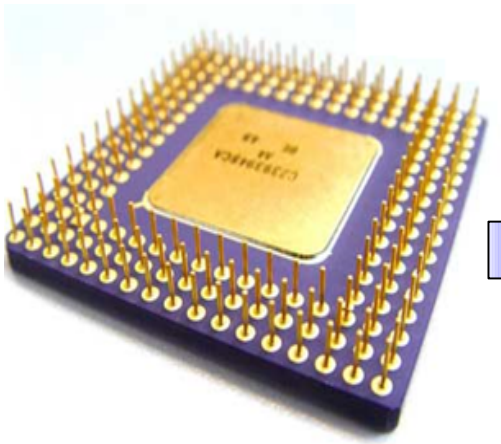
# SMT-based Test Program Generation for Cache Memory

Evgeny Kornikhin
Moscow State University

# Testing by Test Programs

add r1,r2,r3
sub r4, r1, r2
lw r5, r1, 0
lui r2, r1, r4

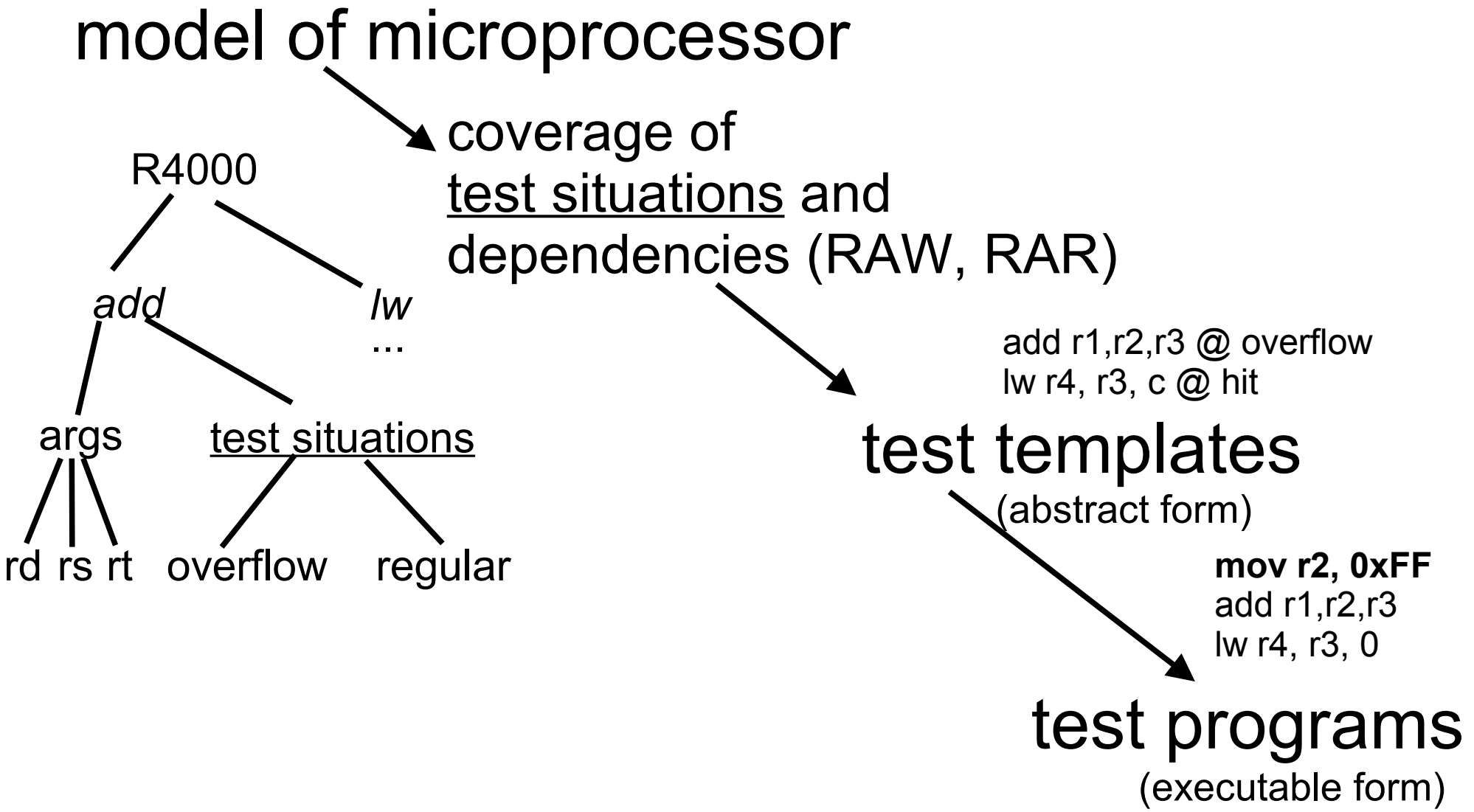


Y/N

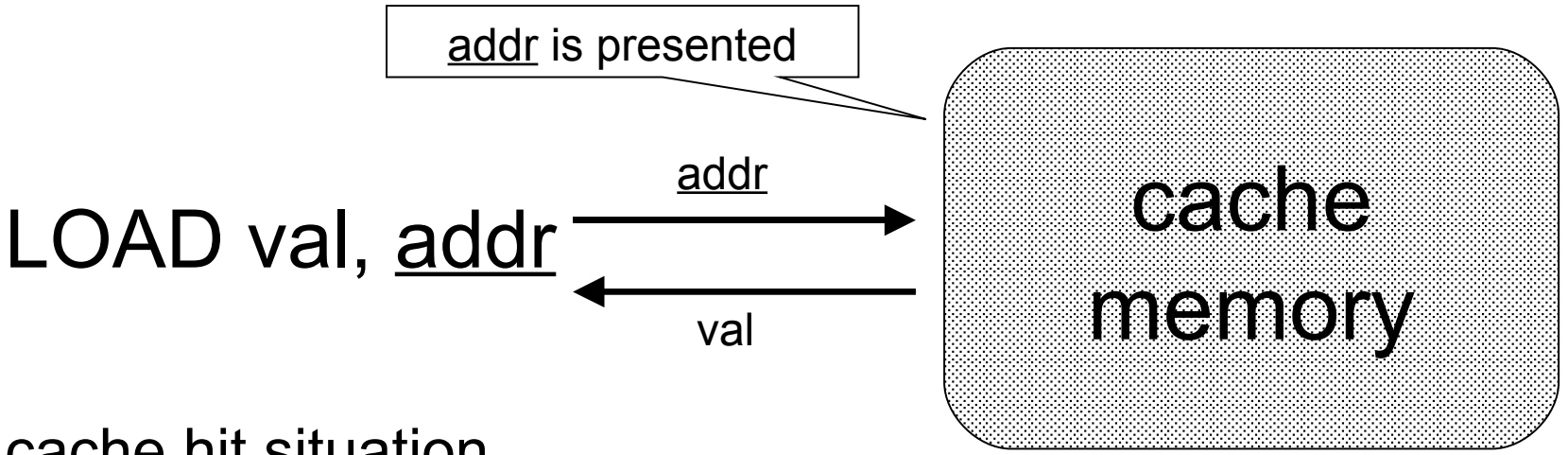
assembler program ( test program )

microprocessor

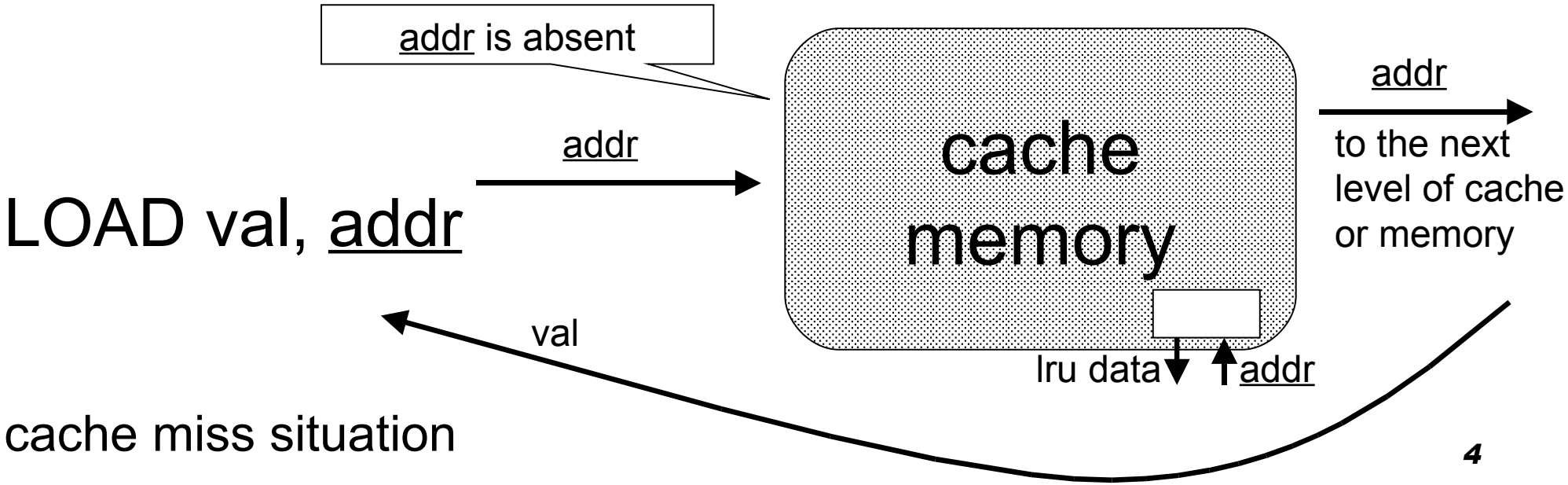
# Test Programs Generation



# Cache memory behavior



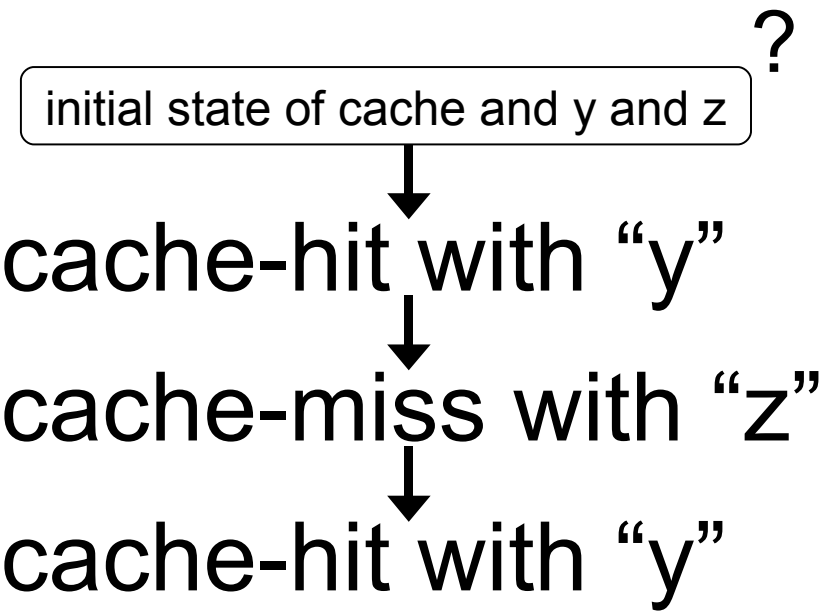

cache hit situation




cache miss situation

# Problem again

LOAD x, y @ hit
STORE u, z @ miss
LOAD z, y @ hit




initial state of cache

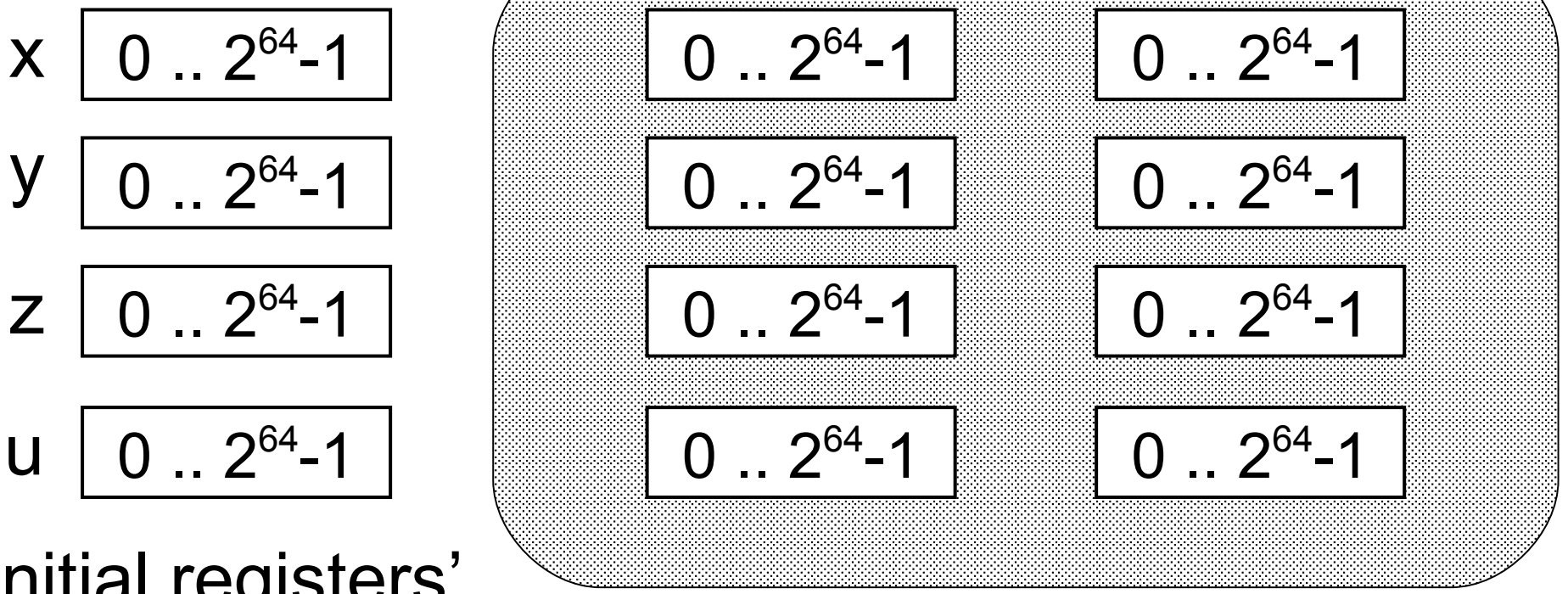
(addresses of cached data)

and registers = ?

# Huge exhaustive search

LOAD x, y @ hit
STORE u, z @ miss
LOAD z, y @ hit

select values
↓ ↑
run template



# Hard behavior constraints

x,y,z,u, L such as :

LOAD x, y @ hit
(cache-hit with “y”)

- index(L,y) > 0
- L’ == inc_counter(L,y)

STORE u,z @ miss→z0
(cache-miss with “z”)

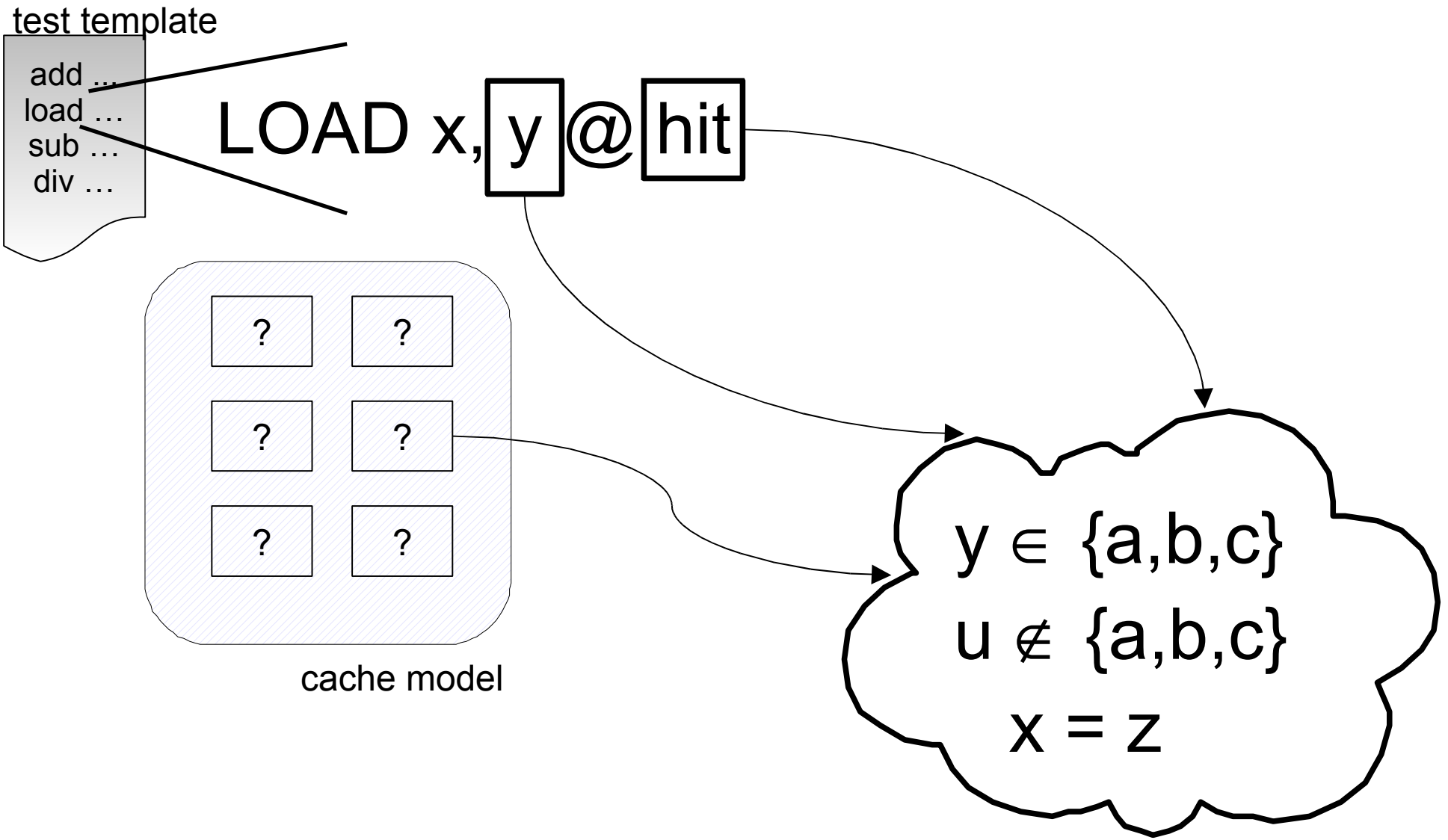
- index(L’,z) == 0
- ∃ z0:counter(L’,z0)→min
- L’’ == remove(L’,z0)
- L’’’ == add(L’’,z)

LOAD z, y @ hit
(cache-hit with “y”)

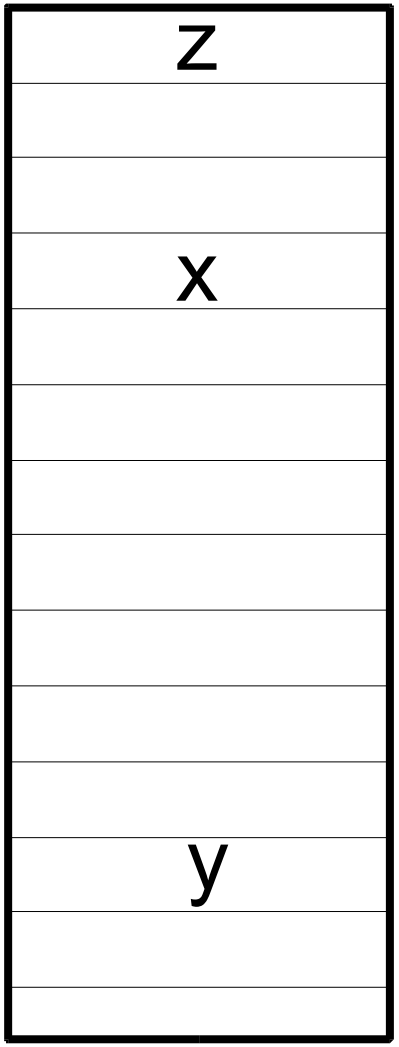
- index(L’’’,y) > 0

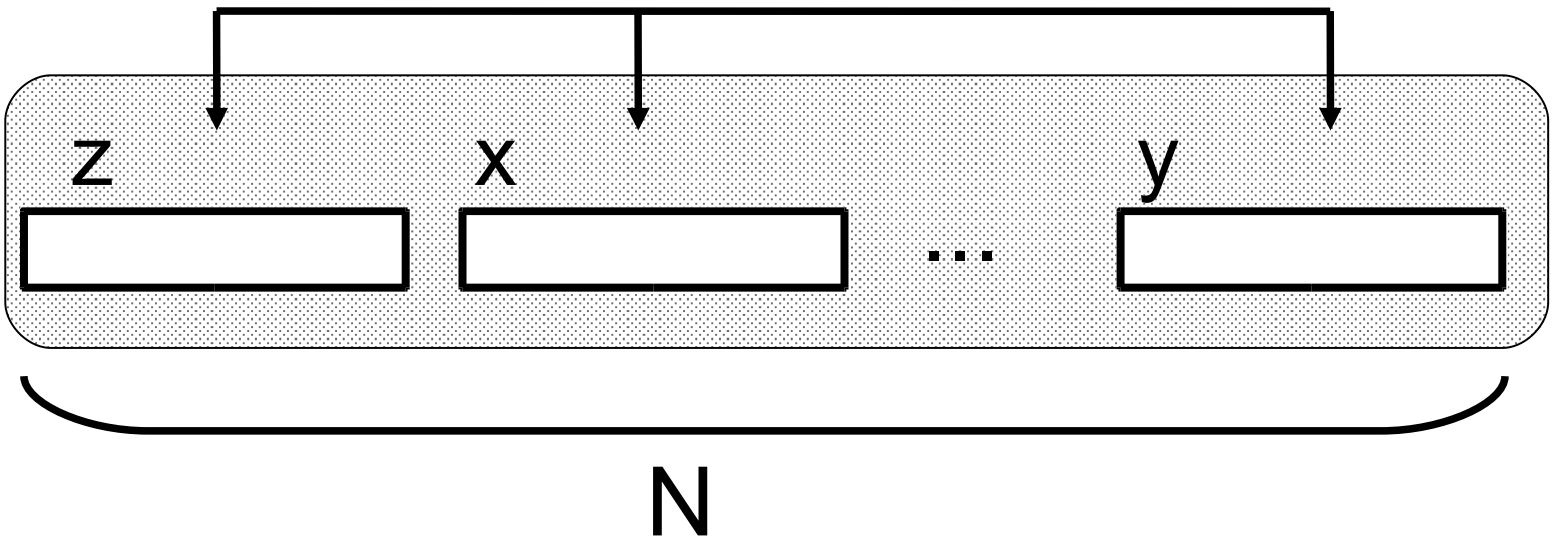
x,y,z,u, L = ?

# Key Idea


test template
add …
load …
sub …
div …
LOAD x, y @ hit
?
?
?
?
?
?
cache model
y ∈ {a,b,c}
u ∉ {a,b,c}
x = z

# Fully Associative Cache

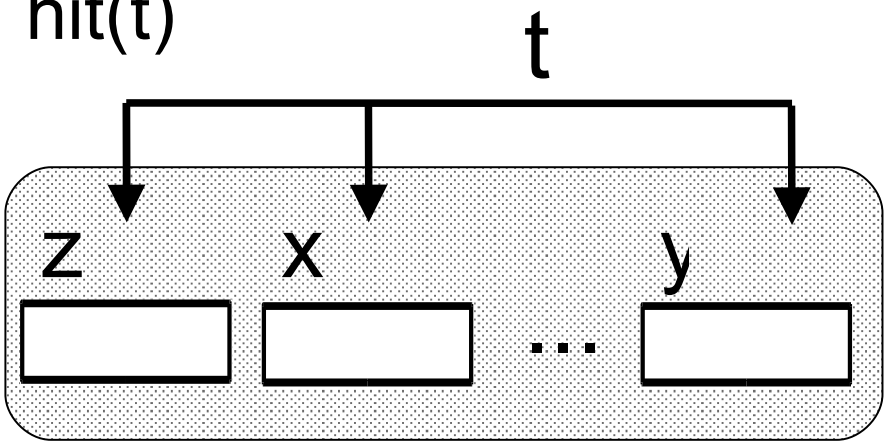



{x,y,z,...} - current state

# Fully Associative Cache

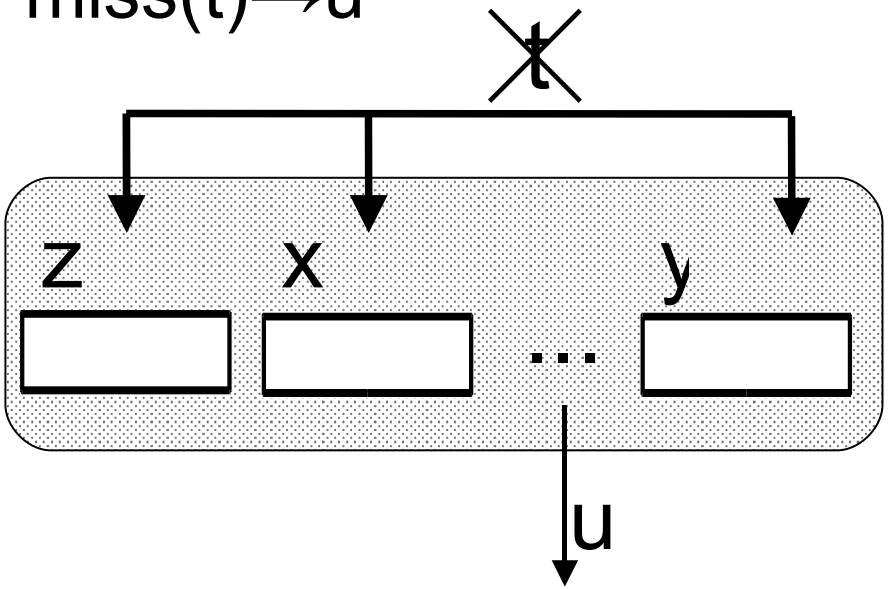
hit(t)




$t \in \{x,y,z...\}$

---

miss(t)→u




$t \notin \{x,y,z...\}$

$u \in \{x,y,z...\}$

lru(u)

new cache $\{t,x,y,z...\} \setminus \{u\}$

# lru(u)

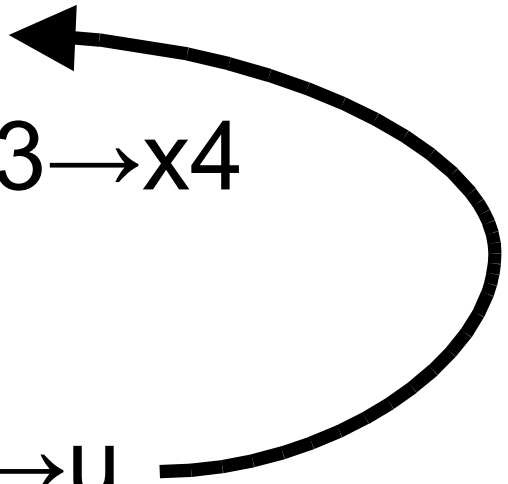
hit x1

hit x2

miss x3→x4

hit x5

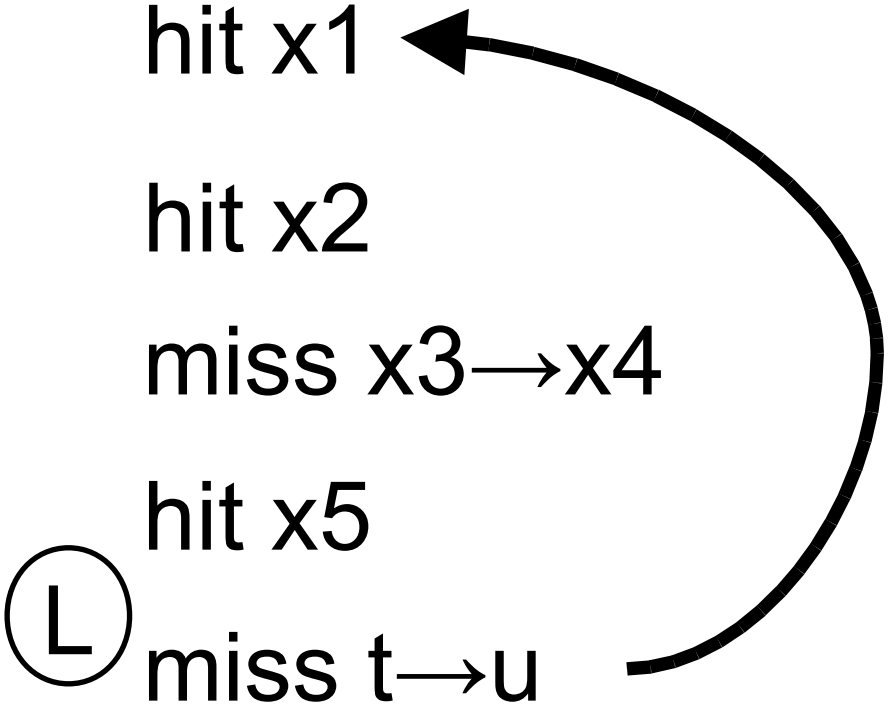
L miss t→u



u = x2

{x3, x5} = L\{u}

# lru(u)



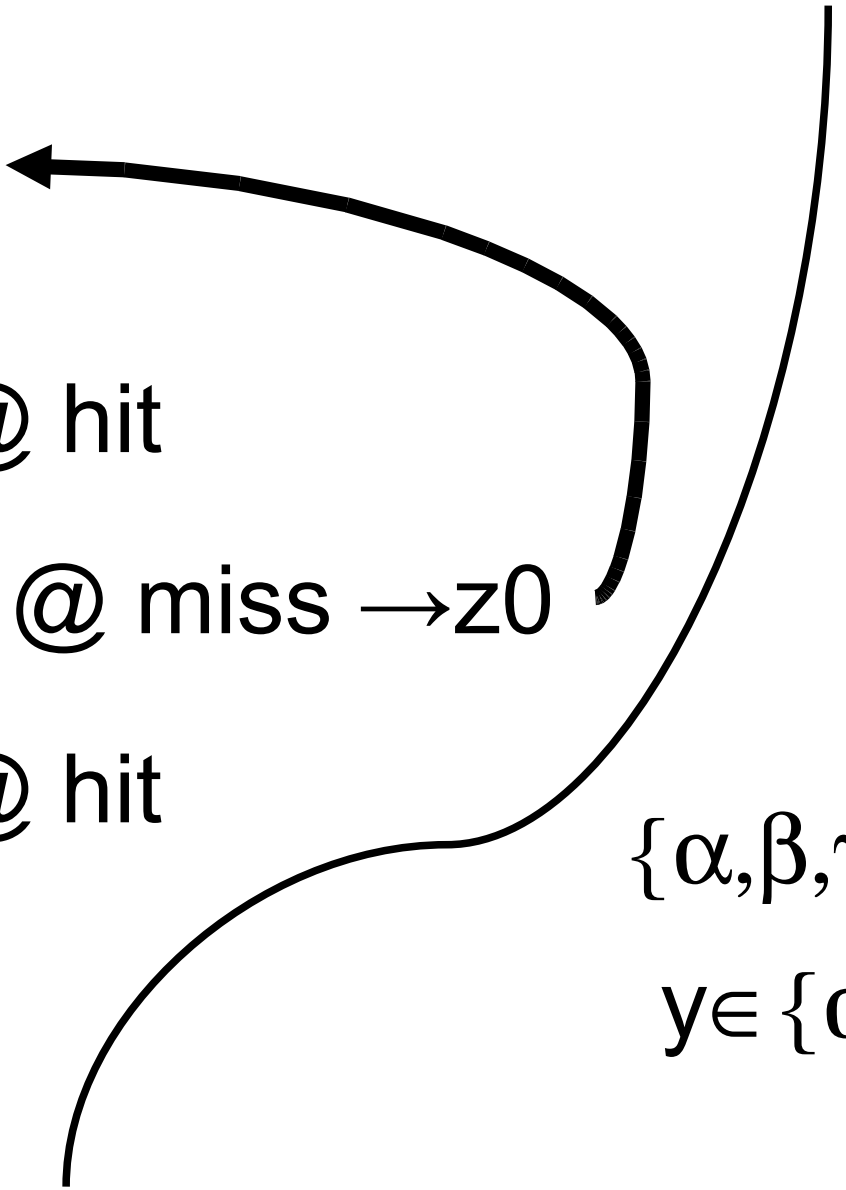

$$\begin{cases} u = x1 \\ \{x2, x3, x5\} = L \backslash \{u\} \end{cases}$$

there are another cases..

# Example

initial state:

$\alpha$
$\beta$
$\gamma$



LOAD x, y @ hit

STORE u, z @ miss $\rightarrow$z0

LOAD z, y @ hit

N = 3

$y \in \{\alpha,\beta,\gamma\}$

$z \notin \{\alpha,\beta,\gamma\}$

$z0 \in \{\alpha,\beta,\gamma\}$

$z0=\beta$

$\{\alpha,\beta,\gamma\} \setminus \{z0\} = \{\gamma,y\}$

$y \in \{\alpha,\beta,\gamma,z\} \setminus \{z0\}$

# Example

$y = \alpha$ $z \notin \{\alpha,\beta,\gamma\}$

$y \in \{\alpha,\beta,\gamma\}$

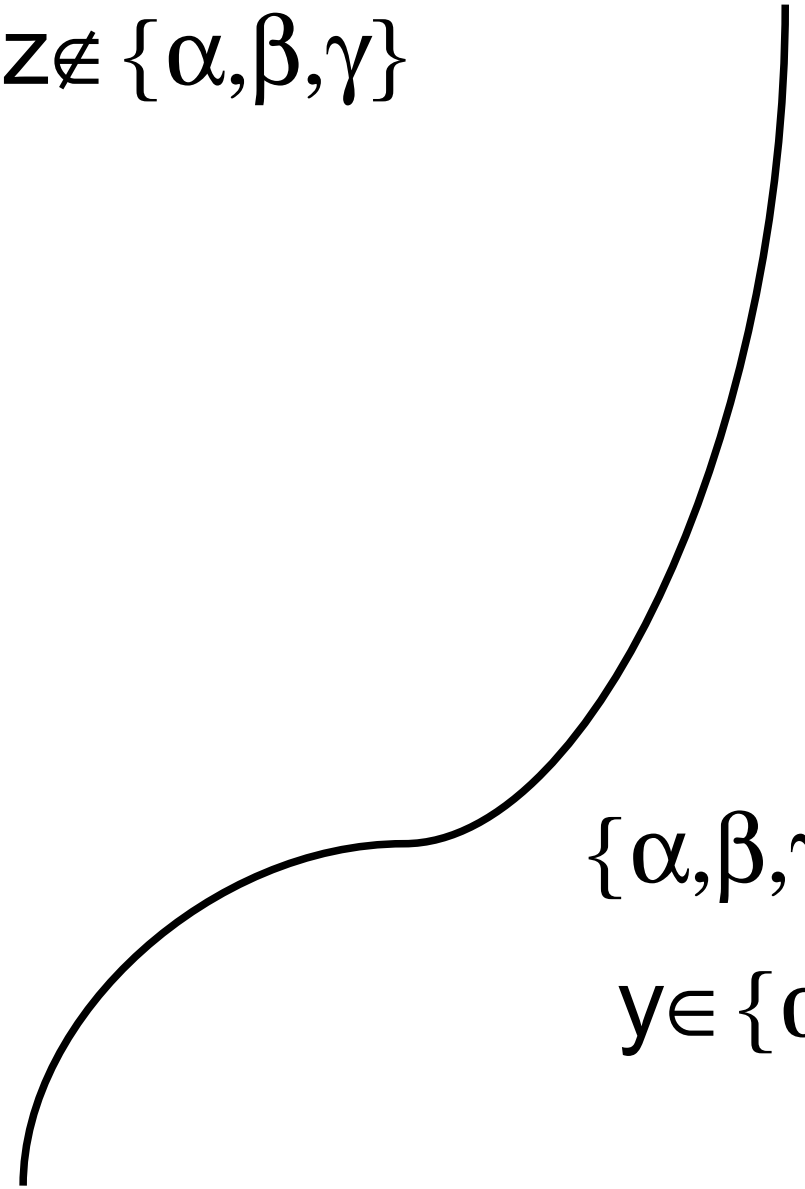
$z \notin \{\alpha,\beta,\gamma\}$

$y=\alpha=0$ $z0 \in \{\alpha,\beta,\gamma\}$

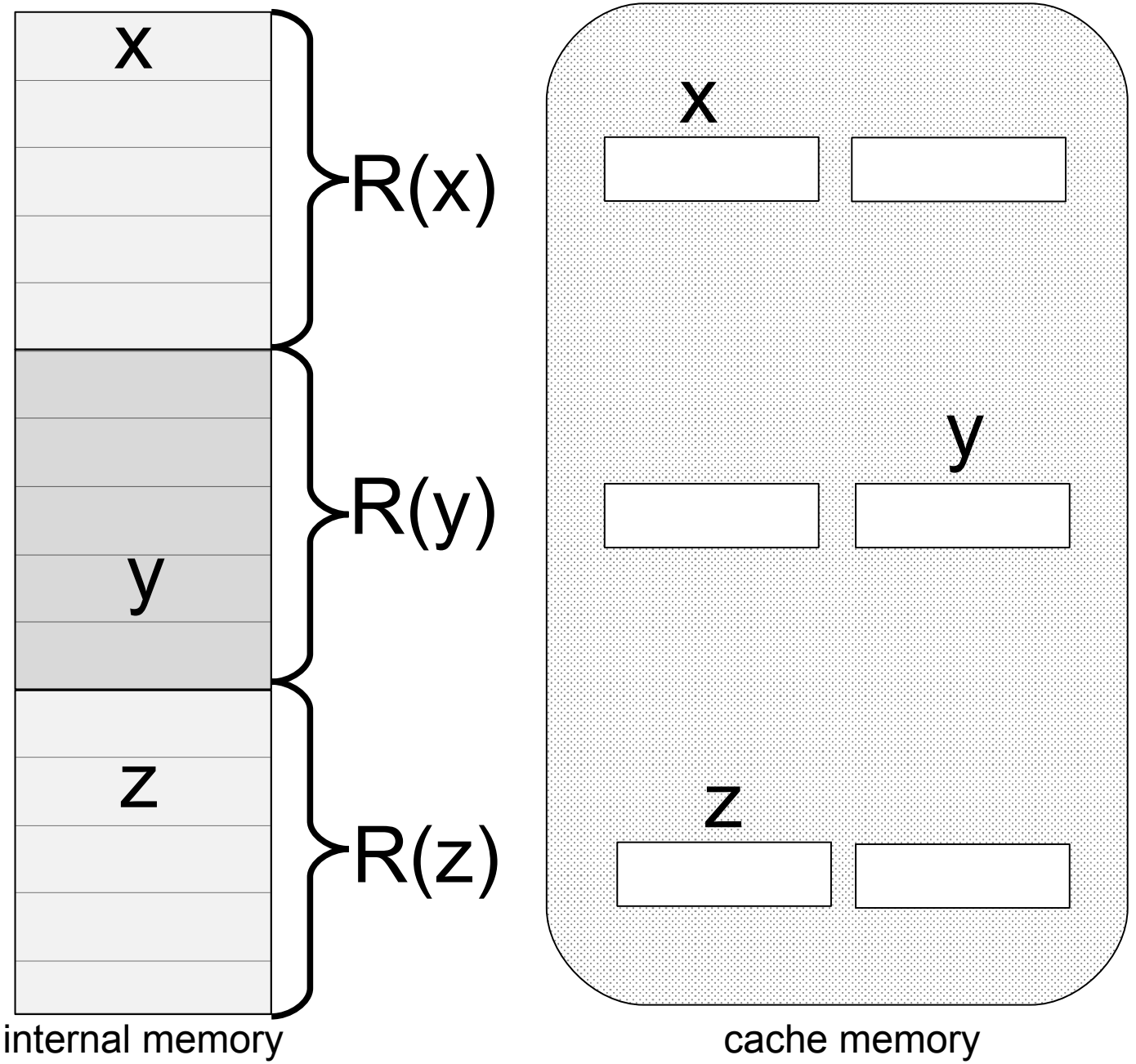
$\beta=1$ $z0=\beta$

$\gamma=2$ $\{\alpha,\beta,\gamma\}\backslash\{z0\}=\{\gamma,y\}$

$z=3$ $y \in \{\alpha,\beta,\gamma,z\}\backslash\{z0\}$



# Common Cache




hit(t)

$t \in L$

miss(t)→u

$u \in L$

$t \notin L$

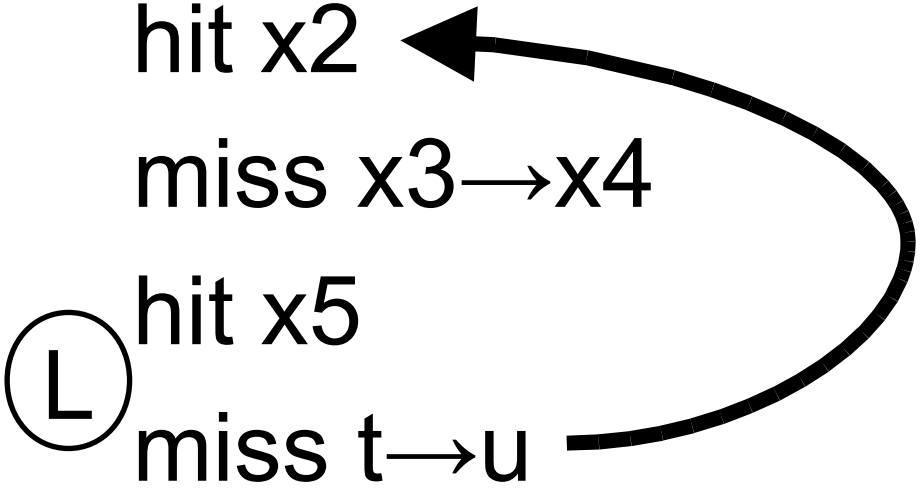
$R(t) = R(u)$

lru(u)

new cache:

$L \cup \{t\} \setminus \{u\}$

# lru(u)




$$\begin{cases} u = x2 \\ \{x3, x5\} \cap R(u) \\ \quad = (L \setminus \{y\}) \cap R(u) \end{cases}$$

# Equations for the whole test template

$x1,x2 \in \{a1,a2,b1,b2,c1,c2\}$

$x3 \notin \{a1,a2,b1,b2,c1,c2\}$

$R(x3) = R(y3)$

$x4 \in \{a1,...,c2,x3\}\backslash\{y3\}$

$x5 \notin \{a1,...,c2,x3\}\backslash\{y3\}$

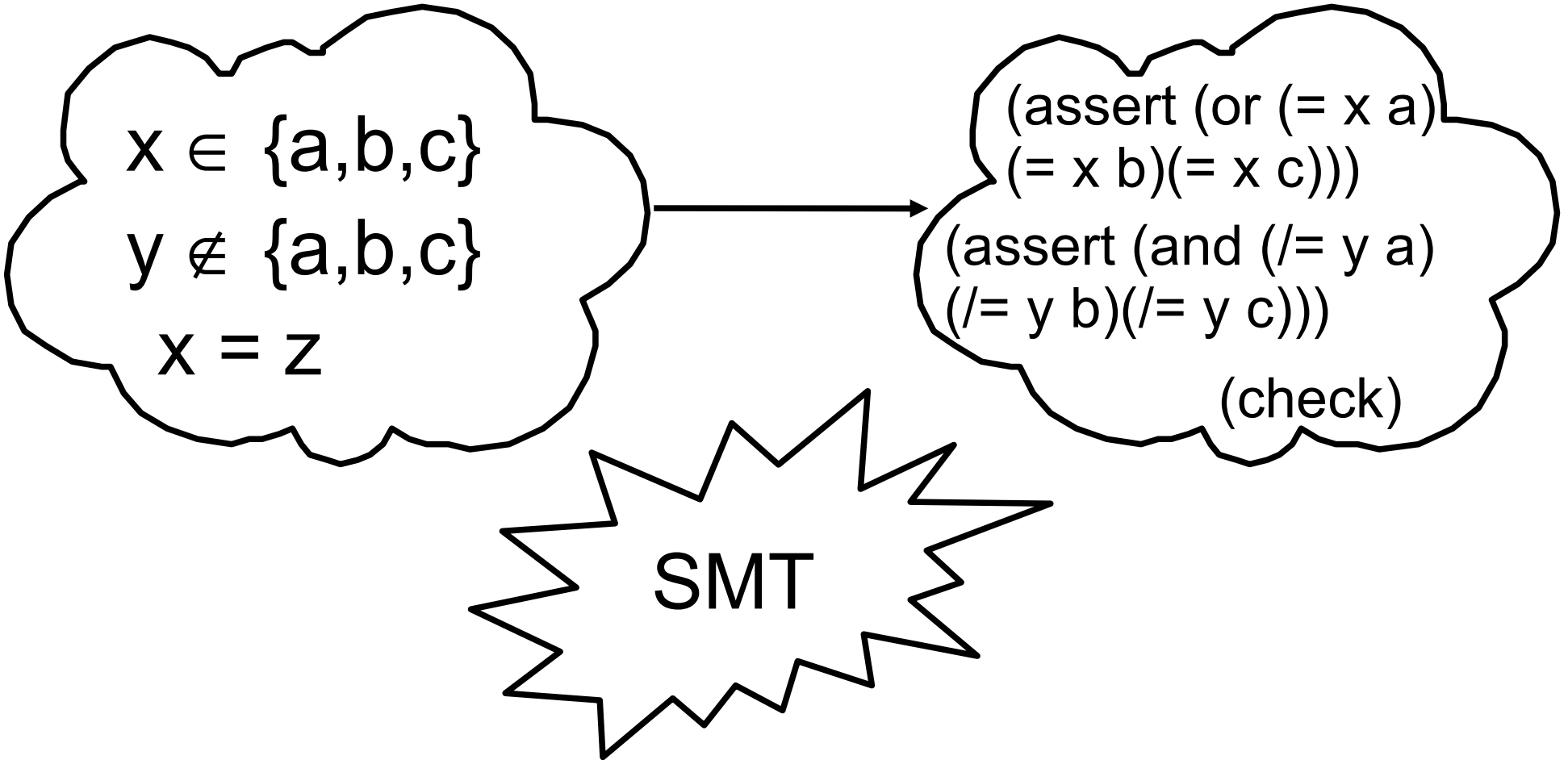
$y3 = c2$

$\{y3\} = (\{a1,...,c2\}\backslash\{x1,x2, y3\})\cap R(y3)$

$y5 = x2$

$\{y5\} = (\{a1...c2,x3\}\backslash\{y3,y5, x3,x4\})\cap R(y5)$

# Solver




SAT modulo theories

Yices

# Conclusions

- Cache behavior can be expressed using equations on addresses' set
- Equations can be solved by SMT-solver

# Contacts

http://tesla-project.googlecode.com

http://hardware.ispras.ru

kornevgen@gmail.com

Thank you for attention!