



REFACTORERL ERLANG PROGRAMOK STATIKUS ELEMZÉSE

TÓTH MELINDA, BOZÓ ISTVÁN

Az Alkalmazásiterület-specifikus nagy megbízhatóságú informatikai megoldások című projekt a Nemzeti Kutatási Fejlesztési és Innovációs Alapból biztosított támogatással, a Tématerületi kiválósági program (TKP2020-NKA-06, Nemzeti Kihívások Alprogram) finanszírozásában valósult meg.



NATIONAL RESEARCH, DEVELOPMENT
AND INNOVATION OFFICE
HUNGARY

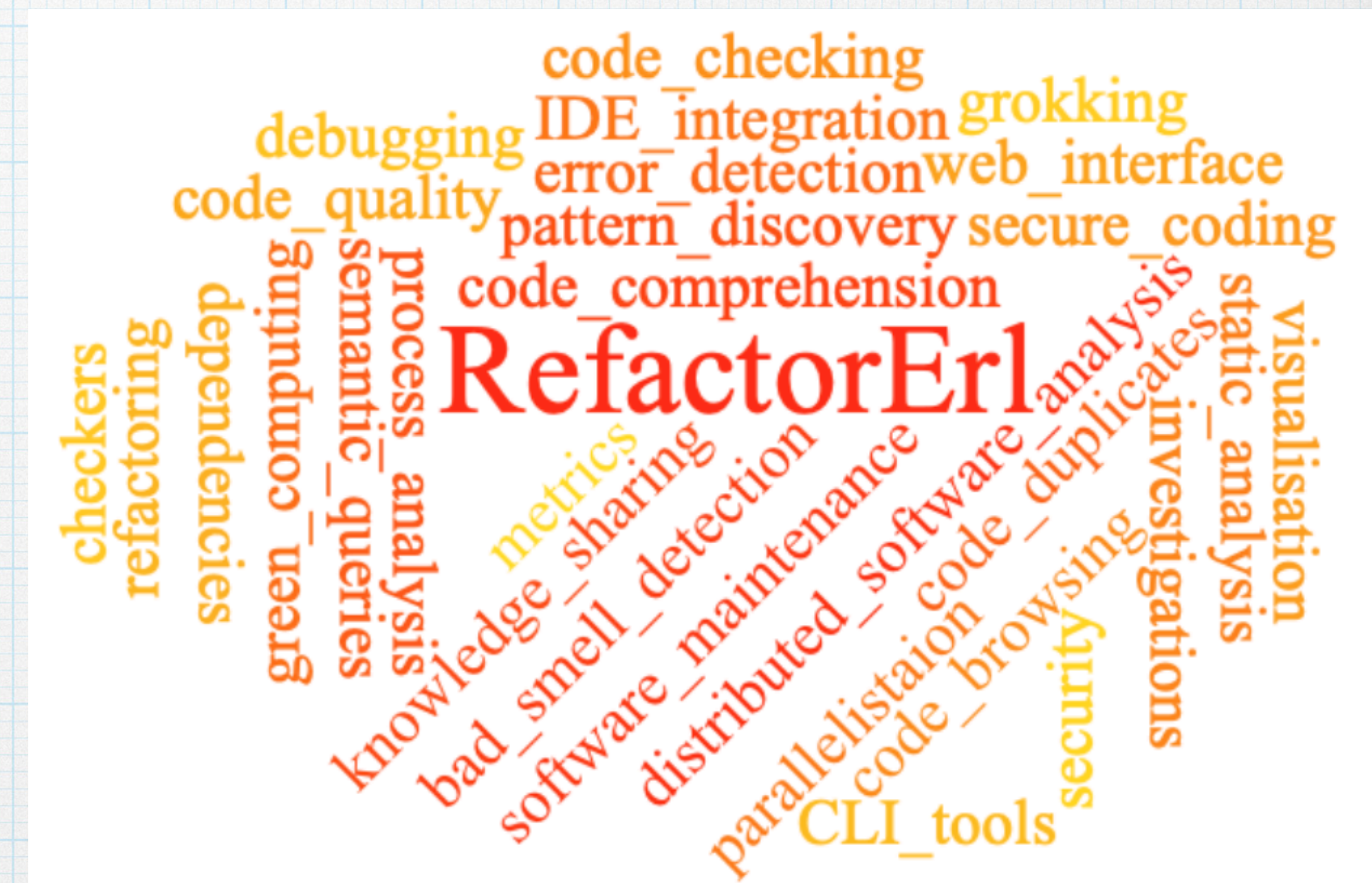
PROGRAM
FINANCED FROM
THE NRDI FUND

Kik vagyunk mi?

Erlang statikus elemző K+F projekt

Elemzések & transzformációk

plc.inf.elte.hu/erlang



Összefoglaló az elmúlt 1 évről

Korábban már megkezdett kutatások összefoglalása publikációkban

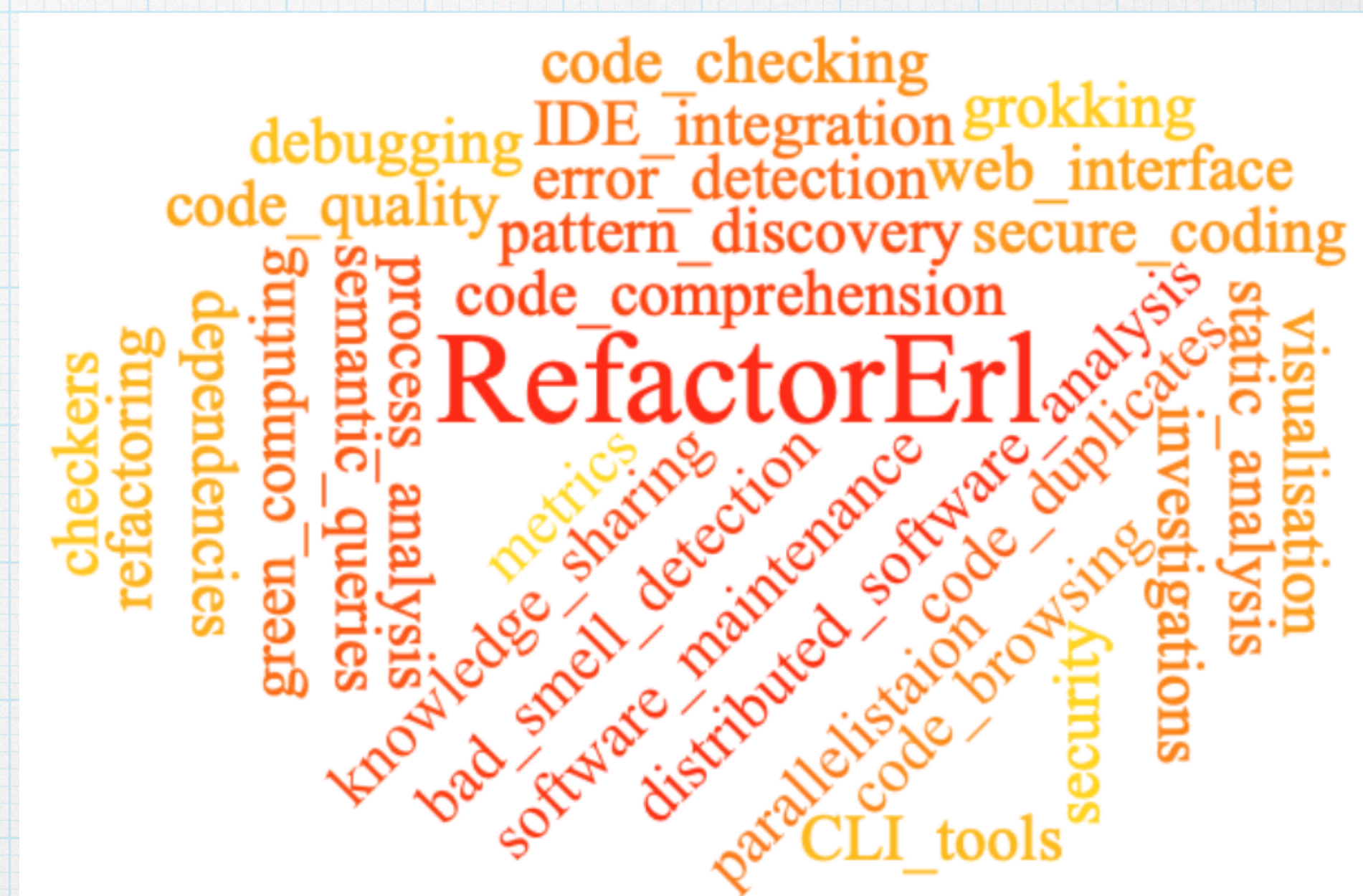
Major revision cikkek javítása

Security checkerek megvalósítása, finomítása, pontosítása

ReafctorErl reprezentáció bővítése: BEAM decompiling

Észköz hibáinak a javítása

Szimbolikus végrehajtás hibadetektálásra



Eredmények az elmúlt 1 évből

Folyóirat cikkek (2 + 2 + 9 + 2)

Konferencia előadások (1 1)

Konferencia cikk (2), absztrakt (9)

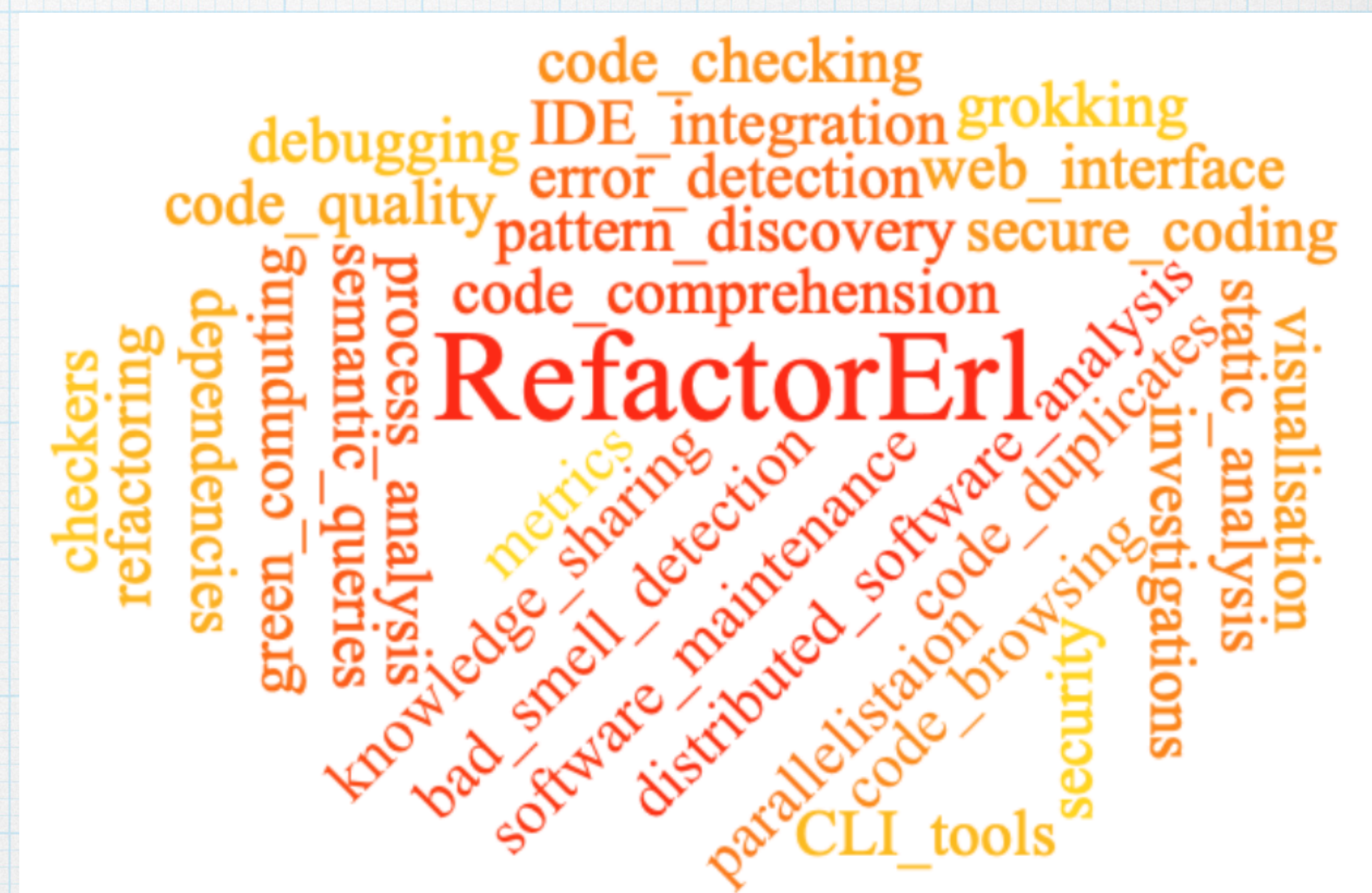
Meghívott előadás (3)

Nemzetközi kapcsolat (SSQSA)

Nemzetközi projekt kapcsolat (COST CA19135 - CERCIRAS)

7 OTDK díj (+ OTDK szervezés)

Ipari kapcsolat (Ericsson, OTP)



A csoport az elmúlt 1 évben

2 kutató, 1 doktori hallgató

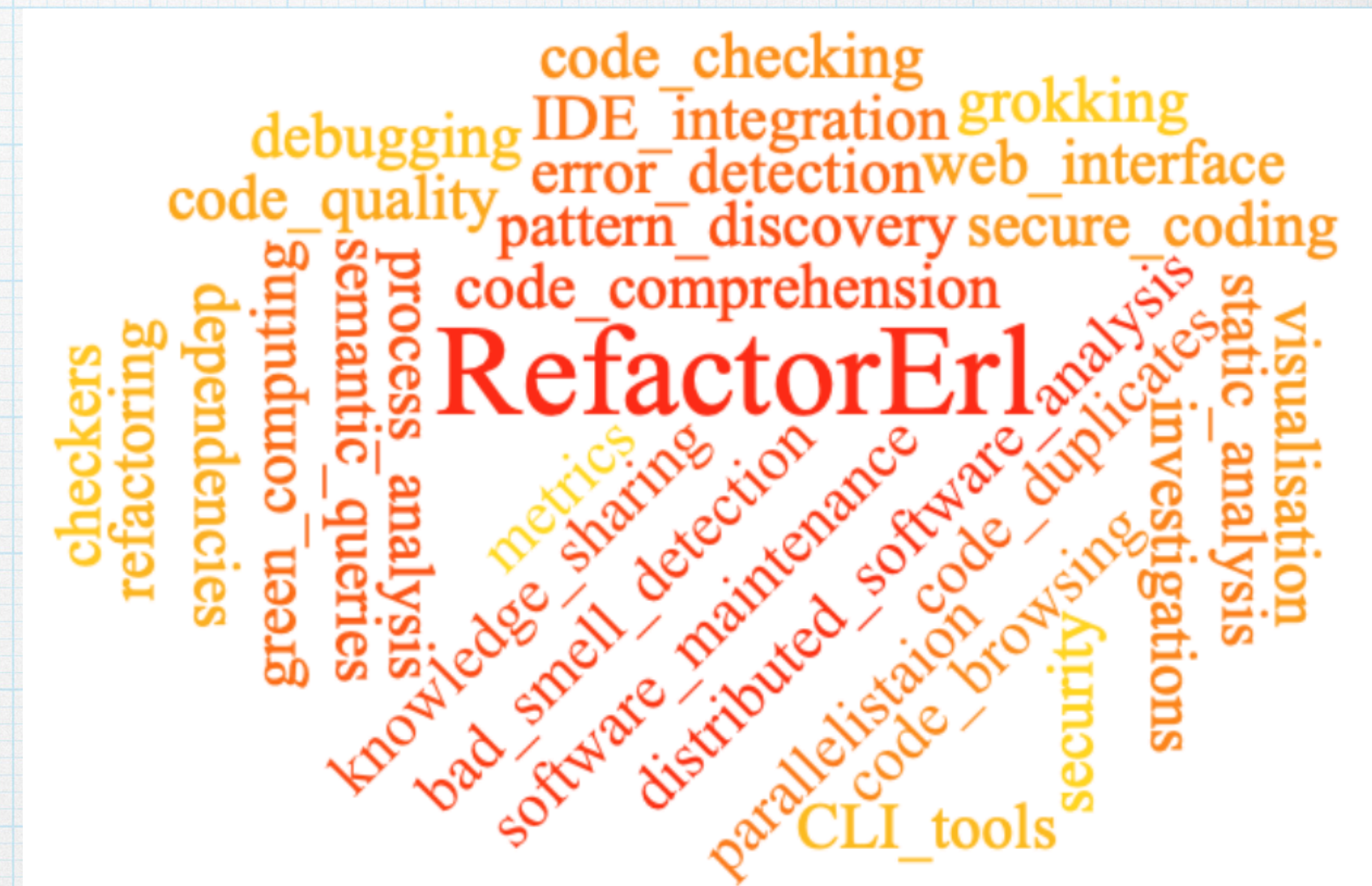
6 (+ lemorzsolódás) MSc

7 (+ 3) BSc

Ebből 2 BSc szakmai gyakorlat

6 cikkírást befejező hallgató novemberig

3 kapcsolódó doktori hallgató



Miért hasznos?

Lerövidítheti a betanulási időt

Hibamegoldási időt csökkentheti

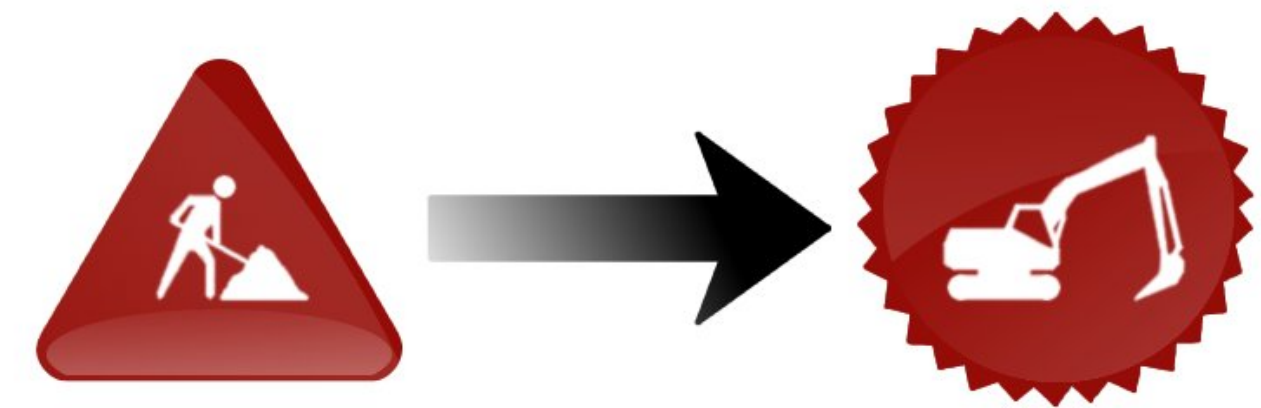
Támogatja a csapatmunkát

Kódminőség javítása

Megkönnyíti a mindennapos időigényes feladatokat

Támogatja a biztonságos szoftverfejlesztést

Effective software maintenance

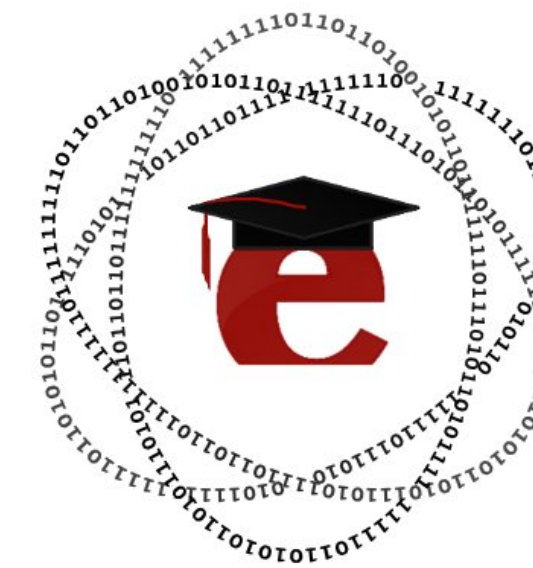


Fő funkciók

Kódmegértés támogatás

Refactoring/Alkalmazás újrastrukturálás

Kód ellenőrzés: bonyolultság/minőség/stílus/
sérülékenységek/különböző tulajdonságok



Knowledge sharing

Statikus elemző keretrendszer

Fordítási-idejű elemzések

Függvények, változók, rekordok, stb.
Láthatóság, hatókör, élettartam
Statikus és dinamikus referenciák

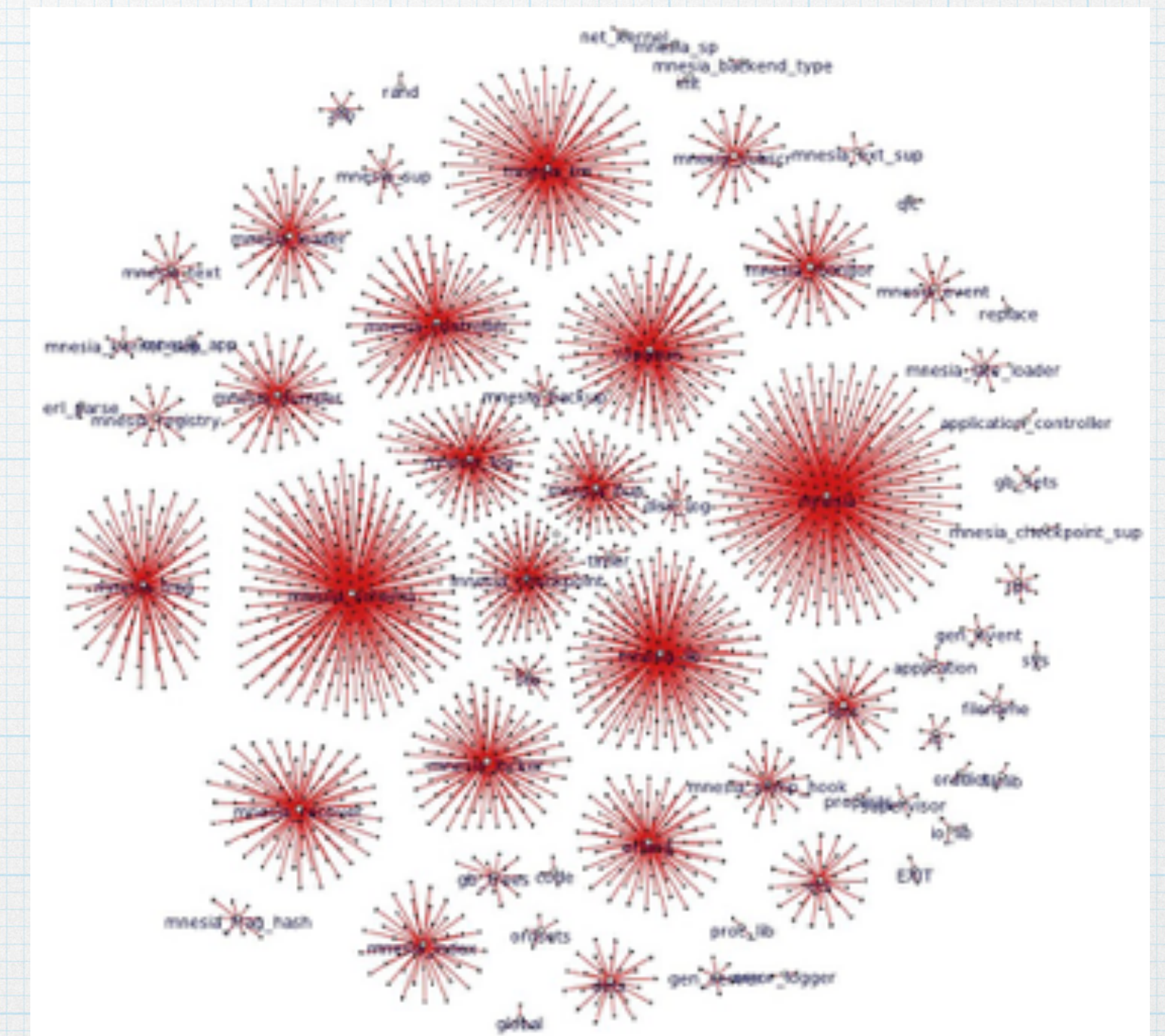
Mellékhatások
Adat-, Vezérlés-folyam
Dinamikus függvényhívás
Rejtett függőségek

in Erlang
for Erlang

in Erlang
for Erlang

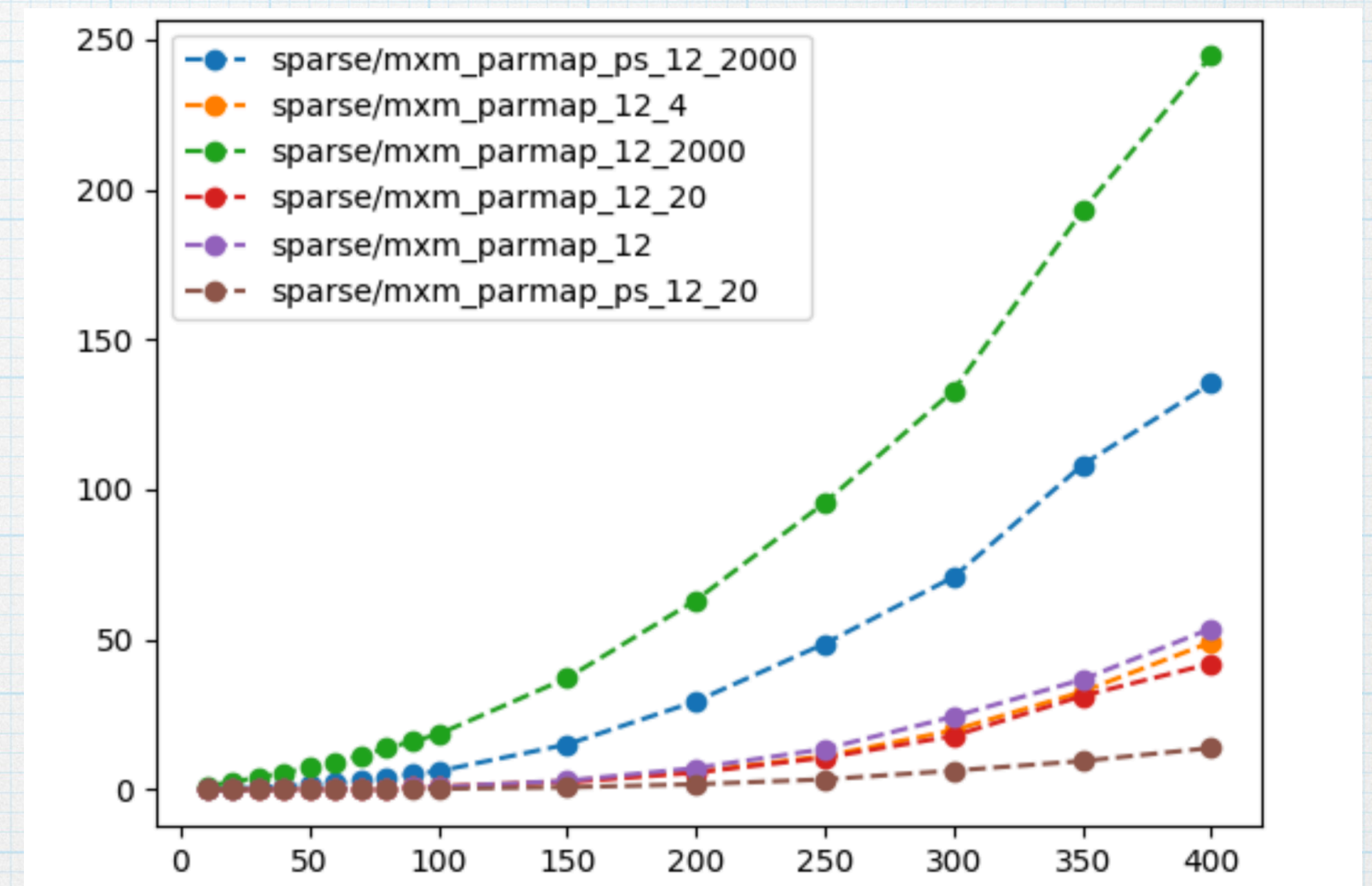
Program fejlesztés támogatása ...

Szemantikus lekérdezések
Szoftver bonyolultsági metrikák
"Bad smellek" keresése
Duplikátumok keresése és eliminálás
Szoftver klaszterezés
Függőségek számolása és
vizualizációja
Biztonságos programozás



És sok egyéb kutatás...

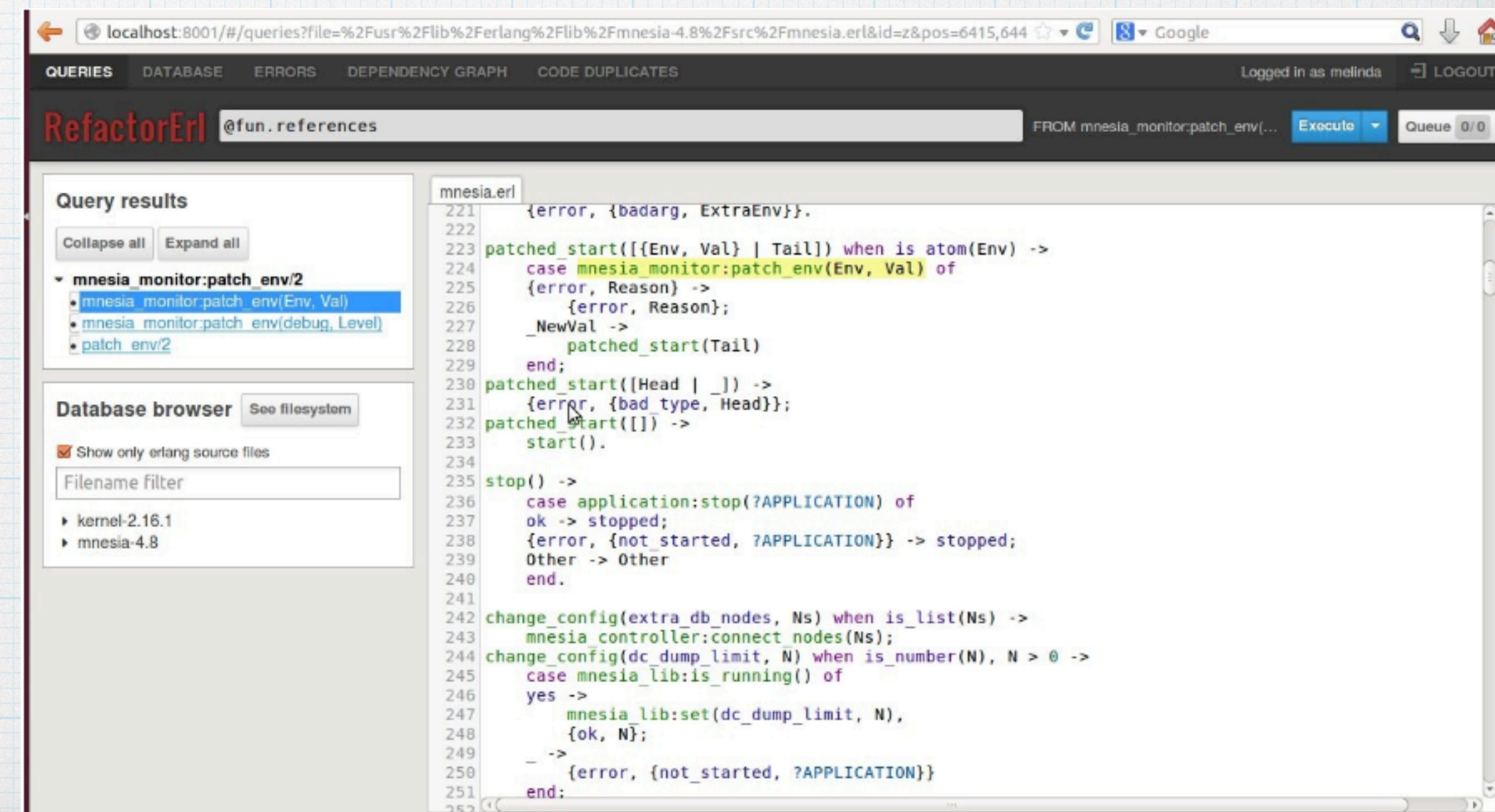
Folyamatok kapcsolatainak/üzenetek
felderítése
Program szeletelés tesztesetek szűrésére
OTP viselkedések elemzése
Kód-visszafejtés
Mintafelismerés és automatikus párhuzamosítás
Ad-hoc párhuzamosítás
Elosztott programok elemzése és
transzformálása
Stílus javítás
Statikus és dinamikus elemzések ötvözése
“Green computing”



Code Checking - Kód ellenőrzés

Mi volt eddig?
Szemantikus Lekérdező Nyelv

Mi az új?
Új kliens: DRC++



Semantic Query Language

Segít a hibakeresésben
kódmegértésben
tanulásban

Beépített és testreszabható lekérdezések
A nyelv szemantikus entitásai köré épül
(modul, függvény, változó, rekord, stb.)
XPath-szerű

```
mods.funs[name = open_log].refs
```

```
mods.funs[name = read].dynrefs
```

```
mods.funs.exprs.sub[type=atom, value ~ ".*error"]
```

```
mods.funs.calls5
```

```
mods.funs.calls+
```


Nem-triviális lekérdezések

```
mods[name=mnesia_log].  
  funs[name=open_log].  
    refs[param[index=1].  
origin[type=atom, value=decision_log]]
```

```
mods[name=mnesia_log].  
  funs.exprs.sub[type=tuple,  
    [sub[index=1].  
origin[type=atom, value=backup_args]]]
```


<http://pnyf.inf.elte.hu/trac/refactorerl/wiki/howto>

Design rule-ok ellenőrzése

```
mods.funs[loc>50]  
mods[max_length_of_line>80]  
mods[no_tabs=false]
```

```
mods.funs.is_tail_rec
```

```
mods.funs[exprs.sub[type=catch_expr]]
```

```
mods[name=M].funs[called_by[module/=M]]
```

```
mods[name~"xyz."].funs.calls[not (module~"xyz.*")]
```

```
mods[name~"mnesia.*"].funs.calls[not ((module~"mnesia.*") or (module in [erlang,application,lists]))]
```


Bonyolultság ellenőrzése

num_of_fun
function_calls_in
function_calls_out
max_depth_of_calling
max_depth_of_cases
branches_of_recursion
McCabe

mods.funs[
max_depth_of_calling > 51

Experimental: UCM, MI, Halstead, OPC

Sérülékenységek a lekérdezőben

unsecure_calls

unsecure_os_calls

unsecure_file_operations

unsecure_compile_operations

unstable_calls

unsecure_process_linkage

unsecure_prioritization

unsecure_ets_traversal

unsafe_network

... 16+1 checks in total

Selectors	Short description
<i>unsecure_calls</i>	Lists all the possible vulnerabilities
<i>unsecure_interoperability</i>	Lists interoperability related weaknesses
<i>unsecure_concurrency</i>	Identifies concurrency related issues
<i>unsecure_os_call</i>	Checks for OS injection
<i>unsecure_port_creation</i>	Identifies port creation related issues
<i>unsecure_file_operation</i>	Lists unsecure file handling
<i>unstable_call</i>	Shows possible atom exhaustion
<i>nif_calls</i>	Identifies unsecure NIF calls
<i>unsecure_port_drivers</i>	Lists the unsecure dll usage

mods.funs.unstable_call

Selectors	Short description
<i>decommissioned_crypto</i>	Lists the legacy functions from crypto module
<i>unsecure_compile_operations</i>	Shows unsecure compile/code loading related operations
<i>unsecure_process_linkage</i>	Lists unsecure process linkage
<i>unsecure_prioritization</i>	Identifies unsecure process prioritization
<i>unsecure_ets_traversal</i>	Lists unsecure ETS traversal
<i>unsafe_network</i>	Checks for unsecure kernel related operations
<i>unsecure_xml_usage</i>	Identifies unsecure xml parsing
<i>unsecure_communication</i>	Lists unsecure communication related settings

Biztonságos szoftverfejlesztés támogatása

Motiváció: CodeChecker, Klockwork, SpotBugs, SonarQube, CppCheck, OWASP eszközök

Nem szándékos programozói hibák felderítése

Sérülékenységi szabályok definiálása

Hogyan csinálják?

- Dinamikus megfigyelés
- Statikus ellenőrzés
(manuális keresés, biztonsági tesztelés, code review)
- "By construction"
- Ellenőrző eszközök
(statikus és dinamikusak is)

Erlanghoz...

Együttműködési képességből fakadó sebezhetőségek

Konkurens programozásból eredő hibalehetőségek

Elosztott programozásból adódó sérülékenységek

Injection támadási lehetőségek

Memória túlterhelést eredményező támadási formák

```
init() ->
  {ok, ExtPrg} = io:read("Provide a program..."),
  ok = erlang:load_nif(ExtPrg, 0).
```

NIF and Port related issues

Unstable ets usage
Modified process priority
Process linking related issues

Unsafe kernel usage: allow, start, connect_node
Unsafe SSL, TLS configuration
Outdated crypto calls

```
ssl:connect("example.net", 443, [
  {padding_check, false},
  {beast_mitigation, disabled},
  {fallback, true}
]).
```

Unchecked OS usage
Unsafe file operation
Dynamic code loading

```
parse_xml(Input) ->
  xmerl_sax_parser:stream(Input, []).
```

Dynamic atom creation
XML parsing

```
parse_uri(Input) ->
  http_uri:parse(Input, [{ipv6_host_with_brackets, true}]).
```


Hogyan működik?

Gyanús függvényhívások azonosítása az SPGn

Sérülékeny argumentumok szűrése és ellenőrzése
adattfolyam elemzéssel

Ismeretlen/nem biztonságos források megjelölése

Tisztító/biztonságossá tevő műveletek ellenőrzése és szűrés

Másodlagos elemzések

Optimalizált gráfbejárások

Fals-pozitív szűrés/prioritizálás

Rekurzív végrehajtási utak keresése

Konkurens végrehajtások keresése

“CheckErl”

```
199 set_config(Key5, Value) when is_list(Key5) ->
200   | set_config(list_to_atom(Key5), Value);

193 cache_os_envvar(Var, Keys, Type) ->
194   [ case os:getenv(Key) of
195     | false -> ok;
196     | Value -> set_config(Var, set_config_value(Value, Type))
197   end || Key <- Keys ].
198

176 cache_os_envvars(Vars) ->
177   [ cache_os_envvar(Var)
178   || Var <- Vars ],
179   ok.
```

```
290 foo(D, N) ->
291   case N of
292   | 0 -> ets:safe_fixtable(D, true);
293   | _ -> new_token(D, N)
294   end.
295
296 Add spec
297 new_token(D, Retries) ->
298   case ets:match_object(D, #drain{token=Token, _='__'}) of
299   | [#drain{}] -> new_token(D, Retries-1);
300   | [] -> Token
301   end.
```


Új interfész: DRC Client

Automatizálás, CI-ba illeszthetőség

Könnyen integrálható és használható

Konfiguráció alapján

Futó RefactorÉrl serverhez kapcsolódik

Választható output

```
{
  "blacklist" : ["_build"],
  "rules" : [
    {
      "name" : "longfuns",
      "type" : "sem-query",
      "query" : "@mod.funs[loc>100]",
      "select" : {"dirs" : ["src"]}
    },
    {
      "name" : "styleguidelines",
      "query-set" : "style",
      "select" : {"group" : "@ALL"}
    }
  ],
  "query-sets" : [
    {
      "name" : "style",
      "queries" : [
        {"query" : "macro-naming"},
        {"query" : "state-for-otp-behaviors"},
        {"query" : "no-nested-try-catch"},
        {"query" : "tail-recursive-servers"},
        {"query" : "flush-message-box"},
        {"query" : "tag-messages"},
        {"query" : "no-imports"},
        {"query" : "find-io-format"},
        {"query" : "exported-without-spec"},
        {"query" : "used-underlined-var"},
        {"query" : "exported-functions-limit", "args" : [10]},
        {"query" : "clauses-limit", "args" : [10]},
        {"query" : "module-naming", "args" : ["^[a-z_]*$"]},
        {"query" : "function-naming", "args" : ["^[a-z0-9_]*$"]}
      ]
    }
  ],
  "groups" : []
}
```

```
drc_client:main(["-p PATH
-j PATH/out.json
-r \"style guidelines\"
PATH/config.json"]).
```

Date	All rules	Failed rules	All modules	Failed modules	Failures
{{2021,5,20},{22,49,49}}	1	1	31	4	9

Date	All rules	Failed rules	All modules	Failed modules	Failures
{2021,5,20},{22,49,49}}	1	1	31	4	9

New interface: DRC Client

```
[#{results =>
  #{args => [],query => <<"macro-naming">>,
    results =>
      [#{'module-name' => mnesia,result => ok},
        #{'module-name' => mnesia_lib,
          result =>
            #{nok =>
              [<<"native /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_lib.erl: 7560-7630, 342,1-342,71\n">>]}},
          #{'module-name' => mnesia_monitor,
            result =>
              #{nok =>
                [<<"current_protocol_version /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_monitor.erl: 1817-1858, 86,1-86,42\n">>,
                  <<"previous_protocol_version /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_monitor.erl: 1861-1902, 88,1-88,42\n">>]}},
            #{'module-name' => mnesia_controller,result => ok},
```

Detailed Results: style guidelines

Index	Query String	Arguments	Result
1	function-naming	["^[a-z0-9_]*\$"]	ok
2	module-naming	["^[a-z_]*\$"]	ok
3	macro-naming	[]	error

Custom output

```
{
  "module-name": "mnesia_locker",
  "result": {
    "nok": [
      "dbg /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1320-1341, 59,1-59,22\n",
      "match_oid_held_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1674-1726, 71,1-71,53\n",
      "match_oid_tid_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1794-1846, 73,1-73,53\n",
      "match_oid_sticky_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1934-1981, 75,1-75,48\n",
      "match_oid_lock_queue /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 2076-2171, 77,1-77,96\n"
    ]
  }
},
{
  "module-name": "mnesia_index",
  "result": "ok"
},
```

Query 3

```
mnesia
  ok
mnesia_lib
  native /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_lib.erl: 7560-7630, 342,1-342,71
mnesia_monitor
  current_protocol_version /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_monitor.erl: 1817-1858, 86,1-86,42
  previous_protocol_version /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_monitor.erl: 1861-1902, 88,1-88,42
mnesia_controller
  ok
mnesia_subscr
  ok
mnesia_sup
  ok
mnesia_tm
  ok
mnesia_locker
  dbg /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1320-1341, 59,1-59,22
  match_oid_held_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1674-1726, 71,1-71,53
  match_oid_tid_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1794-1846, 73,1-73,53
  match_oid_sticky_locks /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 1934-1981, 75,1-75,48
  match_oid_lock_queue /usr/local/Cellar/erlang/23.3.4/lib/erlang/lib/mnesia-4.19/src/mnesia_locker.erl: 2076-2171, 77,1-77,96
mnesia_index
  ok
```


DRC Client

Szemantikus lekérdezések
Sérülékenységek
Beépített ellenőrzések

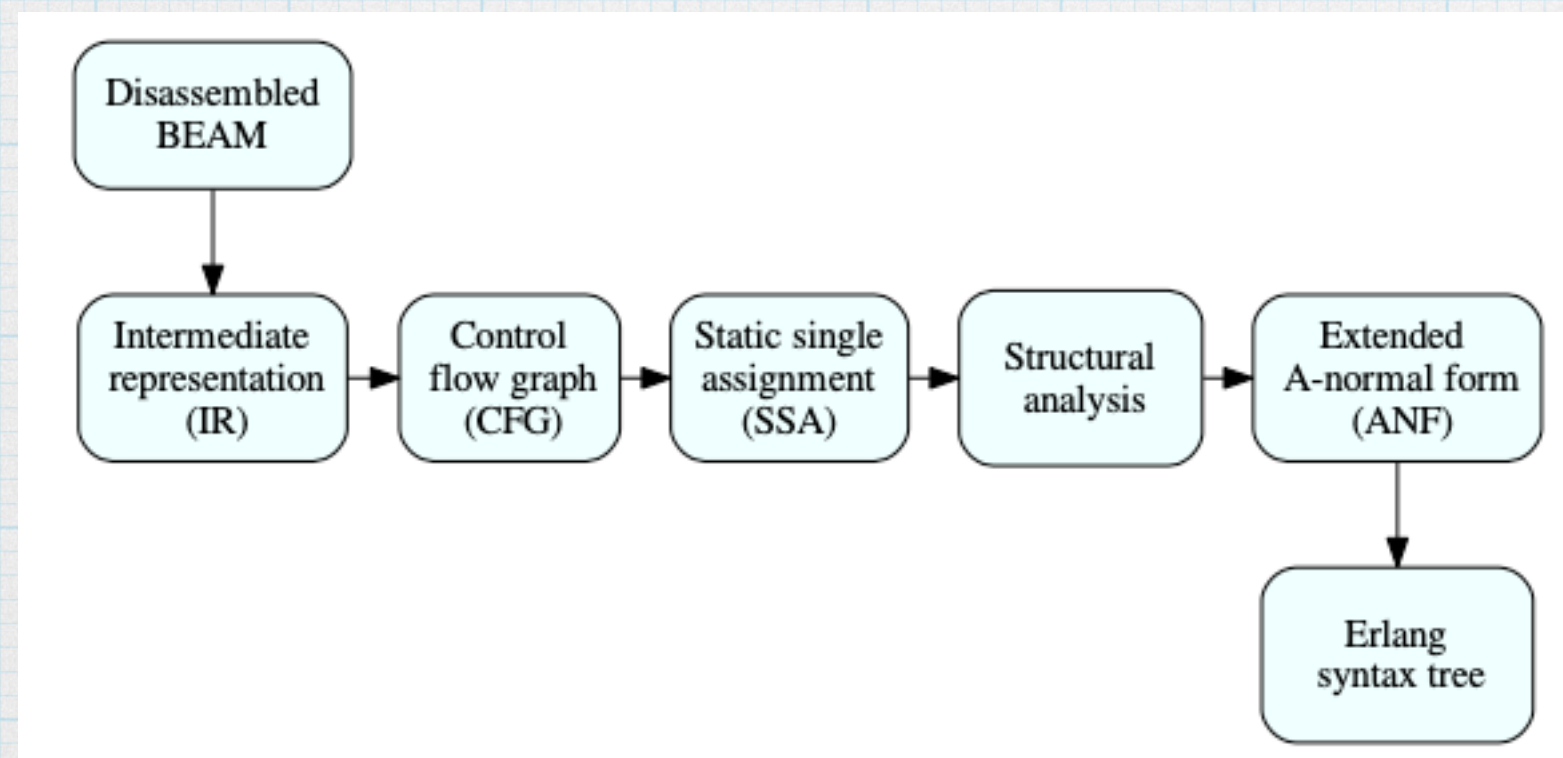
clause-limit
exported-functions-limit
exported-without-spec
used-underlined-var
find-function-call
find-io-format
no-imports
tag-messages
flush-message-box
tail-recursive-servers
macro-naming
no-nested-try-catch
module-naming
function-naming
state-for-otp-behaviours

Kód-visszafejtés

Miért?

Függőségek/sérülékenységek detektálása

Hogyan?



```
3 f() ->
4   M = #{a=>1,b=>3,o=>"4"},
5   #{1:=a,2:=b,3:="c"} = M.
```

```
[{function,f,0,2,
  [{label,1},
   {line,1},
   {func_info,{atom,maps2o},{atom,f},0},
   {label,2},
   {move,{literal,#{a => 1,b => 3,o => "4"}},{x,0}},
   {get_map_elements,{f,3},{x,0},{list,[{atom,a},{x,0}]}}},
   {move,{literal,#{a => 1,b => 3,o => "4"}},{x,1}},
   {get_map_elements,{f,3},{x,1},{list,[{atom,b},{x,1}]}}},
   {move,{literal,#{a => 1,b => 3,o => "4"}},{x,2}},
   {get_map_elements,{f,3},{x,2},{list,[{atom,o},{x,2}]}}},
   {test,is_eq_exact,{f,3},{x,0},{integer,1}}},
   {test,is_eq_exact,{f,3},{x,1},{integer,3}}},
   {test,is_eq_exact,{f,3},{x,2},{literal,"4"}}},
   {move,{literal,#{a => 1,b => 3,o => "4"}},{x,0}},
   return,
  {label,3},
  {line,2},
  {badmatch,{literal,#{a => 1,b => 3,o => "4"}}}],
```

```
5 f() ->
6   X0_1 = #{a=>1, b=>3, o=>"4"},
7   X100_1 = is_map_key(1, X0_1),
8   if
9     (((((X100_1 == true andalso is_map_key(2, #{a=>1, b=>3, o=>"4"}) ==
10      true) andalso is_map_key(3, #{a=>1, b=>3, o=>"4"}) == true) andalso
11      map_get(1, X0_1) == a) andalso map_get(2, #{a=>1, b=>3, o=>"4"}) ==
12      b) andalso map_get(3, #{a=>1, b=>3, o=>"4"}) == "c") ->
13     X0_3 = #{a=>1, b=>3, o=>"4"},
14     X0_3;
15     true -> throw({decompiled_exception, badmatch})
16   end.
```


Futási idejű hibák forrásának felderítése

Vezérlés-folyam gráf alapú végrehajtási út alapján

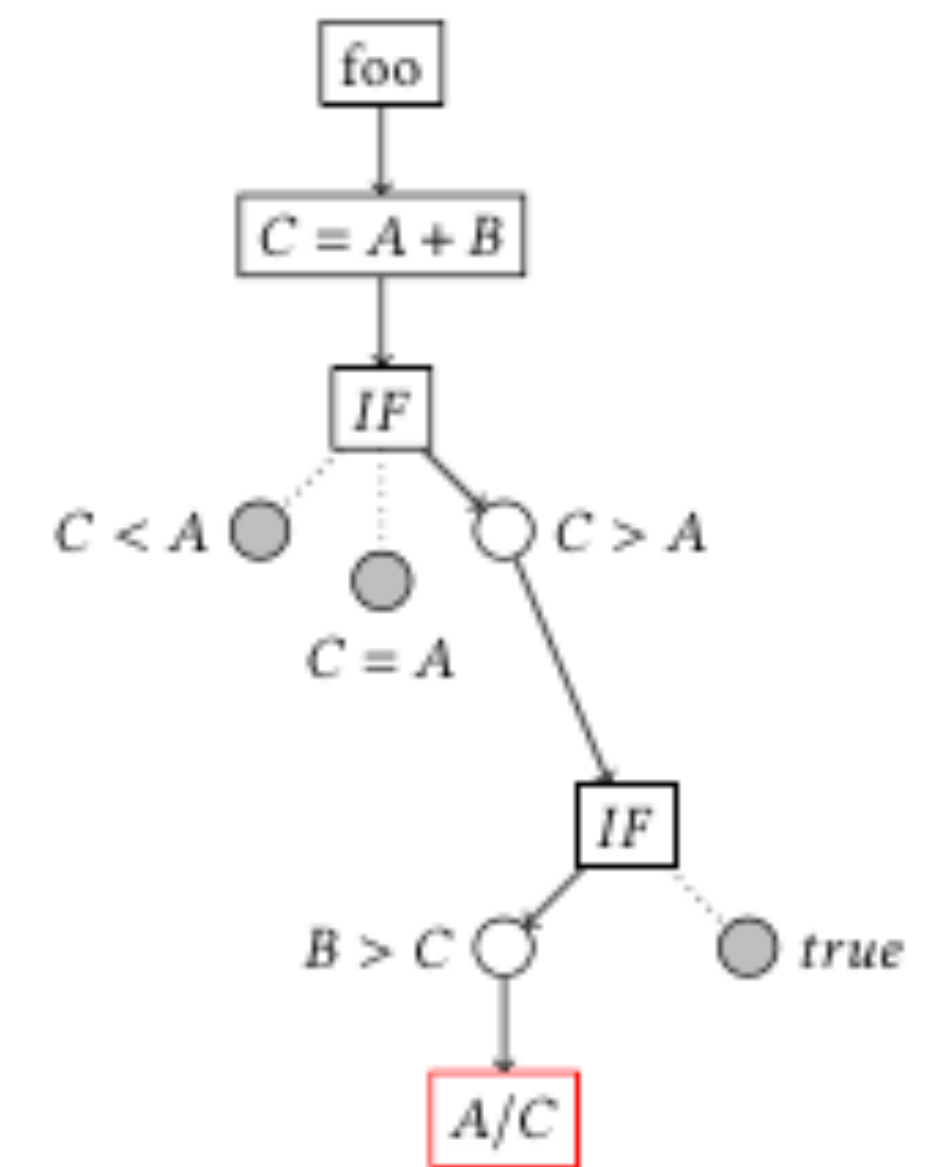
Célzott szimbolikus végrehajtás a megszorítások kinyerésére

Z3 használat

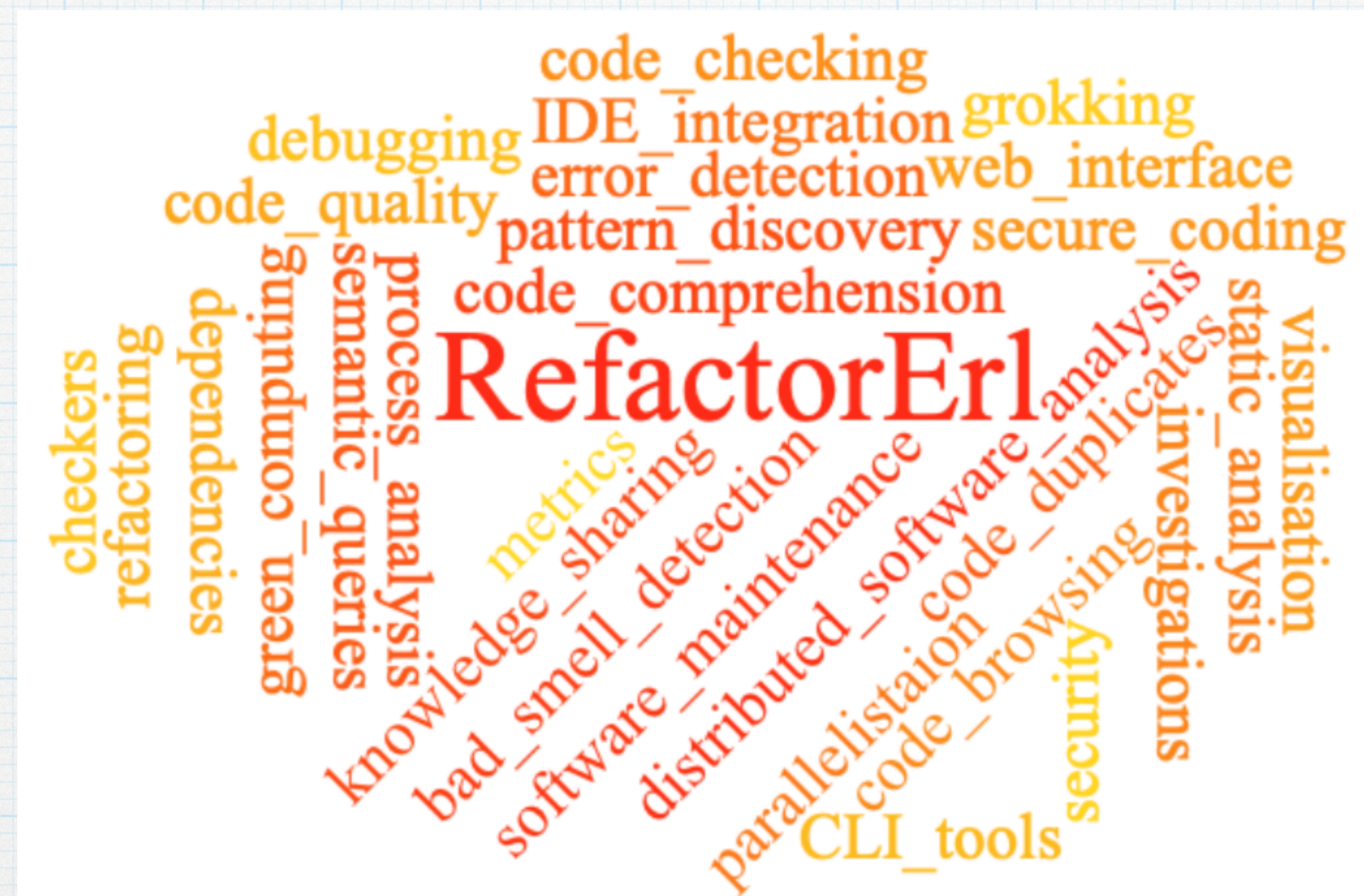
A futási idejű hiba figyelembe vétele

Output: végrehajtási út, mely a hiba előfordulásához vezet

```
1 -module(example1).  
2 -export([foo/2]).  
3 foo(A, B) ->  
4   C = A + B,  
5   if  
6     C < A -> 0;  
7     C == A -> 1;  
8     C > A ->  
9       if  
10        B > C -> A / C;  
11        true -> B / C  
12      end  
13 end.
```



```
5> example1:foo(A,B).  
** exception error: an error occurred when evaluating an arithmetic expression  
    in function example1:foo/2 (example1.erl, line 10)
```

plc.inf.elte.hu/erlang

A legfrissebb K+F eredmények hamarosan belekerülnek a publikus releasbe is



KÖSZÖNÖM A FIGYELMET!

Az Alkalmazásiterület-specifikus nagy megbízhatóságú informatikai megoldások című projekt a Nemzeti Kutatási Fejlesztési és Innovációs Alapból biztosított támogatással, a Tématerületi kiválósági program (TKP2020-NKA-06, Nemzeti Kihívások Alprogram) finanszírozásában valósult meg.



NATIONAL RESEARCH, DEVELOPMENT
AND INNOVATION OFFICE
HUNGARY

PROGRAM
FINANCED FROM
THE NRDI FUND