Real Time Systems – SS2016

Prof. Dr. Karsten Weronek

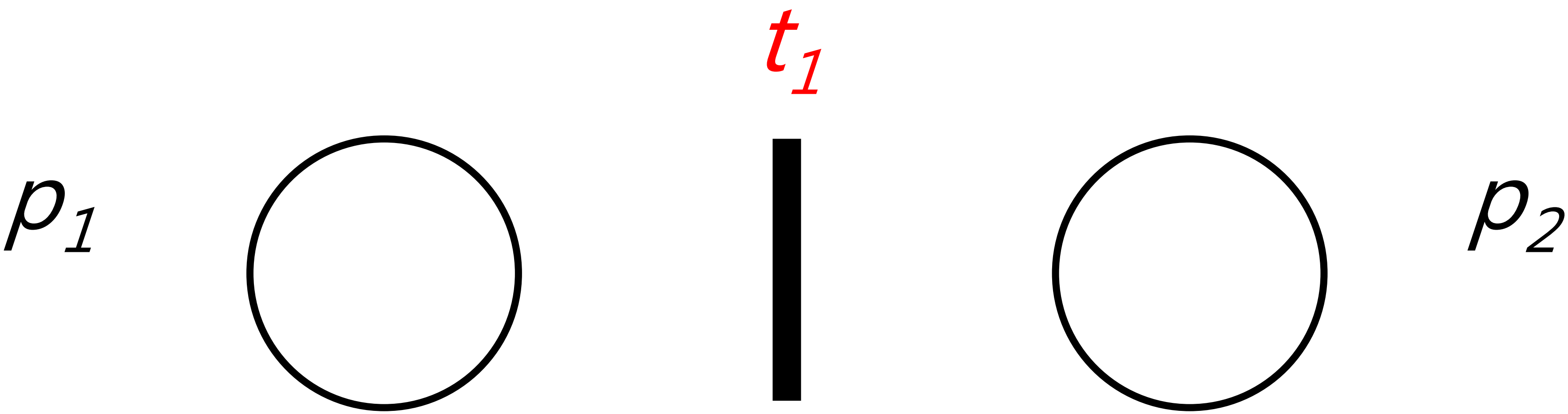
Faculty 2

Computer Science and Engineering

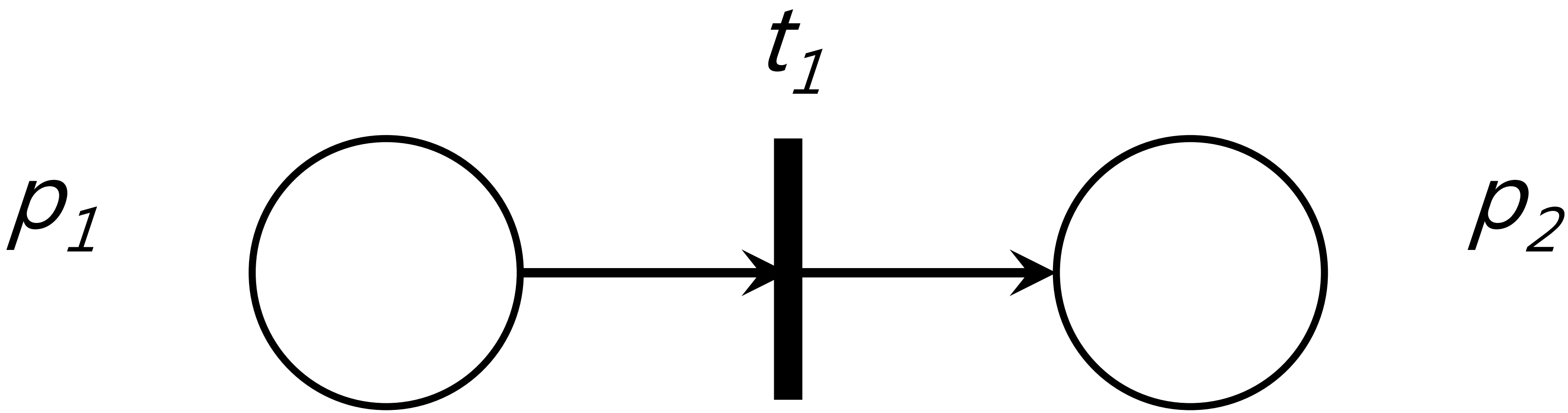
Petri Nets 2

Nodes: Places: $P=\{p_1, p_2, \ldots\}$

Nodes: Transistions: $T=\{ t_1, t_2, \dots \}$



Edges: Flow Relations $F=\{ f_f, f_2, \dots \}$



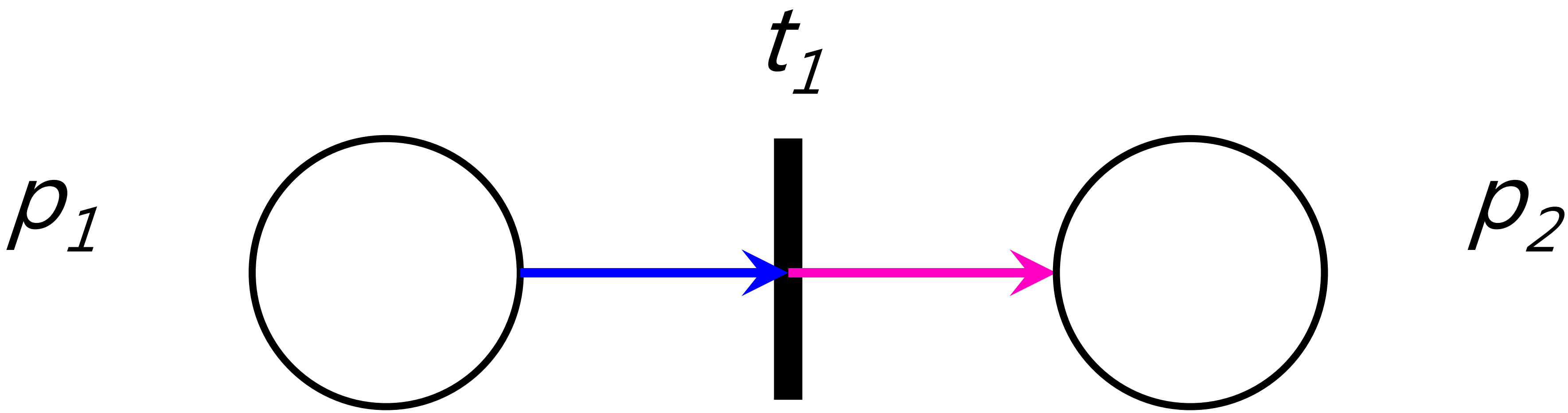
$f_1=(p_1, t_1)$ $f_2=(t_1, p_2)$

$F=F_i \cup F_o=(P \times T_i) \cup (T_o \times P)=\{(p_1,t_1), (t_1,p_2)\}$

F_i : Input Flow Relations,

F_o : Output Flow Relations

Flow Relations: Consumers and Producers



$f_1 = (p_1, t_1)$ $f_2 = (t_1, p_2)$

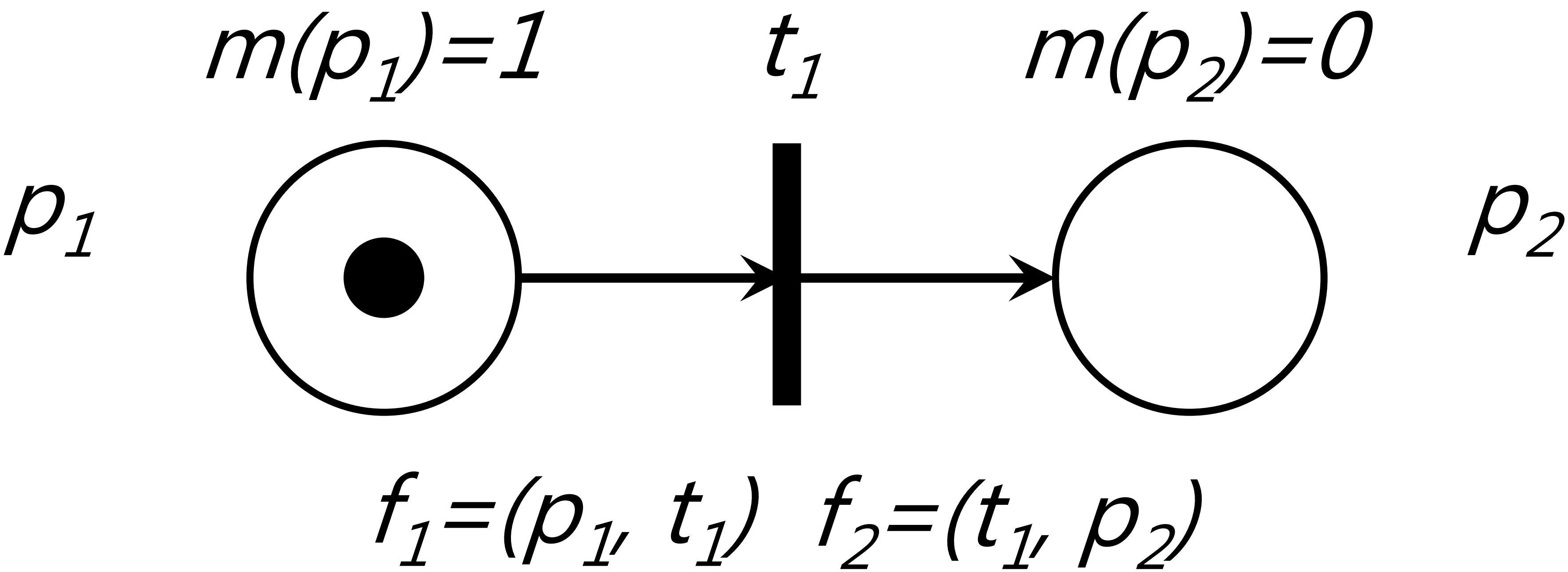
For each transition there are Producers (i) and Consumers (o):

$F_i(t_1) := \{ p \mid (p, t_1) \in F \} = \{ (p_1, t_1) \} = \{ f_1 \}$ (Vorbereich)

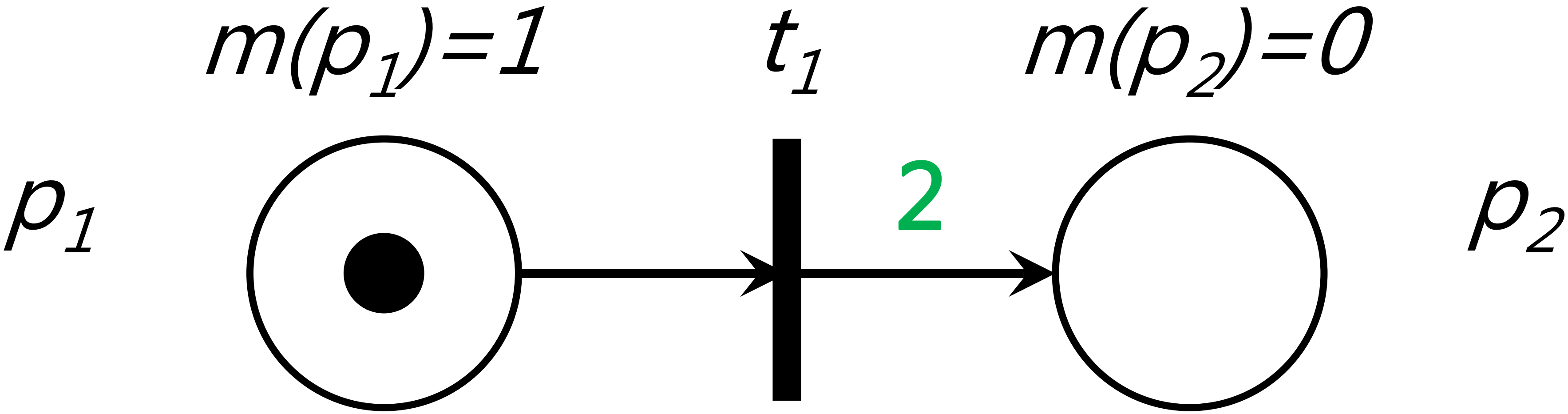
$F_o(t_1) := \{ p \mid (t_1, p) \in F \} = \{ (t_1, p_2) \} = \{ f_2 \}$ (Nachbereich)

Marks in Places: $M = \{\, m(p_1), m(p_2), \ldots \}$ (Tokens)

FRANKFURT UNIVERSITY OF APPLIED SCIENCES



Weights of Flow Relations: $W = \{ w(f_1), w(f_2), \ldots \}$



$f_1=(p_1, t_1)$ $f_2=(t_1, p_2)$

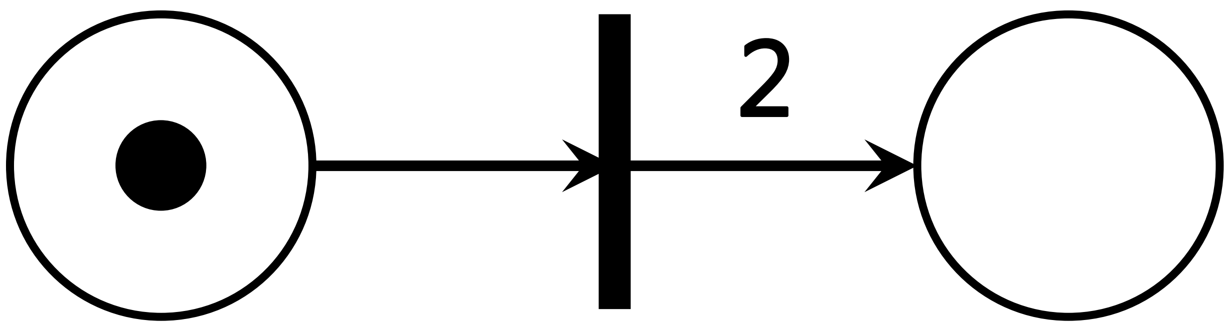
$w(f_1)=1$ $w(f_2)=2$

Capacity of Places: $C = \{ c(p1), c(p2), \ldots \}$

$m(p_1)=1$ t_1 $m(p_2)=0$

p_1

$c(p_1)=1$



p_2

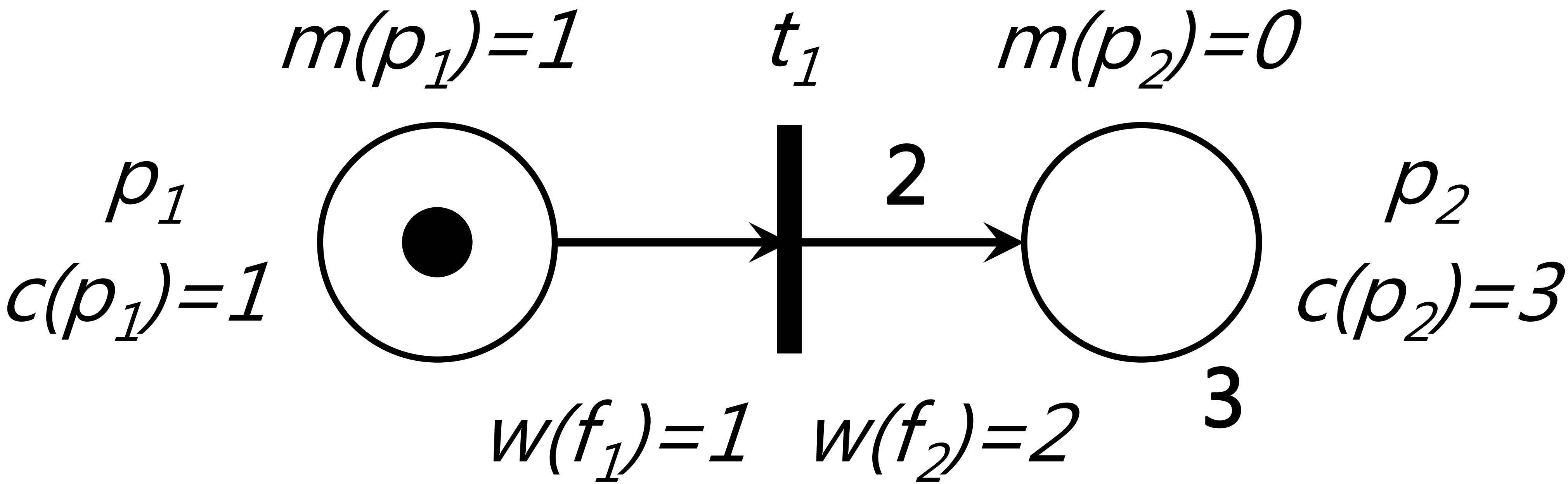
$c(p_2)=3$

3

$f_1=(p_1, t_1)$ $f_2=(t_1, p_2)$

$w(f_1)=1$ $w(f_2)=2$

Switching/Firing-Conditions



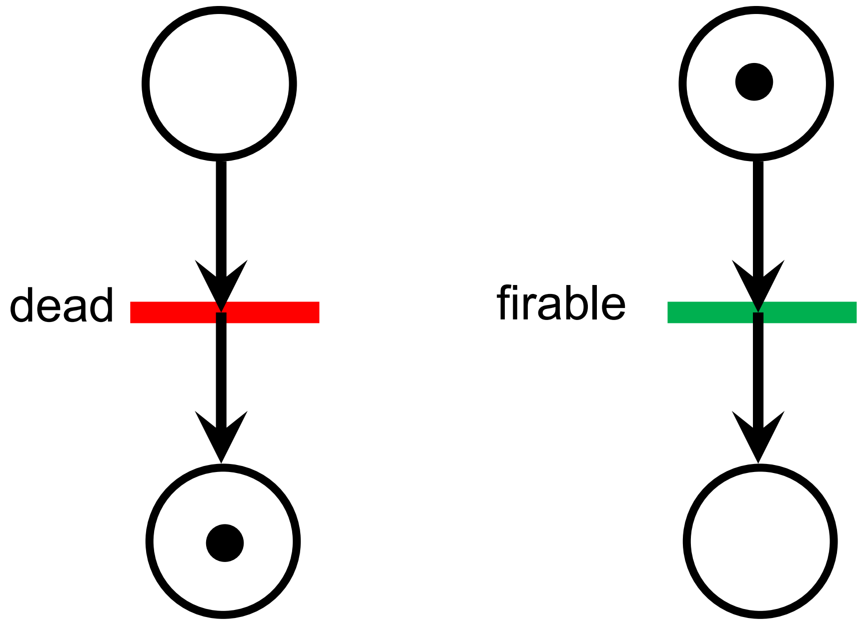
Enabling Rule:

A transition ***t*** is said to be enabled

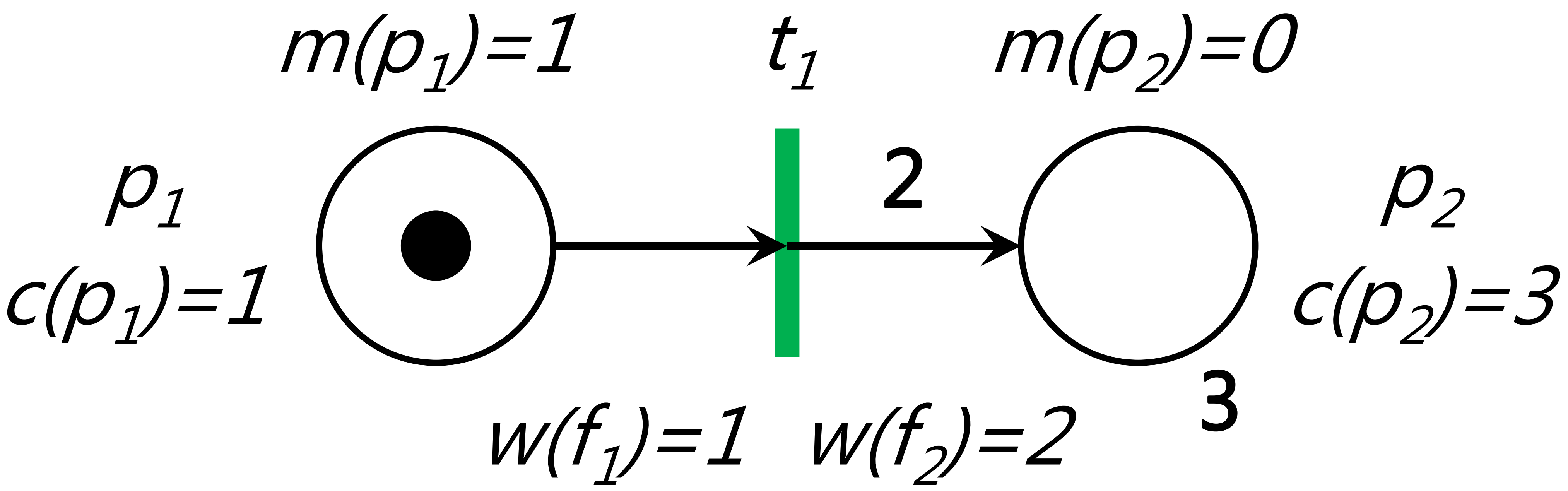
- if each input place ***p*** of ***t*** contains
  at least the number of tokens equal to the weight of the directed arc connection ***p*** to ***t***,
  i.e., $m(p) \geq i(t,p)$ for all ***p*** in ***P*** and
- If each output place ***p*** of ***t*** contains
  at most the number of tokens equal to the weight minus the capacity of the target place
  i.e., $m(p) \leq c(t,p) - o(p,t)$ for all ***p*** in ***P*** and

If $i(t,p) = 0$ or $o(p,t) = 0$, the ***t*** and ***p*** are not connected, so
we do not care about the marking of ***p*** when considering the firing of ***t***.

Colour Code



Switching/Firing-Conditions



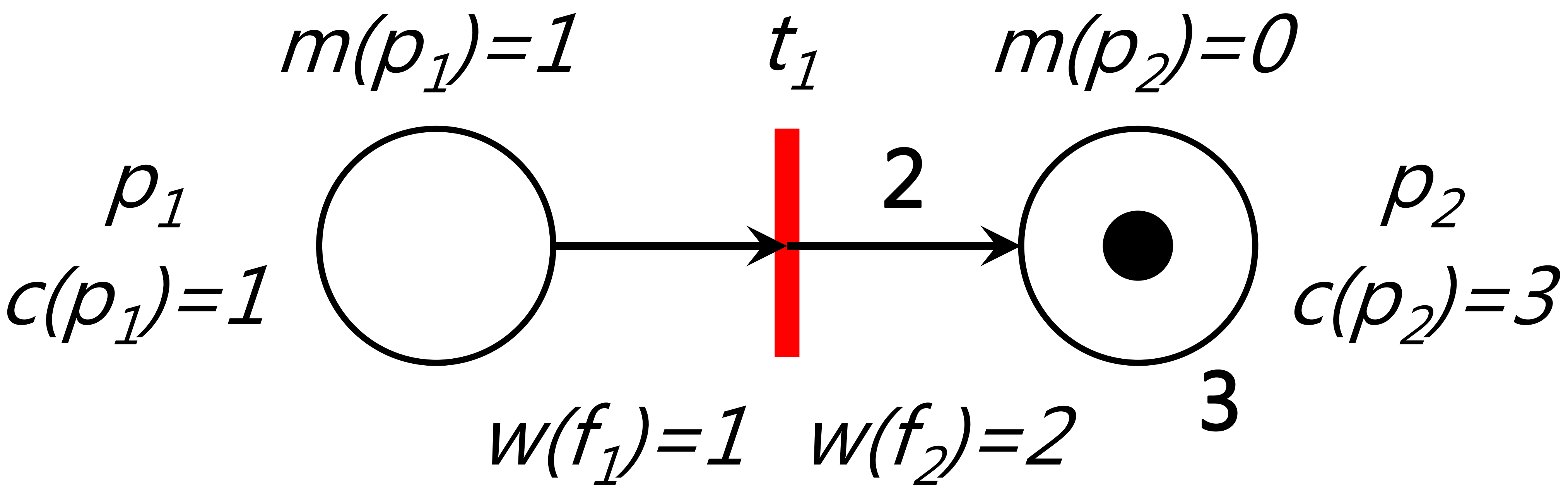
Firing Rule:

Only enabled transitions can fire. The firing of an enabled transition t removes from each input place p the number of tokens equal to i(t,p) and deposits in each output place p the number of tokens equal to o(p,t).

Mathematically, firing t at m yields a new marking

m'(p) = m(p)- i(t,p) + o(p,t) for all p in P

Switching/Firing-Conditions



Firing Rule:

Only enabled transitions can fire. The firing of an enabled transition t removes from each input place p the number of tokens equal to i(t,p) and deposits in each output place p the number of tokens equal to o(t,p).

Mathematically, firing t at m yields a new marking

M'(p) = M(p)- I(t,p) + o(p,t) for all p in P

Lemmas

- Note that since only enabled transitions can fire, the number of tokens in each place always remains nonnegative when a transition is fired.
- Firing can never remove a token that is not there.
- When more than one transition is enabled the decision which one fires first is non-deterministic, unless a firing policy is defined.