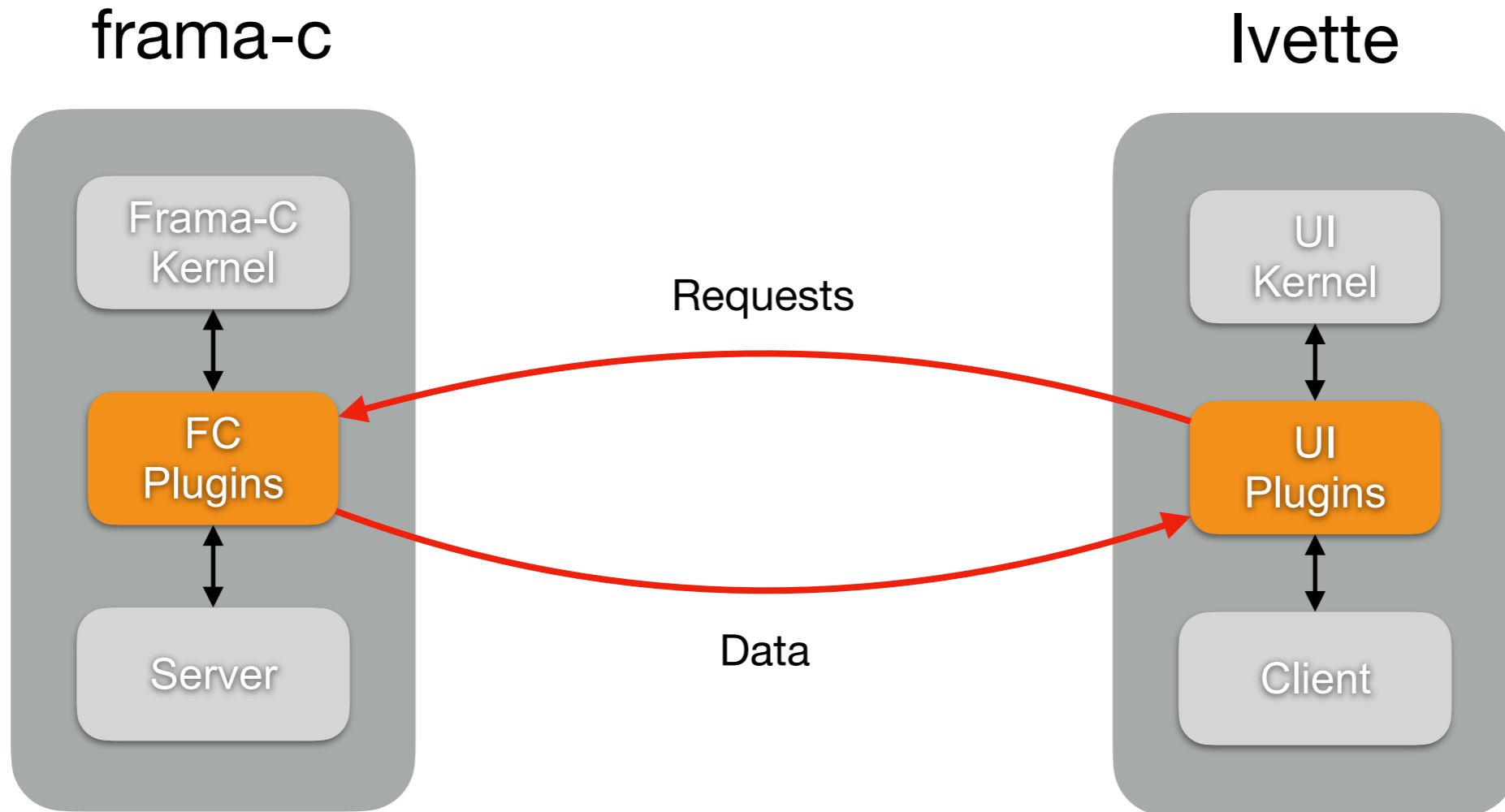


## frama-c



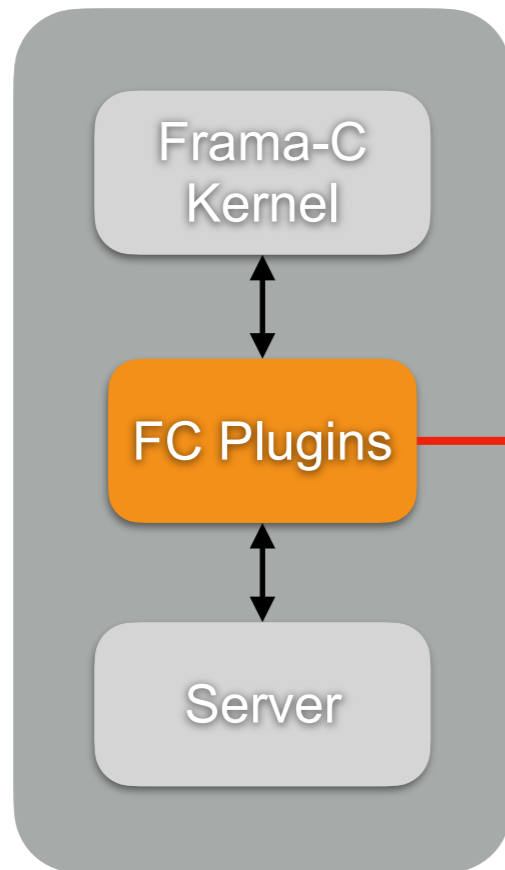
## Ivette





- ★ Declarative
- ★ Incremental
- ★ Cached
- ★ Sync / Async

## frama-c



262 loc

```

let () =
  Request.register
    ~package ~kind:`GET ~name:"regions"
    ~descr:(Md.plain "Returns computed regions for the given declaration")
    ~input:(module Kernel_ast.Decl)
    ~output:(module Regions)
    ~signals:[signal]
  begin fun decl ->
    try Memory.regions @@ map_of_declaration decl
    with Not_found -> []
  end
  
```

- ★ Command Driven (batch)
- ★ Synchronous
- ★ Intensive

- ★ JSON Data Serialization
- ★ Registered Data
- ★ Registered Requests
- ★ Idiomatic OCaml Code
- ★ Auto-documented

## Frama-C Server

### Plugin Wp

#### WP Main Services

- plugins.wp.goal (DATA)
- plugins.wp.prover (DATA)
- plugins.wp.provers (STATE)
- plugins.wp.signalProvers (SIGNAL)
- plugins.wp.getProvers (GET)
- plugins.wp.setProvers (SET)
- plugins.wp.process (STATE)
- plugins.wp.signalProcess (SIGNAL)
- plugins.wp.getProcess (GET)
- plugins.wp.setProcess (SET)
- plugins.wp.timeout (STATE)
- plugins.wp.signalTimeout (SIGNAL)
- plugins.wp.getTimeout (GET)
- plugins.wp.setTimeout (SET)
- plugins.wp.ProverInfos (ARRAY)
- plugins.wp.signalProverInfos (SIGNAL)
- plugins.wp.ProverInfosData (DATA)
- plugins.wp.fetchProverInfos (GET)
- plugins.wp.reloadProverInfos (GET)
- plugins.wp.result (DATA)
- plugins.wp.status (DATA)
- plugins.wp.stats (DATA)
- plugins.wp.goals (ARRAY)
- plugins.wp.signalGoals (SIGNAL)
- plugins.wp.goalsData (DATA)
- plugins.wp.fetchGoals (GET)
- plugins.wp.reloadGoals (GET)
- plugins.wp.generateRTEGuards (EXEC)
- plugins.wp.startProofs (EXEC)
- plugins.wp.serverActivity (SIGNAL)
- plugins.wp.getScheduledTasks (GET)
- plugins.wp.cancelProofTasks (SET)
- WP Interactive Prover
- WP Tactics

## plugins.wp.goalsData (DATA)

Data for array rows `goals`

```
goalsData ::= { fields... }
```

Field	Format	Description
"wpo"	goal	Entry identifier.
"marker"	marker	Associated Marker
"scope" (opt.)	decl	Associated declaration, if any
"property"	marker	Property Marker
"fct" (opt.)	string	Associated function name, if any
"bhv" (opt.)	string	Associated behavior name, if any
"thy" (opt.)	string	Associated axiomatic name, if any
"name"	string	Informal Property Name
"smoke"	boolean	Smoking (or not) goal
"passed"	boolean	Valid or Passed goal
"status"	status	Verdict, Status
"stats"	stats	Prover Stats Summary
"proof"	boolean	Proof Tree
"script" (opt.)	string	Script File
"saved"	boolean	Saved Script

## plugins.wp.fetchGoals (GET)

Data fetcher for array `goals`

```
input ::= number
```

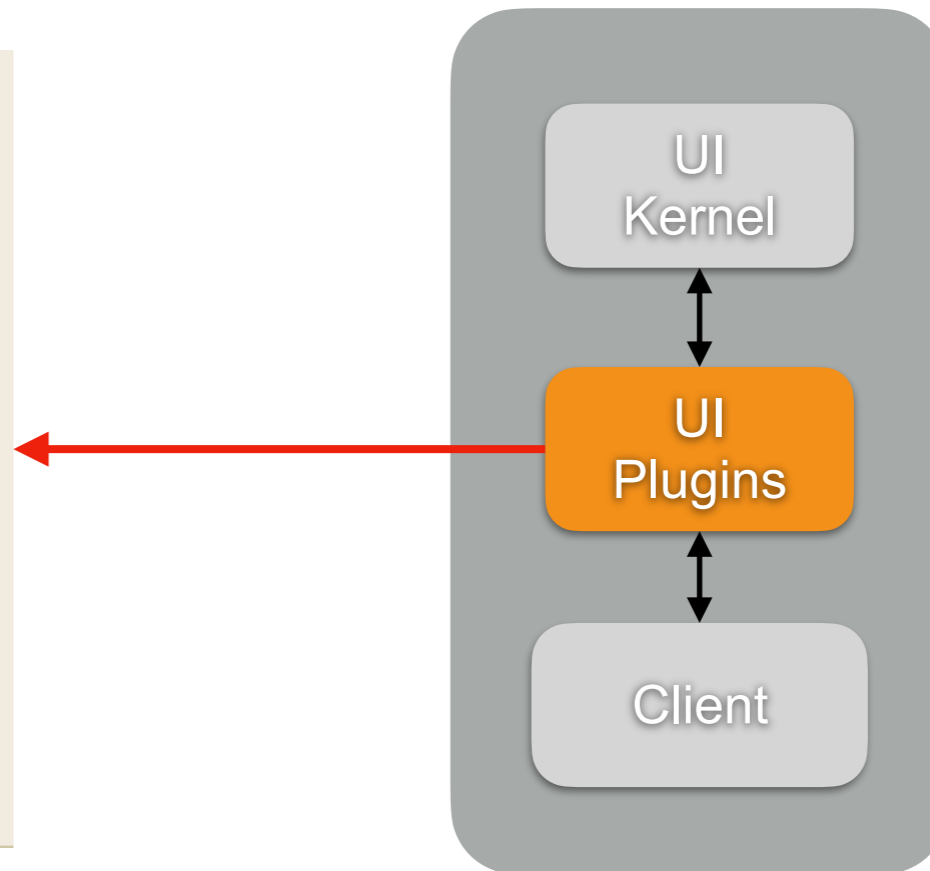
```
output ::= { output... }
```

Output	Format	Description
"reload"	boolean	array fully reloaded
"removed"	goal [ ]	removed entries
"updated"	goalsData [ ]	updated entries
"pending"	number	remaining entries to be fetched

223 loc

```
function RegionAnalys(): JSX.Element {
  // ....
  const [kf, setKf] = React.useState<States.Scope>();
  const scope = States.useCurrentScope();
  const regions = States.useRequest(Region.regions, kf) ?? [];
  // ...
  return (
    <>
      <Tools.ToolBar>
        <Label label='Function' />
        <LCD className='region-lcd' label={kfName ?? '---'} />
        <Tools.Button ... />
        // ...
      </Tools.ToolBar>
      <MemoryView regions={regions} />
    </>
  );
}
```

## Ivette



- ★ Generated Request API
- ★ Idiomatic X-TypeScript Code
- ★ Rich Toolkit

- ★ Interactive
- ★ Asynchronous (threads)
- ★ Slow & Sparse

Ivette
declaration

Console
Eva Values\*
Eva Summary
WP View
Source Code\*

Types 3

Variables 0

Functions

job

AST

```

void job(FB *fb)
{
  SN *inp = & fb->inp1;
  SN *out = & fb->out1;
  SL *idx = & fb->idx1;
  {
    int i = 0;
    while (i < 3) {
      {
        *(out + i)->v = *(inp + i)->v + (fb->prm)->v;
        *(out + i)->s = 0;
        *(idx + i)->v = *(inp + i)->s;
        *(idx + i)->s = 0;
      }
      i ++;
    }
    (fb->sum)->v = ((fb->out1)->v + (fb->out2)->v) + (fb->out3)->v;
    (fb->sum)->s = 0;
    return;
  }
}
          
```

Region Analysis

Function job

Inspector

Function void job(FB \*fb)

Location src/plugins/region/tests/region/fb\_SORT.i:18

Var global

Type void (FB \*fb)

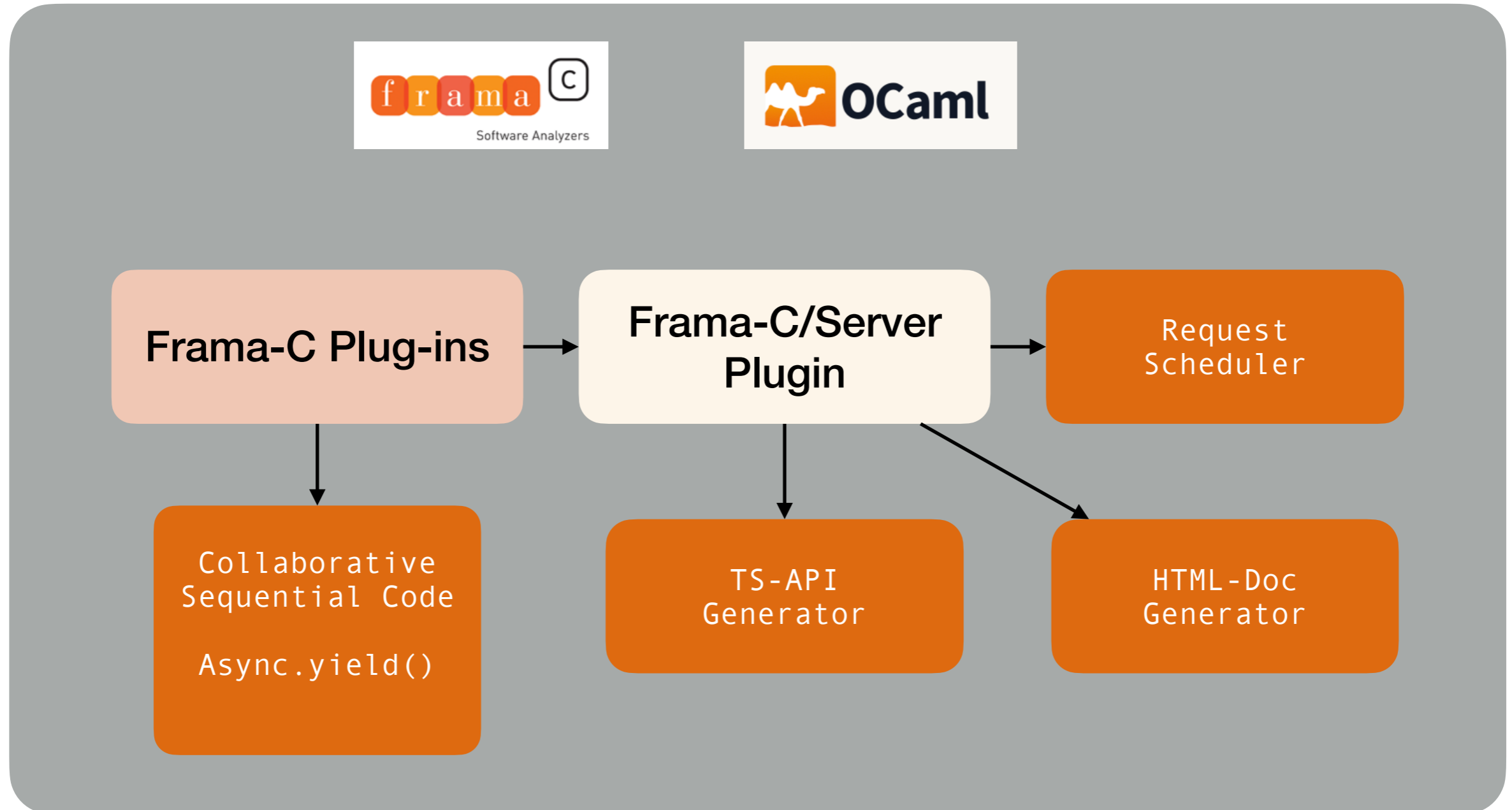
Region Analysis
Inspector
Console

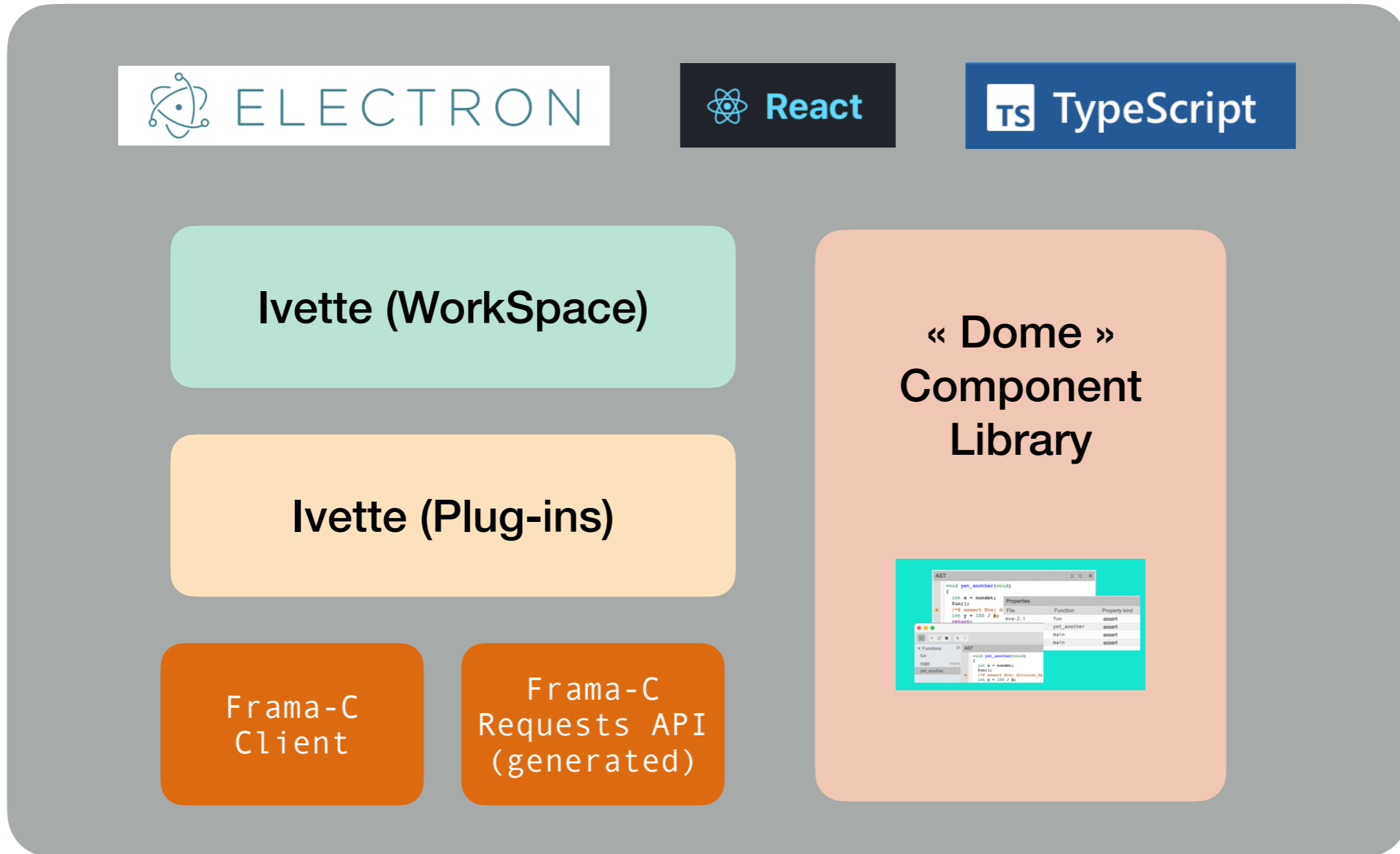
# What's that kind of Magic ?

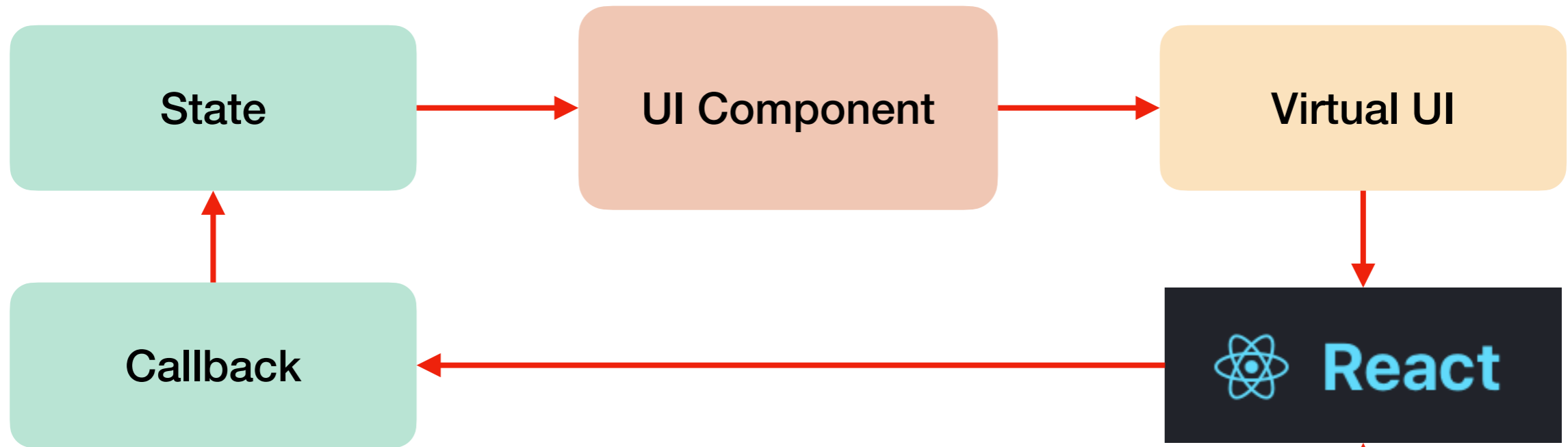
---

— Middleware —

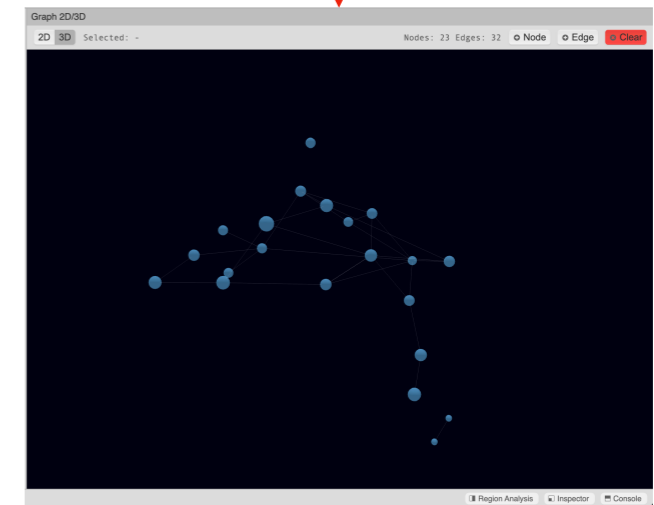


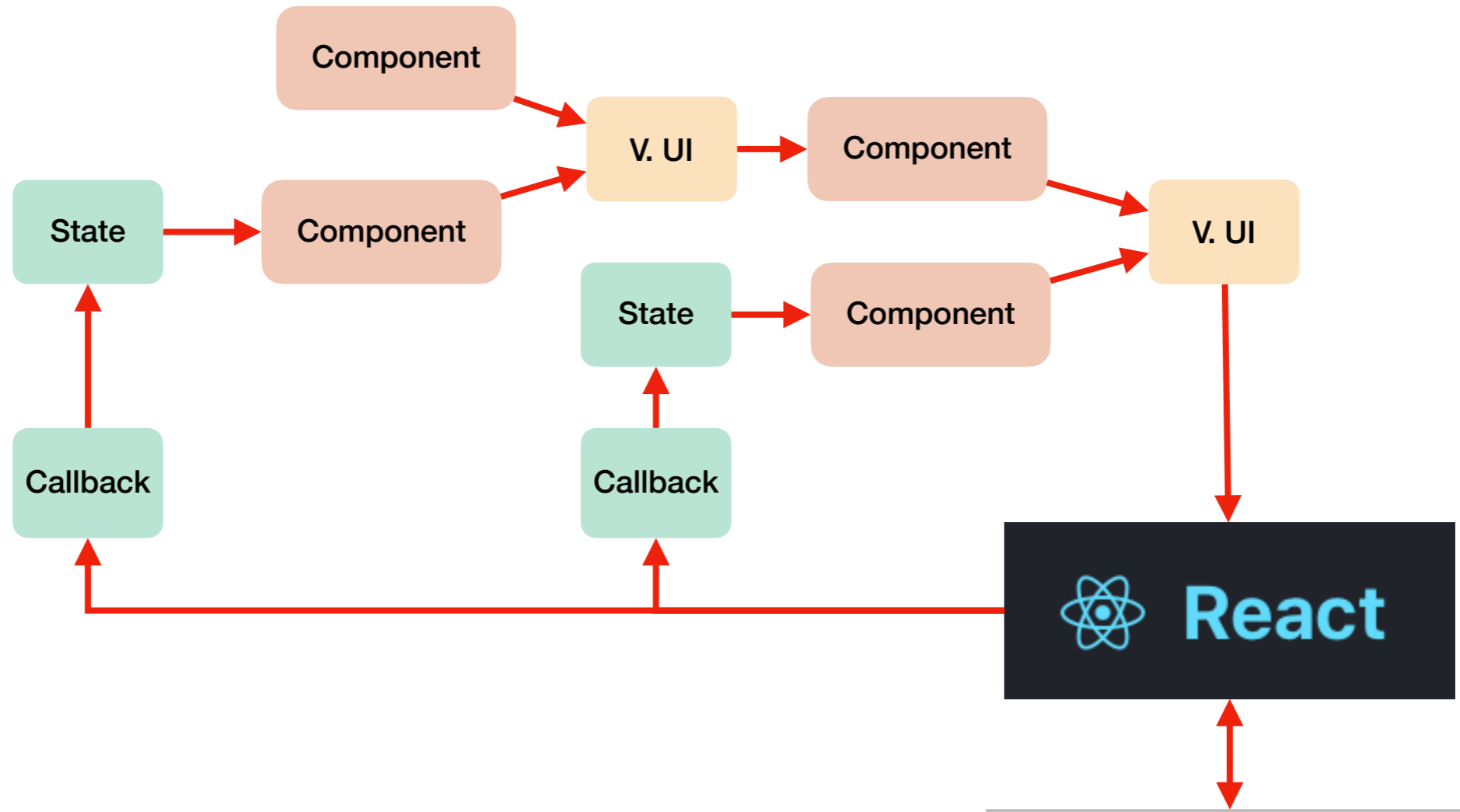




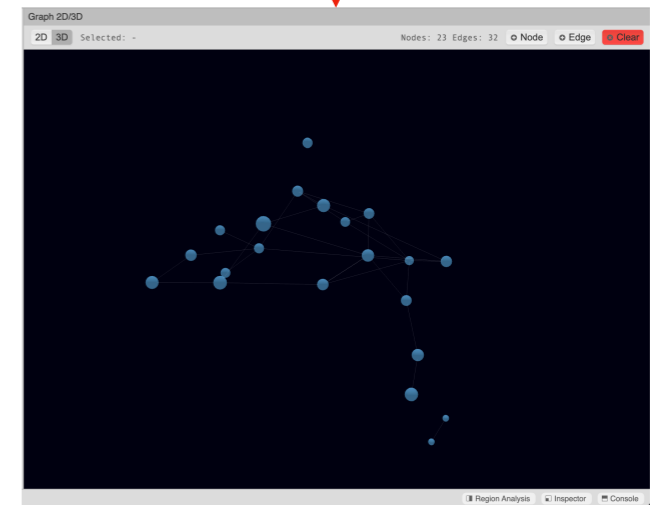


- ★ Simple Model
- ★ Functional Essence
- ★ Chromium Engine

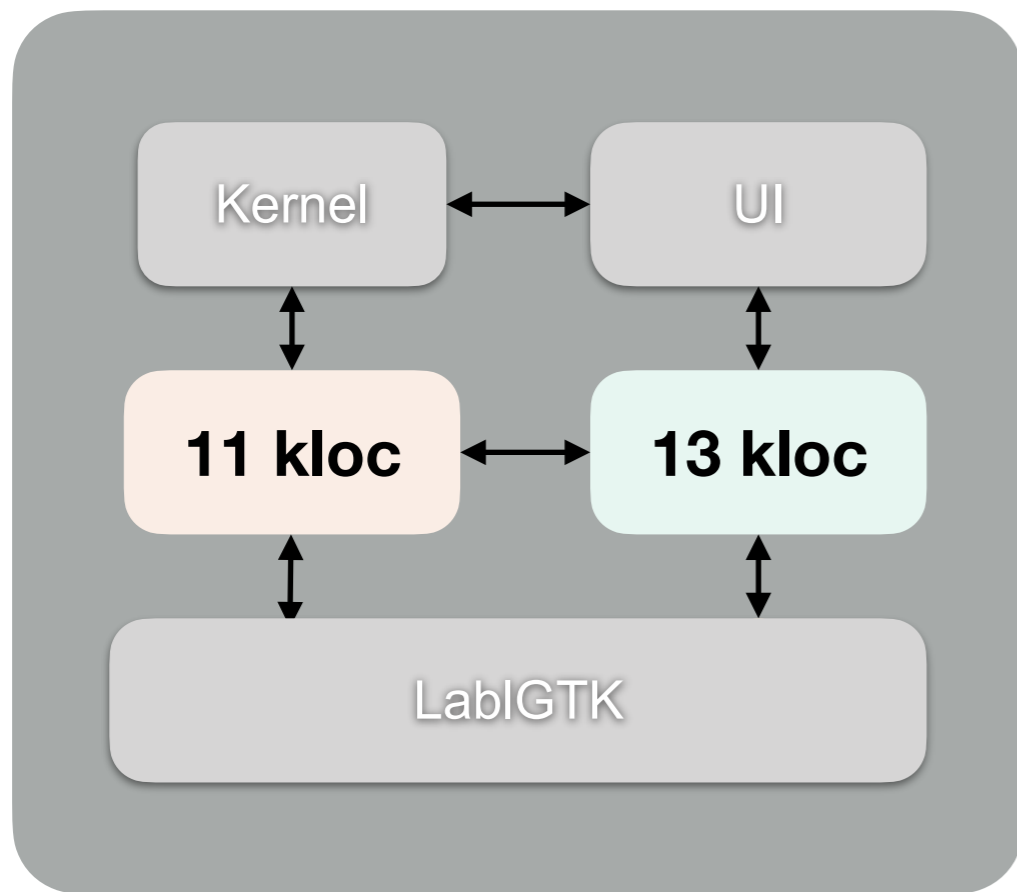




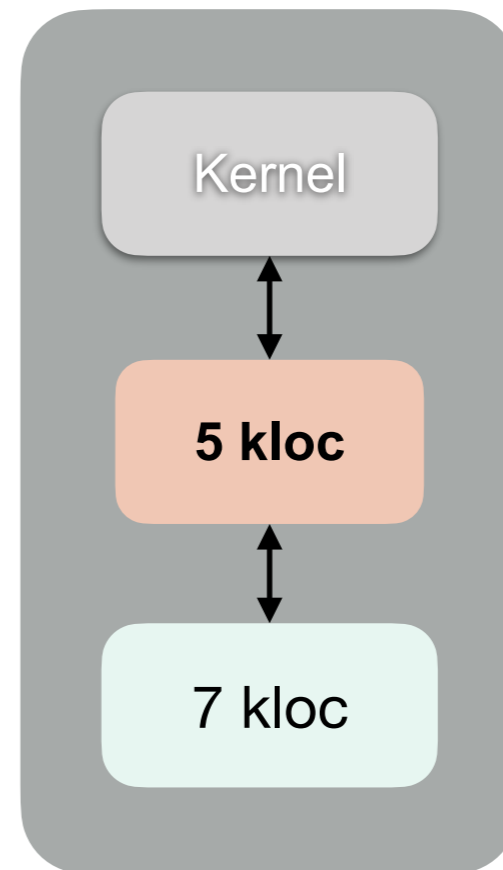
- ★ Functional
- ★ Compositional
- ★ Scales
- ★ Rich Ecosystem



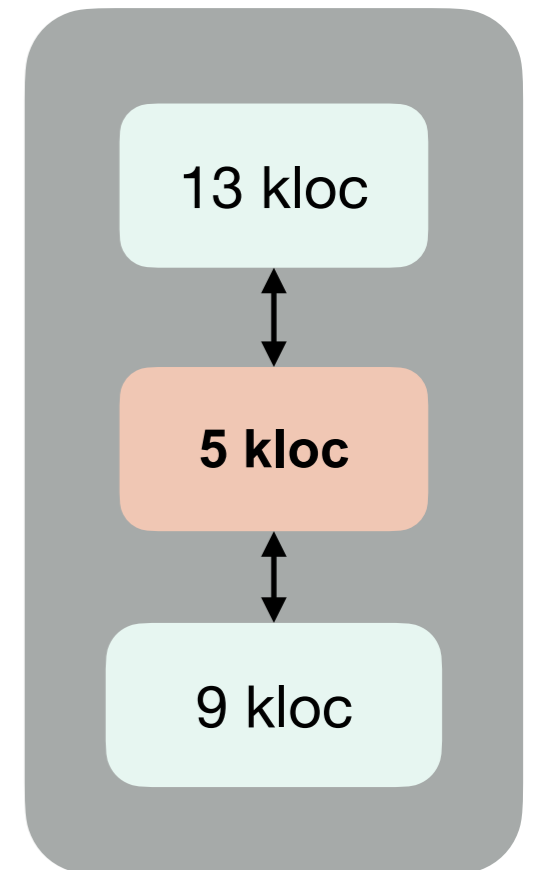
## frama-c-gui



## frama-c



## Ivette



## Clear Benefits

- ★ Skills Separation
- ★ Lightweight Plugins
- ★ Smooth Learning Curve

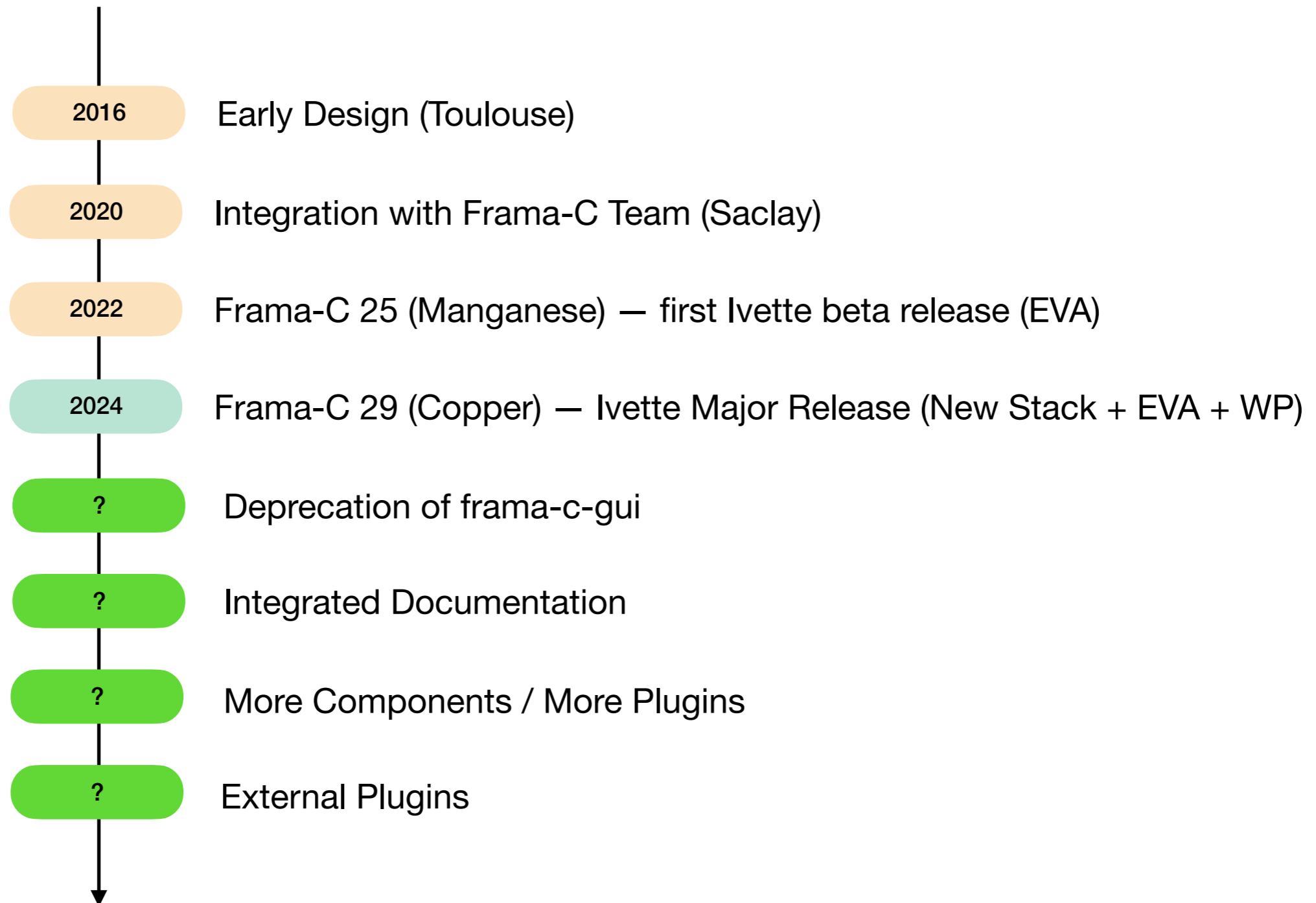
## Attention Points

- ★ Outsourcing Still Difficult
- ★ Heavyweight Infrastructure
- ★ Web Ecosystem is Volatile

## What's Next ?

---

— Roadmap —







D. Buhler



L. Correnson



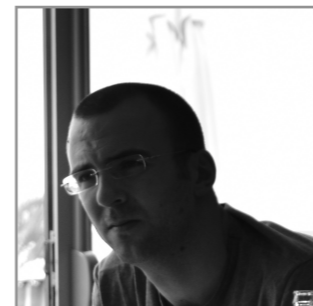
V. Perrelle



M. Jacquemin



A. Maroneze



M. Alberti



R. Lazarini

*And many others...*

Thanks !

<https://frama-c.com>

