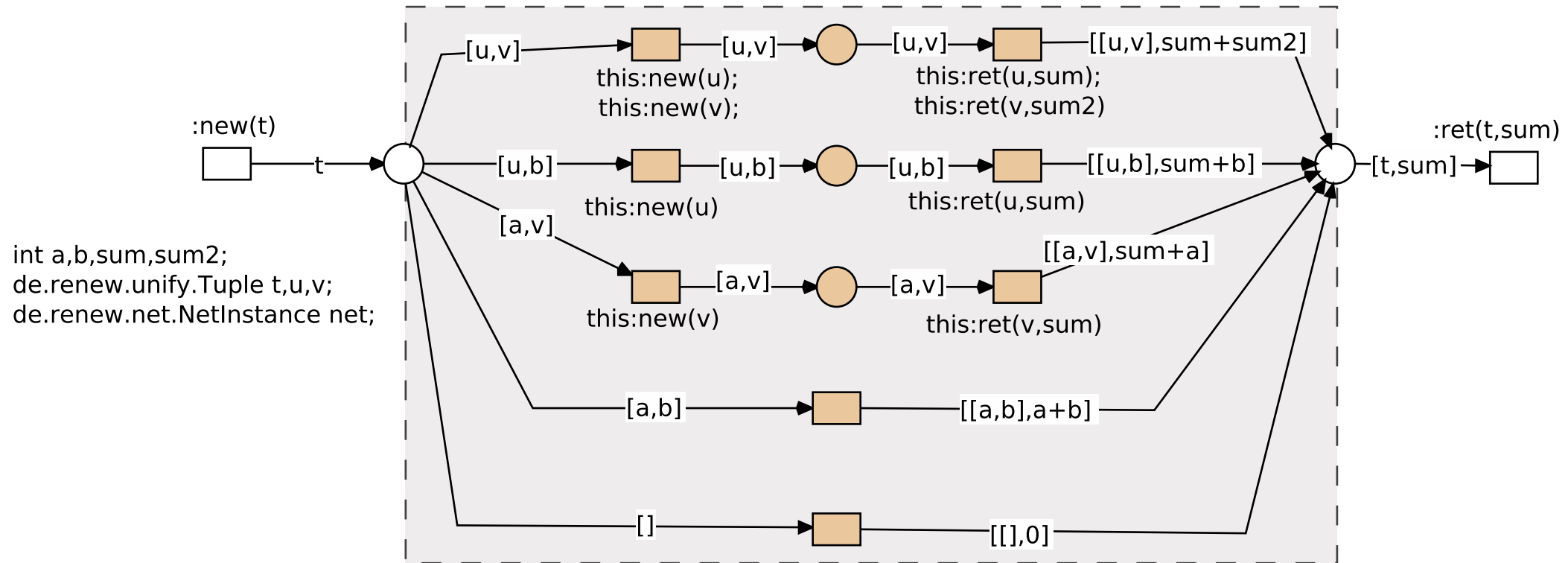# tsum: Tree Sum

Net as call-back function. Single call single return per instance. Call with ':new tsum()' and keep reference for return value.
Expects tuples to be binary (i.e. 2-tuples).
@author Felgenhauer, Hildebrandt, Reichenbach @version 42 @date 2016-01-10

:new(t)

[u,v]

this:new(u);
this:new(v);

[u,v] → [u,v] → [u,v]

this:ret(u,sum);
this:ret(v,sum2)

[[u,v],sum+sum2]

t

[u,b]

this:new(u)

[u,b] → [u,b] → [u,b]

this:ret(u,sum)

[[u,b],sum+b]

:ret(t,sum)

[t,sum]

[a,v]

this:new(v)

[a,v] → [a,v] → [a,v]

this:ret(v,sum)

[[a,v],sum+a]

int a,b,sum,sum2;
de.renew.unify.Tuple t,u,v;
de.renew.net.NetInstance net;

[a,b]

[[a,b],a+b]

[]

[[],0]

[[[7,8],1],[2,[3,4]]];
[[[7,8],1],[[[[7,8],1],[[[[7,8],1],[2,[3,4]]],[3,4]]],[3,4]]];
[1,2];
[[1,2],[3,4]];
[];
[[1,2],1];
[[1,2],[]]

manual

t

[net,t]

net:new tsum(t)

[net,t]

net:ret(t,sum)

[t,sum]