EPAX

0.01

Generated by Doxygen 1.7.6.1

Fri Feb 7 2014 09:40:33

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Namespace Documentation

3.1 EPAX Namespace Reference

Typedefs

- typedef Binary ∗ BIN
- typedef Section ∗ SECT
- typedef Function ∗ FUNC
- typedef ControlFlow ∗ CFG
- typedef Loop ∗ LOOP
- typedef BasicBlock ∗ BBL
- typedef Instruction ∗ INSN
- typedef Symbol ∗ SYM
- typedef FlowEquation ∗ FLOW

Functions

- BIN BIN_create (std::string fileName)
- std::string BIN_getName (BIN bin)
- void BIN_destroy (BIN bin)
- void BIN_run (BIN bin, int argc, char ∗∗argv)
- FUNC BIN_firstFunc (BIN bin)
- FUNC BIN_nextFunc (BIN bin, FUNC func)
- bool BIN_isLastFunc (BIN bin, FUNC func)
- uint32_t BIN_countFunc (BIN bin)
- bool BIN_isExecutable (BIN bin)
- uint32_t BIN_fileSize (BIN bin)
- void BIN_printStaticFile (BIN bin, std::string fname)
- FUNC BIN_findFunc (BIN bin, uint64_t addr)
- FUNC FUNC_create (uint8_t ∗bytes, uint32_t size)
- void FUNC_Destroy (FUNC func)

- void FUNC_print (FUNC func)
- std::string FUNC_name (FUNC func)
- uint32_t FUNC_size (FUNC func)
- uint64_t FUNC_addr (FUNC func)
- std::string FUNC_secName (FUNC func)
- BIN FUNC_bin (FUNC func)
- uint32_t FUNC_countBbl (FUNC func)
- BBL FUNC_findBbl (FUNC func, uint64_t addr)
- BBL FUNC_firstBbl (FUNC func)
- BBL FUNC_nextBbl (FUNC func, BBL bbl)
- bool FUNC_isLastBbl (FUNC func, BBL bbl)
- uint32_t FUNC_countInsn (FUNC func)
- INSN FUNC_findInsn (FUNC func, uint64_t addr)
- INSN FUNC_firstInsn (FUNC func)
- INSN FUNC_nextInsn (FUNC func, INSN insn)
- bool FUNC_isLastInsn (FUNC func, INSN insn)
- CFG FUNC_cfg (FUNC func)
- uint32_t FUNC_countTargets (FUNC func)
- uint32_t FUNC_targets (FUNC func, std::vector< FUNC > &funcList)
- uint32_t CFG_countLoop (CFG cfg)
- LOOP CFG_findLoop (CFG cfg, uint64_t addr)
- LOOP CFG_firstLoop (CFG cfg)
- LOOP CFG_nextLoop (CFG cfg, LOOP loop)
- bool CFG_isLastLoop (CFG cfg, LOOP loop)
- CFG LOOP_cfg (LOOP loop)
- FUNC LOOP_func (LOOP loop)
- uint32_t LOOP_size (LOOP loop)
- uint32_t LOOP_countBbl (LOOP loop)
- BBL LOOP_findBbl (LOOP loop, uint64_t addr)
- BBL LOOP_firstBbl (LOOP loop)
- BBL LOOP_nextBbl (LOOP loop, BBL bbl)
- bool LOOP_isLastBbl (LOOP loop, BBL bbl)
- uint32_t LOOP_countInsn (LOOP loop)
- INSN LOOP_findInsn (LOOP loop, uint64_t addr)
- INSN LOOP_firstInsn (LOOP loop)
- INSN LOOP_nextInsn (LOOP loop, INSN insn)
- bool LOOP_isLastInsn (LOOP loop, INSN insn)
- BBL LOOP_head (LOOP loop)
- BBL LOOP_tail (LOOP loop)
- uint32_t LOOP_countExits (LOOP loop)
- uint32_t LOOP_exits (LOOP loop, std::vector< INSN > &insnList)
- bool LOOP_isInnerLoop (LOOP loop1, LOOP loop2)
- LOOP LOOP_parent (LOOP loop)
- uint32_t LOOP_index (LOOP loop)
- uint32_t LOOP_depth (LOOP loop)
- bool BBL_isHead (BBL bbl, INSN insn)

- bool BBL_isTail (BBL bbl, INSN insn)
- INSN BBL_head (BBL bbl)
- INSN BBL_tail (BBL bbl)
- FUNC BBL_func (BBL bbl)
- LOOP BBL_loop (BBL bbl)
- uint32_t BBL_size (BBL bbl)
- uint64_t BBL_addr (BBL bbl)
- uint32_t BBL_countInsn (BBL bbl)
- INSN BBL_findInsn (BBL bbl, uint64_t addr)
- INSN BBL_firstInsn (BBL bbl)
- INSN BBL_nextInsn (BBL bbl, INSN insn)
- bool BBL_isLastInsn (BBL bbl, INSN insn)
- uint32_t BBL_countTargets (BBL bbl)
- uint32_t BBL_targets (BBL bbl, std::vector< BBL > &bblList)
- bool BBL_hasFallthroughTarget (BBL bbl)
- BBL BBL_fallthroughTarget (BBL bbl)
- uint32_t BBL_countJumpTargets (BBL bbl)
- uint32_t BBL_jumpTargets (BBL bbl, std::vector< BBL > &bblList)
- uint32_t BBL_countSources (BBL bbl)
- uint32_t BBL_sources (BBL bbl, std::vector< BBL > &bblList)
- uint32_t INSN_targets (INSN insn, std::vector< uint64_t > &tlist)
- BBL INSN_bbl (INSN insn)
- FUNC INSN_func (INSN insn)
- LOOP INSN_loop (INSN insn)
- uint64_t INSN_addr (INSN insn)
- std::string INSN_string (INSN insn)
- uint64_t INSN_callTarget (INSN insn)
- bool INSN_isBranch (INSN insn)
- bool INSN_isFpop (INSN insn)
- bool INSN_isMemop (INSN insn)
- uint32_t INSN_size (INSN insn)
- std::string INSN_condName (INSN insn)
- bool INSN_fallsThrough (INSN insn)
- uint32_t INSN_sourceRegisterSizeInBits (INSN insn)
- uint32_t INSN_sourceDatatypeSizeInBits (INSN insn)

3.1.1 Typedef Documentation

3.1.1.1 typedef BasicBlock∗ EPAX::BBL

Definition at line 52 of file Interface.hpp.

3.1.1.2 typedef Binary∗ EPAX::BIN

Definition at line 43 of file Interface.hpp.

3.1.1.3 typedef ControlFlow∗ EPAX::CFG

Definition at line 50 of file Interface.hpp.

3.1.1.4 typedef FlowEquation∗ EPAX::FLOW

Definition at line 55 of file Interface.hpp.

3.1.1.5 typedef Function∗ EPAX::FUNC

Definition at line 49 of file Interface.hpp.

3.1.1.6 typedef Instruction∗ EPAX::INSN

Definition at line 53 of file Interface.hpp.

3.1.1.7 typedef Loop∗ EPAX::LOOP

Definition at line 51 of file Interface.hpp.

3.1.1.8 typedef Section∗ EPAX::SECT

Definition at line 48 of file Interface.hpp.

3.1.1.9 typedef Symbol∗ EPAX::SYM

Definition at line 54 of file Interface.hpp.

3.1.2 Function Documentation

3.1.2.1 uint64_t EPAX::BBL_addr (BBL *bbl*)

3.1.2.2 uint32_t EPAX::BBL_countInsn (BBL *bbl*)

Get the number of INSNs in a BBL

Parameters

bbl	a BBL object

Returns

the number of INSNs in a BBL

3.1.2.3 uint32_t EPAX::BBL_countJumpTargets (BBL *bbl*)

Counts the number of non-fallthrough targets for a BBL

Parameters

bbl	a BBL object

Returns

the number of targets for bbl that are not fallthrough targets

3.1.2.4 uint32_t EPAX::BBL_countSources (BBL *bbl*)

Counts the number of control source blocks for a BBL

Parameters

bbl	a BBL object

Returns

the number of control source blocks for bbl

3.1.2.5 uint32_t EPAX::BBL_countTargets (BBL *bbl*)

Gets the number of control flow targets for a BBL

Parameters

bbl	a BBL object

Returns

the number of BBLs that are control flow targets for bbl

3.1.2.6 BBL EPAX::BBL_fallthroughTarget (BBL *bbl*)

Gets the fallthrough target for a BBL

Parameters

bbl	a BBL object

Returns

the BBL that is the fallthrough target of bbl, or NULL if no such BBL exists

3.1.2.7 INSN EPAX::BBL_findInsn (BBL *bbl*, uint64_t *addr*)

Find the INSN within a BBL at a given address

Parameters

bbl	a BBL object
addr	a virtual address

Returns

the INSN within BBL that intersects with addr, or NULL if no such INSN exists

3.1.2.8 INSN EPAX::BBL_firstInsn (BBL *bbl*)

Get the first INSN object in a BBL

Parameters

bbl	a BBL object

Returns

the first INSN in bbl

3.1.2.9 FUNC EPAX::BBL_func (BBL *bbl*)

Get the function containing a BBL

Parameters

bbl	a BBL object

Returns

the FUNC containing bbl

3.1.2.10 bool EPAX::BBL_hasFallthroughTarget (BBL *bbl*)

Tells whether control can fall through the end of a BBL

Parameters

bbl	a BBL object

Returns

true iff control can fall through the end of bbl

3.1.2.11 INSN EPAX::BBL_head (BBL *bbl*)

Get the head INSN of a BBL

Parameters

bbl	a BBL object

Returns

the head INSN of bbl

3.1.2.12 bool EPAX::BBL_isHead (BBL *bbl,* INSN *insn*)

Is an insn the head of a BBL

Parameters

bbl	a BBL object
insn	an INSN object

Returns

true iff insn is the head of bbl

3.1.2.13 bool EPAX::BBL_isLastInsn (BBL *bbl,* INSN *insn*)

Tests whether a INSN is the last in a BBL

Parameters

bbl	a BBL object
insn	a INSN object

Returns

true iff insn is the last INSN object in bbl, false otherwise

3.1.2.14 bool EPAX::BBL_isTail (BBL *bbl*, INSN *insn*)

Is an insn the tail of a BBL

Parameters

bbl	a BBL object
insn	an INSN object

Returns

true iff insn is the tail of bbl

3.1.2.15 uint32_t EPAX::BBL_jumpTargets (BBL *bbl*, std::vector< BBL > & *bblList*)

Gets the non-fallthrough targets for a BBL

Parameters

bbl	a BBL object
(out)	the non-fallthrough targets for bbl

Returns

the number of non-fallthrough targets for bbl

3.1.2.16 LOOP EPAX::BBL_loop (BBL *bbl*)

Get the loop containing a BBL

Parameters

bbl	a BBL object

Returns

the LOOP containing bbl, of NULL if no such LOOP exists

3.1.2.17 INSN EPAX::BBL_nextInsn (BBL *bbl*, INSN *insn*)

Get the next INSN object in a BBL

Parameters

bbl	a BBL object
insn	a INSN object

Returns

the INSN from bbl that is subsequent to insn, or NULL if no such INSN exists

3.1.2.18 uint32_t EPAX::BBL_size (BBL *bbl*)

Get the size of a BBL

Parameters

bbl	a BBL object

Returns

the size in bytes of bbl

3.1.2.19 uint32_t EPAX::BBL_sources (BBL *bbl,* std::vector< BBL > & *bblList*)

Gets the control source blocks for a BBL

Parameters

bbl	a BBL object
(out)	bblList the control source blocks for bbl

Returns

the number of control source blocks for bbl

3.1.2.20 INSN EPAX::BBL_tail (BBL *bbl*)

Get the tail INSN of a BBL

Parameters

bbl	a BBL object

Returns

the tail INSN of bbl

3.1.2.21 uint32_t EPAX::BBL_targets (BBL *bbl,* std::vector< BBL > & *bblList*)

Gets the control flow targets for a BBL

Parameters

bbl	a BBL object
(out)	the BBLs that are control flow targets for bbl

Returns

the number of BBLs that are control targets for bbl

3.1.2.22 uint32_t EPAX::BIN_countFunc (BIN *bin*)

Count the functions in a BIN

Parameters

bin	a BIN

Returns

the number of FUNCs in bin

3.1.2.23 BIN EPAX::BIN_create (std::string *fileName*)

Creates a BIN object

Parameters

fileName	The name of a binary file. Allowed formats are: ELF, MachO

Returns

a BIN object created using the input parameter

3.1.2.24 void EPAX::BIN_destroy (BIN *bin*)

frees all memory associated with a BIN object

Parameters

bin	a BIN object, which is set to NULL during this operation.

Returns

none

3.1.2.25 uint32_t EPAX::BIN_fileSize (BIN *bin*)

Find the file size of a BIN

Parameters

bin	a BIN

Returns

the size of the file used to create bin

3.1.2.26 FUNC EPAX::BIN_findFunc (BIN *bin,* uint64_t *addr*)

Find the function at a given virtual address

Parameters

bin	a BIN
addr	a virtual address

Returns

the FUNC at addr in bin

3.1.2.27 FUNC EPAX::BIN_firstFunc (BIN *bin*)

Gets the first function in a BIN object

Parameters

bin	a BIN object

Returns

the first logical function in binary

3.1.2.28 std::string EPAX::BIN_getName (BIN *bin*)

returns the name of a BIN object

Parameters

bin	a BIN

Returns

the name of the file used to create bin

3.1.2.29 bool EPAX::BIN_isExecutable (BIN *bin*)

Is the BIN executable

Parameters

bin	a BIN

Returns

true iff bin is an executable file

3.1.2.30 bool EPAX::BIN_isLastFunc (BIN *bin*, FUNC *func*)

Is a FUNC the last logical function in its BIN

Parameters

bin	a BIN
func	a FUNC from bin

Returns

true iff func is the last logical function in bin

3.1.2.31 FUNC EPAX::BIN_nextFunc (BIN *bin*, FUNC *func*)

Gets the next logical function in a BIN object

Parameters

bin	a BIN object
func	a FUNC from binary

Returns

the logical function following func from bin, or NULL if func is the last such function

3.1.2.32 void EPAX::BIN_printStaticFile (BIN *bin*, std::string *fname*)

Print a static file containing detailed information about the structures found in a BIN

Parameters

bin	a BIN
fname	the name of the output file to catch static analysis

Returns

none

3.1.2.33 void EPAX::BIN_run (BIN *bin*, int *argc*, char ** *argv*)

Runs a the program represented by BIN with arguments; does not return.

Parameters

bin	a BIN object for which BIN_isExecutable returns true
argc	the number of program arguments
argv	the program arguments

Returns

none

3.1.2.34 uint32_t EPAX::CFG_countLoop (CFG *cfg*)

Count the number of loops in a CFG

Parameters

cfg	a CFG object

Returns

the number of loops in cfg

3.1.2.35 LOOP EPAX::CFG_findLoop (CFG *cfg*, uint64_t *addr*)

Find the LOOP within a CFG ad a given address

Parameters

cfg	a CFG object
addr	a virtual address

Returns

the loop within cfg at addr, or NULL if no such loop exists

3.1.2.36 LOOP EPAX::CFG_firstLoop (CFG *cfg*)

Get the first loop in a CFG

Parameters

cfg	a CFG object

Returns

the first loop in cfg

3.1.2.37 bool EPAX::CFG_isLastLoop (CFG *cfg,* LOOP *loop*)

Tests whether a LOOP is the last in a CFG

Parameters

cfg	a CFG object
loop	a LOOP object

Returns

true iff loop is the last LOOP in cfg

3.1.2.38 LOOP EPAX::CFG_nextLoop (CFG *cfg,* LOOP *loop*)

Get the next loop in a CFG

Parameters

cfg	a CFG object
loop	a LOOP object

Returns

the successor to loop within cfg, or NULL if no such LOOP exists

3.1.2.39 uint64_t EPAX::FUNC_addr (FUNC *func*)

Get the virtual address of a FUNC

Parameters

func	a FUNC object

Returns

the virtual address of func

3.1.2.40 BIN EPAX::FUNC_bin (FUNC *func*)

Get the BIN object that contains a FUNC

Parameters

func	a FUNC object

Returns

the BIN object associated with func

3.1.2.41 CFG EPAX::FUNC_cfg (FUNC *func*)

Get the CFG attached to a FUNC

Parameters

func	a FUNC object

Returns

the CFG attached to func

3.1.2.42 uint32_t EPAX::FUNC_countBbl (FUNC *func*)

Get the number of BBL objects in a FUNC

Parameters

func	a FUNC object

Returns

the number of BBL objects in func

3.1.2.43 uint32_t **EPAX::FUNC_countInsn** (FUNC *func*)

Get the number of INSNs in a FUNC

Parameters

func	a FUNC object

Returns

the number of INSNs in a FUNC

3.1.2.44 uint32_t **EPAX::FUNC_countTargets** (FUNC *func*)

Get the number of targets of (functions called by) a FUNC

Parameters

func	a FUNC object

Returns

the number of unique targets of the text of func

3.1.2.45 FUNC **EPAX::FUNC_create** (uint8_t ∗ *bytes*, uint32_t *size*)

Generate a function using the supplied bytes. Note that the size of the function found may be smaller than the size of the input buffer supplied. Use FUNC_size on the returned FUNC to find its size.

Parameters

bytes	a buffer of raw instruction bytes
size	the size of the buffer

Returns

a FUNC generated using the bytes supplied in buf

3.1.2.46 void **EPAX::FUNC_Destroy** (FUNC *func*)

Destroy a function; note that it is an error to destroy a function that was not created with FUNC_create

Parameters

func	a FUNC object that was created with FUNC_Create

Returns

none

3.1.2.47 BBL EPAX::FUNC_findBbl (FUNC *func*, uint64_t *addr*)

Find the BBL within a FUNC at a given address

Parameters

func	a FUNC object
addr	a virtual address

Returns

the BBL within FUNC that intersects with addr, or NULL if no such BBL exists

3.1.2.48 INSN EPAX::FUNC_findInsn (FUNC *func*, uint64_t *addr*)

Find the INSN within a FUNC at a given address

Parameters

func	a FUNC object
addr	a virtual address

Returns

the INSN within FUNC that intersects with addr, or NULL if no such INSN exists

3.1.2.49 BBL EPAX::FUNC_firstBbl (FUNC *func*)

Get the first BBL object in a FUNC

Parameters

func	a FUNC object

Returns

the first BBL in func

3.1.2.50 INSN EPAX::FUNC_firstInsn (FUNC *func*)

Get the first INSN object in a FUNC

Parameters

func	a FUNC object

Returns

the first INSN in func

3.1.2.51 bool EPAX::FUNC_isLastBbl (FUNC *func,* BBL *bbl*)

Tests whether a BBL is the last in a FUNC

Parameters

func	a FUNC object
bbl	a BBL object

Returns

true iff bbl is the last BBL object in func, false otherwise

3.1.2.52 bool EPAX::FUNC_isLastInsn (FUNC *func,* INSN *insn*)

Tests whether a INSN is the last in a FUNC

Parameters

func	a FUNC object
insn	a INSN object

Returns

true iff insn is the last INSN object in func, false otherwise

3.1.2.53 std::string EPAX::FUNC_name (FUNC *func*)

Get the name of a FUNC

Parameters

func	a FUNC object

Returns

the name of func, or NULL if no name can be found

3.1.2.54 BBL EPAX::FUNC_nextBbl (FUNC *func,* BBL *bbl*)

Get the next BBL object in a FUNC

Parameters

func	a FUNC object
bbl	a BBL object

Returns

the BBL from func that is subsequent to bbl, or NULL if no such BBL exists

3.1.2.55 INSN EPAX::FUNC_nextInsn (FUNC *func,* INSN *insn*)

Get the next INSN object in a FUNC

Parameters

func	a FUNC object
insn	a INSN object

Returns

the INSN from func that is subsequent to insn, or NULL if no such INSN exists

3.1.2.56 void EPAX::FUNC_print (FUNC *func*)

Print a FUNC

Parameters

func	a FUNC object

Returns

none

3.1.2.57 std::string EPAX::FUNC_secName (FUNC *func*)

Get the name of the section that contains a FUNC

Parameters

func	a FUNC object

Returns

the name of the section containing func, or NULL if it is unknown

3.1.2.58 uint32_t **EPAX::FUNC_size** (FUNC *func*)

Get the size of a FUNC

Parameters

func	a FUNC object

Returns

the size of func in bytes

3.1.2.59 uint32_t **EPAX::FUNC_targets** (FUNC *func,* std::vector< FUNC > & *funcList*)

Get the unique targets of (functions called by) a FUNC

Parameters

func	a FUNC object
(out)	funcList the unique targets of func

Returns

the number of unique targets of func

3.1.2.60 uint64_t **EPAX::INSN_addr** (INSN *insn*)

Get the virtual address of an INSN

Parameters

insn	an INSN object

Returns

the virtual address of insn

3.1.2.61 BBL EPAX::INSN_bbl (INSN *insn*)

Get the basic block of an INSN

Parameters

insn	an INSN object

Returns

the BBL that contains insn, or NULL if no such BBL exists

3.1.2.62 uint64_t EPAX::INSN_callTarget (INSN *insn*)

Get the call target of an INSN

Parameters

insn	an INSN object

Returns

the address of the call target of insn, or 0 if the target cannot be found

3.1.2.63 std::string EPAX::INSN_condName (INSN *insn*)

Get the string rep of the predicate condition of an INSN

Parameters

insn	an INSN object

Returns

the string representation of the predicate condition of insn

3.1.2.64 bool EPAX::INSN_fallsThrough (INSN *insn*)

Can control fall through an INSN

Parameters

insn	an INSN object

Returns

true iff control can fall through insn

3.1.2.65 FUNC EPAX::INSN_func (INSN *insn*)

Get the function of an INSN

Parameters

insn	an INSN object

Returns

the FUNC that contains insn, or NULL if no such FUNC exists

3.1.2.66 bool EPAX::INSN_isBranch (INSN *insn*)

Is an INSN a branch

Parameters

insn	an INSN object

Returns

true iff insn is a branch instruction of any kind

3.1.2.67 bool EPAX::INSN_isFpop (INSN *insn*)

Is an INSN an fp op

Parameters

insn	an INSN object

Returns

true iff either source or destination operands is fp data

3.1.2.68 bool EPAX::INSN_isMemop (INSN *insn*)

Is an INSN a mem op

Parameters

insn	an INSN object

Returns

true iff the insns touches memory

3.1.2.69 LOOP EPAX::INSN_loop (INSN *insn*)

Get the loop of an INSN

Parameters

insn	an INSN object

Returns

the LOOP that contains insn, or NULL if no such LOOP exists

3.1.2.70 uint32_t EPAX::INSN_size (INSN *insn*)

Get the size in of an INSN in bytes

Parameters

insn	an INSN object

Returns

the size (in bytes) of insn

3.1.2.71 uint32_t EPAX::INSN_sourceDatatypeSizeInBits (INSN *insn*)

Size of source datatype in bits

Parameters

insn	an INSN object

Returns

the number of bits in a source operand

3.1.2.72 uint32_t **EPAX::INSN_sourceRegisterSizeInBits** (INSN *insn*)

Size of a source register in bits

Parameters

insn	an INSN object

Returns

the number of bits in a source register

3.1.2.73 std::string **EPAX::INSN_string** (INSN *insn*)

Get a string representation of an INSN

Parameters

insn	an INSN object

Returns

the decoded string representation of insn

3.1.2.74 uint32_t **EPAX::INSN_targets** (INSN *insn,* std::vector< uint64_t > & *tlist*)

Get the control target INSNs for an INSN

Parameters

insn	an INSN object
tlist	(out) the target INSNs of insn

Returns

the number of control targets of insn

3.1.2.75 CFG **EPAX::LOOP_cfg** (LOOP *loop*)

Get the CFG associated with a LOOP

Parameters

loop	a LOOP object

Returns

the CFG associated with loop

3.1.2.76 uint32_t EPAX::LOOP_countBbl (LOOP *loop*)

Get the number of BBL objects in a LOOP

Parameters

loop	a LOOP object

Returns

the number of BBL objects in loop

3.1.2.77 uint32_t EPAX::LOOP_countExits (LOOP *loop*)

Get the number of exit points from a LOOP

Parameters

loop	a LOOP object

Returns

the number of exit points in loop

3.1.2.78 uint32_t EPAX::LOOP_countInsn (LOOP *loop*)

Get the number of INSNs in a LOOP

Parameters

loop	a LOOP object

Returns

the number of INSNs in a LOOP

3.1.2.79 uint32_t EPAX::LOOP_depth (LOOP *loop*)

Get the depth of a LOOP

Parameters

loop	a LOOP object

Returns

the depth of loop

3.1.2.80 uint32_t EPAX::LOOP_exits (LOOP *loop,* std::vector< INSN > & *insnList*)

Get the instructions that are exit points from a particular LOOP

Parameters

loop	a LOOP object
(out)	insnList loop's exit points

Returns

the number of exit points in loop

3.1.2.81 BBL EPAX::LOOP_findBbl (LOOP *loop,* uint64_t *addr*)

Find the BBL within a LOOP at a given address

Parameters

loop	a LOOP object
addr	a virtual address

Returns

the BBL within LOOP that intersects with addr, or NULL if no such BBL exists

3.1.2.82 INSN EPAX::LOOP_findInsn (LOOP *loop,* uint64_t *addr*)

Find the INSN within a LOOP at a given address

Parameters

loop	a LOOP object
addr	a virtual address

Returns

the INSN within LOOP that intersects with addr, or NULL if no such INSN exists

3.1.2.83 BBL EPAX::LOOP_firstBbl (LOOP *loop*)

Get the first BBL object in a LOOP

Parameters

loop	a LOOP object

Returns

the first BBL in loop

3.1.2.84 INSN EPAX::LOOP_firstInsn (LOOP *loop*)

Get the first INSN object in a LOOP

Parameters

loop	a LOOP object

Returns

the first INSN in loop

3.1.2.85 FUNC EPAX::LOOP_func (LOOP *loop*)

Get the FUNC associated with a LOOP

Parameters

loop	a LOOP object

Returns

the FUNC associated with loop

3.1.2.86 BBL EPAX::LOOP_head (LOOP *loop*)

Get the head basic block from a LOOP

Parameters

loop	a LOOP object

Returns

the head (target of the back edge) BBL in loop

3.1.2.87 uint32_t EPAX::LOOP_index (LOOP *loop*)

Get the index of a LOOP

Parameters

loop	a LOOP object

Returns

the index of loop, which is unique within the containing FUNC/CFG

3.1.2.88 bool EPAX::LOOP_isInnerLoop (LOOP *loop1*, LOOP *loop2*)

Find out whether a LOOP is an inner loop of another LOOP

Parameters

loop	a LOOP object
other	a LOOP object

Returns

true iff loop2 is an inner loop of loop1

3.1.2.89 bool EPAX::LOOP_isLastBbl (LOOP *loop*, BBL *bbl*)

Tests whether a BBL is the last in a LOOP

Parameters

loop	a LOOP object
bbl	a BBL object

Returns

true iff bbl is the last BBL object in loop, false otherwise

3.1.2.90 bool EPAX::LOOP_isLastInsn (LOOP *loop,* INSN *insn*)

Tests whether a INSN is the last in a LOOP

Parameters

loop	a LOOP object
insn	a INSN object

Returns

true iff insn is the last INSN object in loop, false otherwise

3.1.2.91 BBL EPAX::LOOP_nextBbl (LOOP *loop,* BBL *bbl*)

Get the next BBL object in a LOOP

Parameters

loop	a LOOP object
bbl	a BBL object

Returns

the BBL from loop that is subsequent to bbl, or NULL if no such BBL exists

3.1.2.92 INSN EPAX::LOOP_nextInsn (LOOP *loop,* INSN *insn*)

Get the next INSN object in a LOOP

Parameters

loop	a LOOP object
insn	a INSN object

Returns

the INSN from loop that is subsequent to insn, or NULL if no such INSN exists

3.1.2.93 LOOP EPAX::LOOP_parent (LOOP *loop*)

Get the parent LOOP of a LOOP

Parameters

loop	a LOOP object

Returns

the parent LOOP of loop, or NULL no such loop exists

3.1.2.94 uint32_t EPAX::LOOP_size (LOOP *loop*)

Get the size of a LOOP

Parameters

loop	a LOOP object

Returns

the size in bytes of loop

3.1.2.95 BBL EPAX::LOOP_tail (LOOP *loop*)

Get the tail basic block from a LOOP

Parameters

loop	a LOOP object

Returns

the tail (source of the back edge) BBL in loop

Chapter 4

File Documentation

4.1 Interface.hpp File Reference

`#include <stdint.h> #include <string> #include <vector>`

Namespaces

- namespace EPAX

Typedefs

- typedef Binary * EPAX::BIN
- typedef Section * EPAX::SECT
- typedef Function * EPAX::FUNC
- typedef ControlFlow * EPAX::CFG
- typedef Loop * EPAX::LOOP
- typedef BasicBlock * EPAX::BBL
- typedef Instruction * EPAX::INSN
- typedef Symbol * EPAX::SYM
- typedef FlowEquation * EPAX::FLOW

Functions

- BIN EPAX::BIN_create (std::string fileName)
- std::string EPAX::BIN_getName (BIN bin)
- void EPAX::BIN_destroy (BIN bin)
- void EPAX::BIN_run (BIN bin, int argc, char **argv)
- FUNC EPAX::BIN_firstFunc (BIN bin)
- FUNC EPAX::BIN_nextFunc (BIN bin, FUNC func)
- bool EPAX::BIN_isLastFunc (BIN bin, FUNC func)

- uint32_t EPAX::BIN_countFunc (BIN bin)
- bool EPAX::BIN_isExecutable (BIN bin)
- uint32_t EPAX::BIN_fileSize (BIN bin)
- void EPAX::BIN_printStaticFile (BIN bin, std::string fname)
- FUNC EPAX::BIN_findFunc (BIN bin, uint64_t addr)
- FUNC EPAX::FUNC_create (uint8_t ∗bytes, uint32_t size)
- void EPAX::FUNC_Destroy (FUNC func)
- void EPAX::FUNC_print (FUNC func)
- std::string EPAX::FUNC_name (FUNC func)
- uint32_t EPAX::FUNC_size (FUNC func)
- uint64_t EPAX::FUNC_addr (FUNC func)
- std::string EPAX::FUNC_secName (FUNC func)
- BIN EPAX::FUNC_bin (FUNC func)
- uint32_t EPAX::FUNC_countBbl (FUNC func)
- BBL EPAX::FUNC_findBbl (FUNC func, uint64_t addr)
- BBL EPAX::FUNC_firstBbl (FUNC func)
- BBL EPAX::FUNC_nextBbl (FUNC func, BBL bbl)
- bool EPAX::FUNC_isLastBbl (FUNC func, BBL bbl)
- uint32_t EPAX::FUNC_countInsn (FUNC func)
- INSN EPAX::FUNC_findInsn (FUNC func, uint64_t addr)
- INSN EPAX::FUNC_firstInsn (FUNC func)
- INSN EPAX::FUNC_nextInsn (FUNC func, INSN insn)
- bool EPAX::FUNC_isLastInsn (FUNC func, INSN insn)
- CFG EPAX::FUNC_cfg (FUNC func)
- uint32_t EPAX::FUNC_countTargets (FUNC func)
- uint32_t EPAX::FUNC_targets (FUNC func, std::vector< FUNC > &funcList)
- uint32_t EPAX::CFG_countLoop (CFG cfg)
- LOOP EPAX::CFG_findLoop (CFG cfg, uint64_t addr)
- LOOP EPAX::CFG_firstLoop (CFG cfg)
- LOOP EPAX::CFG_nextLoop (CFG cfg, LOOP loop)
- bool EPAX::CFG_isLastLoop (CFG cfg, LOOP loop)
- CFG EPAX::LOOP_cfg (LOOP loop)
- FUNC EPAX::LOOP_func (LOOP loop)
- uint32_t EPAX::LOOP_size (LOOP loop)
- uint32_t EPAX::LOOP_countBbl (LOOP loop)
- BBL EPAX::LOOP_findBbl (LOOP loop, uint64_t addr)
- BBL EPAX::LOOP_firstBbl (LOOP loop)
- BBL EPAX::LOOP_nextBbl (LOOP loop, BBL bbl)
- bool EPAX::LOOP_isLastBbl (LOOP loop, BBL bbl)
- uint32_t EPAX::LOOP_countInsn (LOOP loop)
- INSN EPAX::LOOP_findInsn (LOOP loop, uint64_t addr)
- INSN EPAX::LOOP_firstInsn (LOOP loop)
- INSN EPAX::LOOP_nextInsn (LOOP loop, INSN insn)
- bool EPAX::LOOP_isLastInsn (LOOP loop, INSN insn)
- BBL EPAX::LOOP_head (LOOP loop)
- BBL EPAX::LOOP_tail (LOOP loop)

- uint32_t EPAX::LOOP_countExits (LOOP loop)
- uint32_t EPAX::LOOP_exits (LOOP loop, std::vector< INSN > &insnList)
- bool EPAX::LOOP_isInnerLoop (LOOP loop1, LOOP loop2)
- LOOP EPAX::LOOP_parent (LOOP loop)
- uint32_t EPAX::LOOP_index (LOOP loop)
- uint32_t EPAX::LOOP_depth (LOOP loop)
- bool EPAX::BBL_isHead (BBL bbl, INSN insn)
- bool EPAX::BBL_isTail (BBL bbl, INSN insn)
- INSN EPAX::BBL_head (BBL bbl)
- INSN EPAX::BBL_tail (BBL bbl)
- FUNC EPAX::BBL_func (BBL bbl)
- LOOP EPAX::BBL_loop (BBL bbl)
- uint32_t EPAX::BBL_size (BBL bbl)
- uint64_t EPAX::BBL_addr (BBL bbl)
- uint32_t EPAX::BBL_countInsn (BBL bbl)
- INSN EPAX::BBL_findInsn (BBL bbl, uint64_t addr)
- INSN EPAX::BBL_firstInsn (BBL bbl)
- INSN EPAX::BBL_nextInsn (BBL bbl, INSN insn)
- bool EPAX::BBL_isLastInsn (BBL bbl, INSN insn)
- uint32_t EPAX::BBL_countTargets (BBL bbl)
- uint32_t EPAX::BBL_targets (BBL bbl, std::vector< BBL > &bblList)
- bool EPAX::BBL_hasFallthroughTarget (BBL bbl)
- BBL EPAX::BBL_fallthroughTarget (BBL bbl)
- uint32_t EPAX::BBL_countJumpTargets (BBL bbl)
- uint32_t EPAX::BBL_jumpTargets (BBL bbl, std::vector< BBL > &bblList)
- uint32_t EPAX::BBL_countSources (BBL bbl)
- uint32_t EPAX::BBL_sources (BBL bbl, std::vector< BBL > &bblList)
- uint32_t EPAX::INSN_targets (INSN insn, std::vector< uint64_t > &tlist)
- BBL EPAX::INSN_bbl (INSN insn)
- FUNC EPAX::INSN_func (INSN insn)
- LOOP EPAX::INSN_loop (INSN insn)
- uint64_t EPAX::INSN_addr (INSN insn)
- std::string EPAX::INSN_string (INSN insn)
- uint64_t EPAX::INSN_callTarget (INSN insn)
- bool EPAX::INSN_isBranch (INSN insn)
- bool EPAX::INSN_isFpop (INSN insn)
- bool EPAX::INSN_isMemop (INSN insn)
- uint32_t EPAX::INSN_size (INSN insn)
- std::string EPAX::INSN_condName (INSN insn)
- bool EPAX::INSN_fallsThrough (INSN insn)
- uint32_t EPAX::INSN_sourceRegisterSizeInBits (INSN insn)
- uint32_t EPAX::INSN_sourceDatatypeSizeInBits (INSN insn)

4.1.1 Detailed Description

4.1.2 LICENSE

Definition in file Interface.hpp.