4. File Transfer Protocol

Jean-Raymond Abrial

2009

Purpose of this Lecture

- To introduce another example: the file transfer protocol

- To present a number of additional mathematical conventions

- To slighly enlarge the usage of the Proof Obligation Rules

- Example studied in many places, in particular in the following book

- L. Lamport *Specifying Systems: The TLA+ Language and Tools for Hardware and Software Engineers* Addison-Wesley 1999

An Example: File Transfer Protocol

- A file is to be transfered from a Sender to a Receiver

- On the Sender's side the file is called f

- On the Receiver's side the file is called g

- At the beginning of the protocol, g is supposed to be empty

- At the end of the protocol, g should be equal to f

The protocol ensures the copy of a file from one site to another one	FUN-1

The file is supposed to be made of a sequence of items	FUN-2

The file is send piece by piece between the two sites	FUN-3

Modeling Approach

- Our approach at modeling is one of an external observer

- The observer “sees” the state space first from very far away

- He then approaches the future system and sees more details

- As he approaches he also sees more things happening

Development Approach

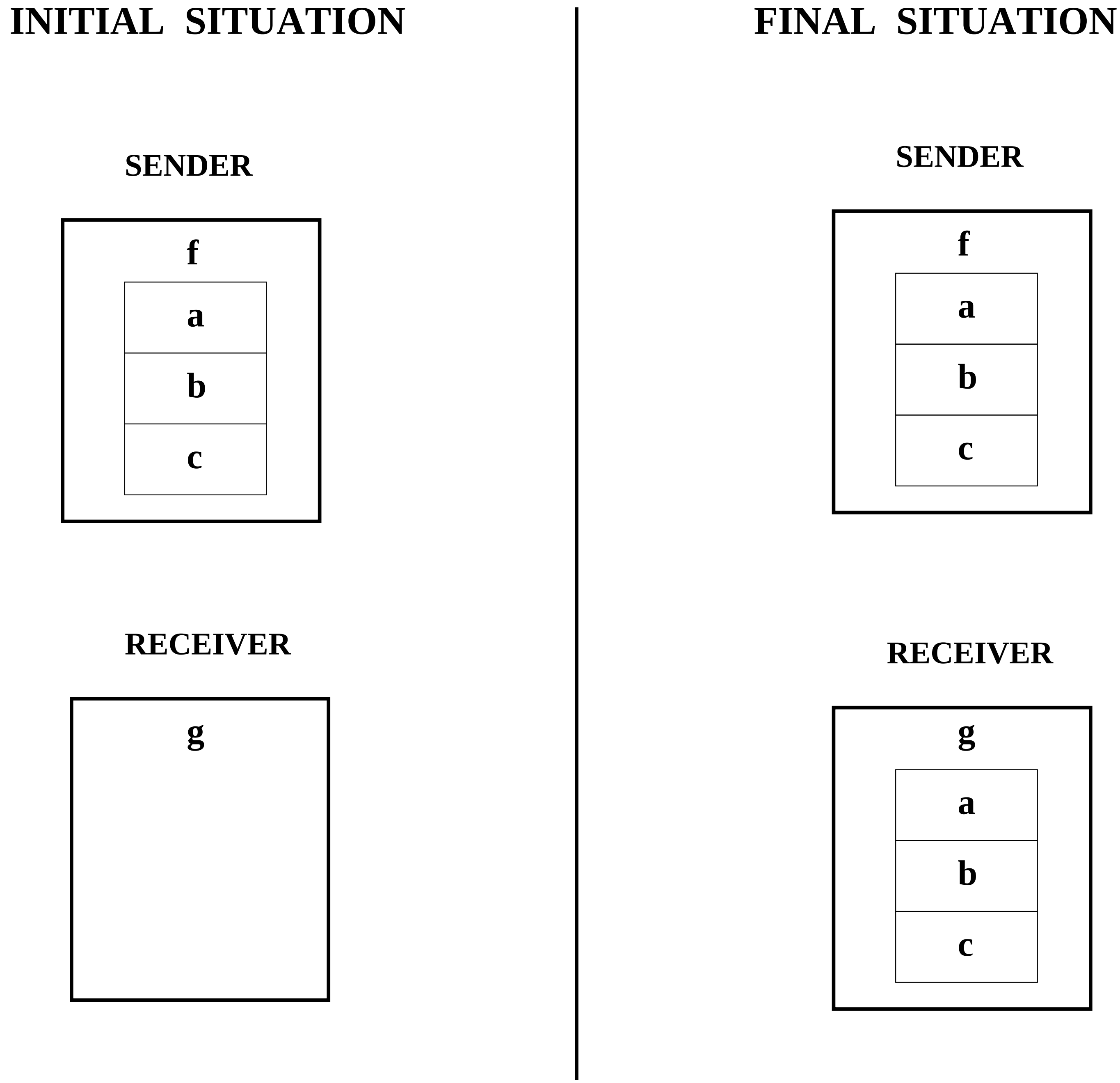
- Initial model: The file is transmitted in one shot (FUN1 and FUN2)

- First refinement: The file is transmitted gradually (FUN3)

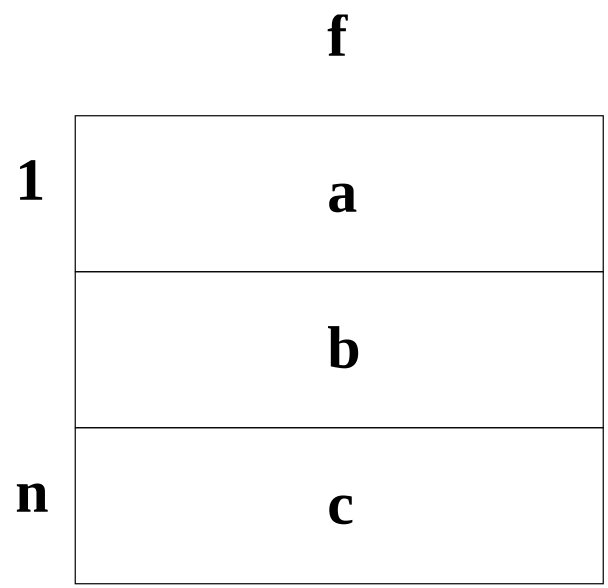
- Second refinement: The two agents are separated

- Third refinement: Towards an implementation

The Sender and the Receiver: a First View



File transfer. The constant part of the state: n and f



carrier sets: D

constants: n, f

axm0_1: $n \in \mathbb{N}$

axm0_2: $0 < n$

axm0_3: $f \in 1 \,..\, n \longrightarrow D$

variables: g

inv0_1: $g \in \mathbb{N} \leftrightarrow D$

- The carrier set D makes this development generic

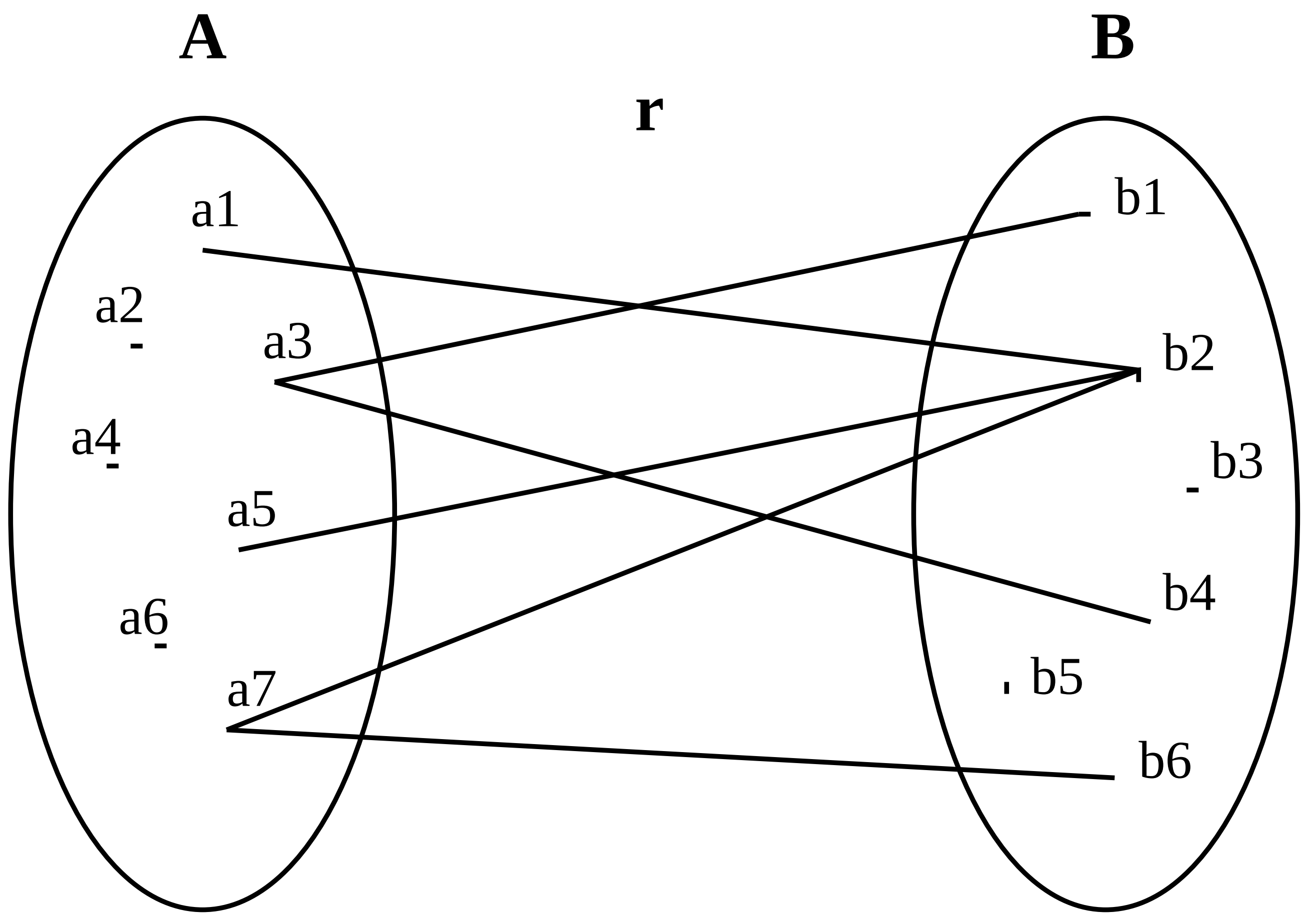
Reminder of Mathematical Conventions (1)

$x \in S$	set membership operator
$\mathbb{N}$	set of natural numbers: $\{0, 1, 2, 3, \ldots\}$
$a \mathinner{..} b$	interval from a to b: $\{a, a+1, \ldots, b\}$ (empty when $b < a$)
$a \mapsto b$	pair constructing operator
$S \times T$	Cartesian product operator
$S \subseteq T$	set inclusion operator
$\mathbb{P}(S)$	power set operator

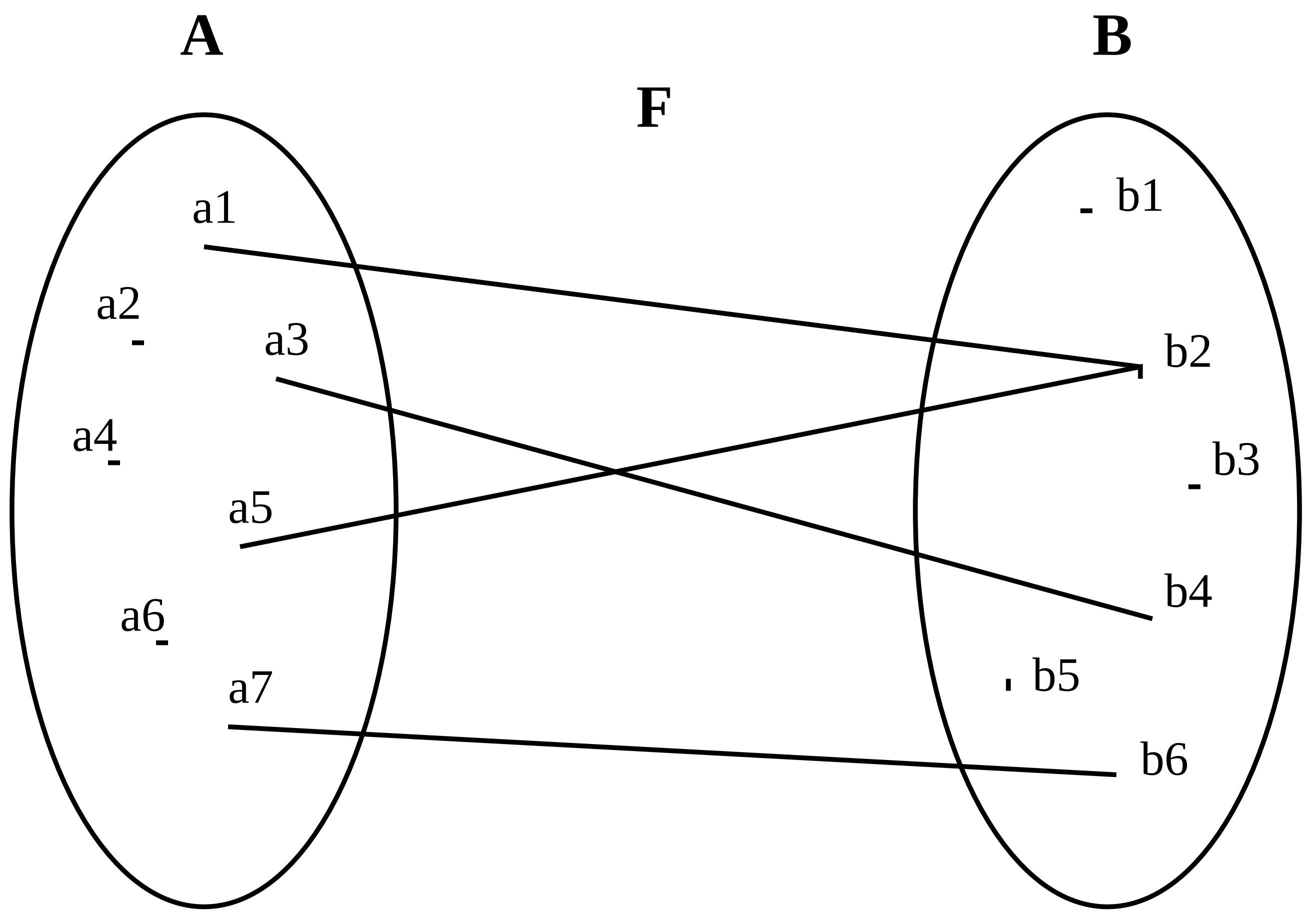
Reminder of Mathematical Conventions (2)

$S \leftrightarrow T$	set of binary relations from S to T
$S \rightarrow T$	set of total functions from S to T
$S \nrightarrow T$	set of partial functions from S to T
$\mathrm{dom}(r)$	domain of a relation r
$\mathrm{ran}(r)$	range of a relation r

A Binary Relation r from a Set A to a Set B

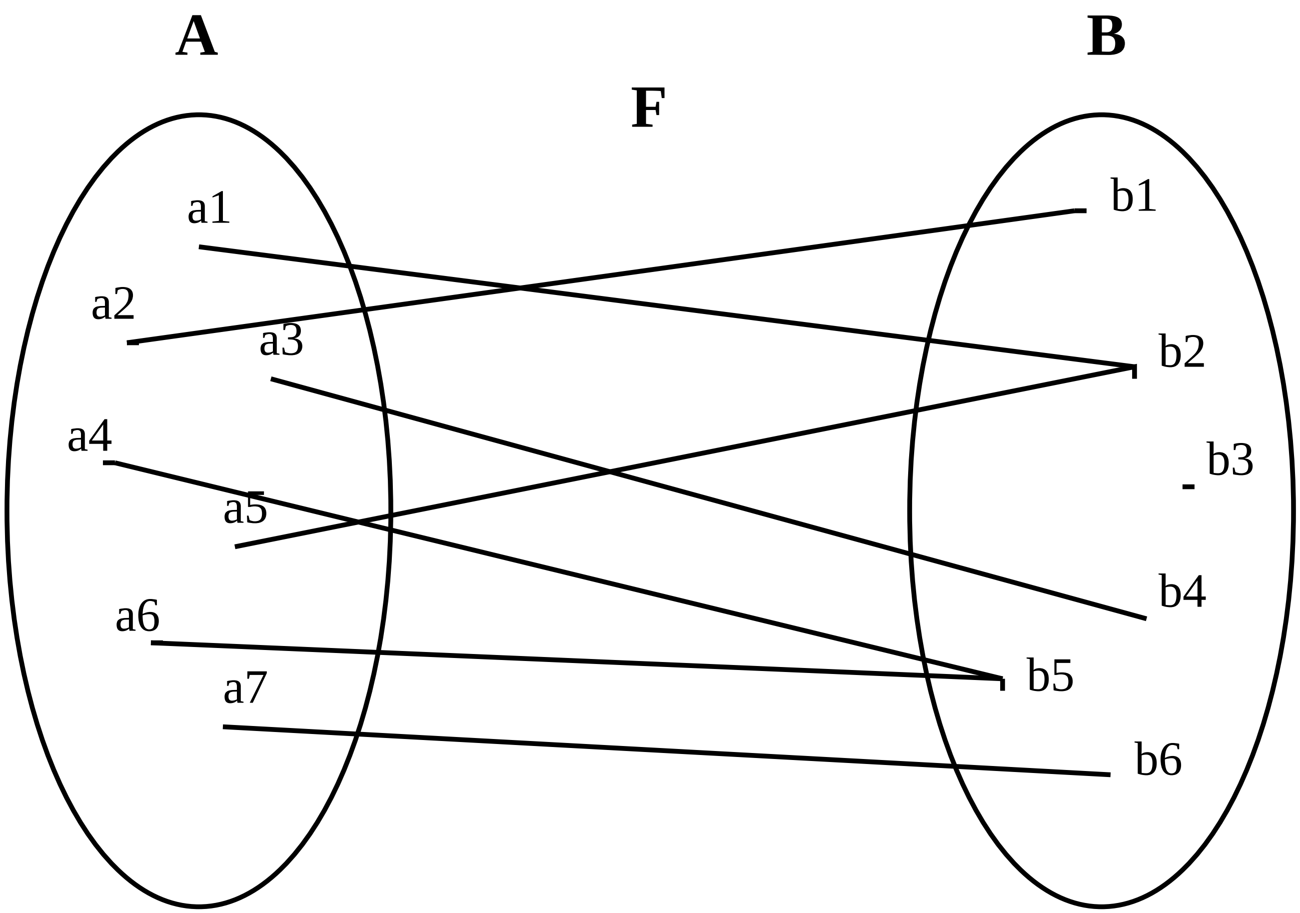


A Partial Function F from a Set A to a Set B



$$\begin{aligned} F &= \{a1 \mapsto b2,\ \ a3 \mapsto b4,\ \ a5 \mapsto b2,\ \ a7 \mapsto b6\} \\ \mathsf{dom}\,(F) &= \{a1,\ \ a3,\ \ a5,\ \ a7\} \\ \mathsf{ran}\,(F) &= \{b2,\ \ b4,\ \ b6\} \end{aligned}$$

A Total Function F from a Set A to a Set B



$$\text{dom}\,(F) \;=\; A$$

init
  $g :\in \mathbb{N} \leftrightarrow D$

final
  when
    $g = f$
  then
    skip
  end

- An anticipated event will be updated later and made convergent

progress
  status
    anticipated
  then
    $g :\in \mathbb{N} \leftrightarrow D$
  end

Development Approach

- Initial model: The file is transmitted in one shot (FUN1 and FUN2)

- First refinement: The file is transmitted gradually (FUN3)

- Second refinement: The two agents are separated

- Third refinement: Towards an implementation

First Refinement

- The observer comes closer to the future system

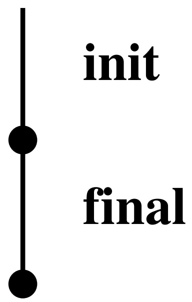
- So far he was just seeing the beginning and the end

- Now the observer will see some intermediate moves

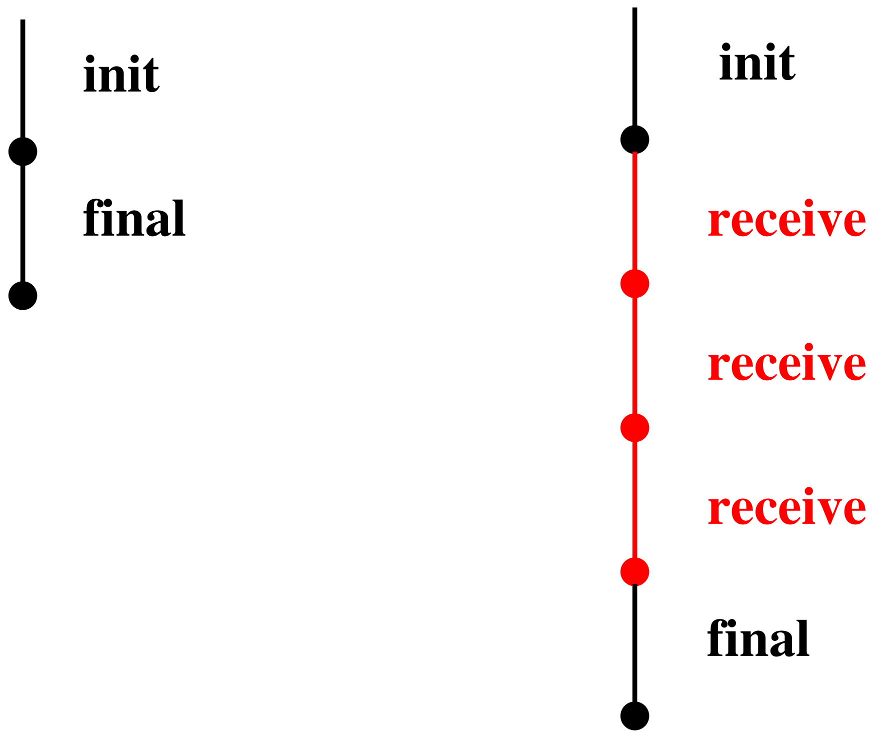
- He sees the file being gradually transfered from Sender to Receiver

- But he still has a partial view

What the Observer has Seen so far

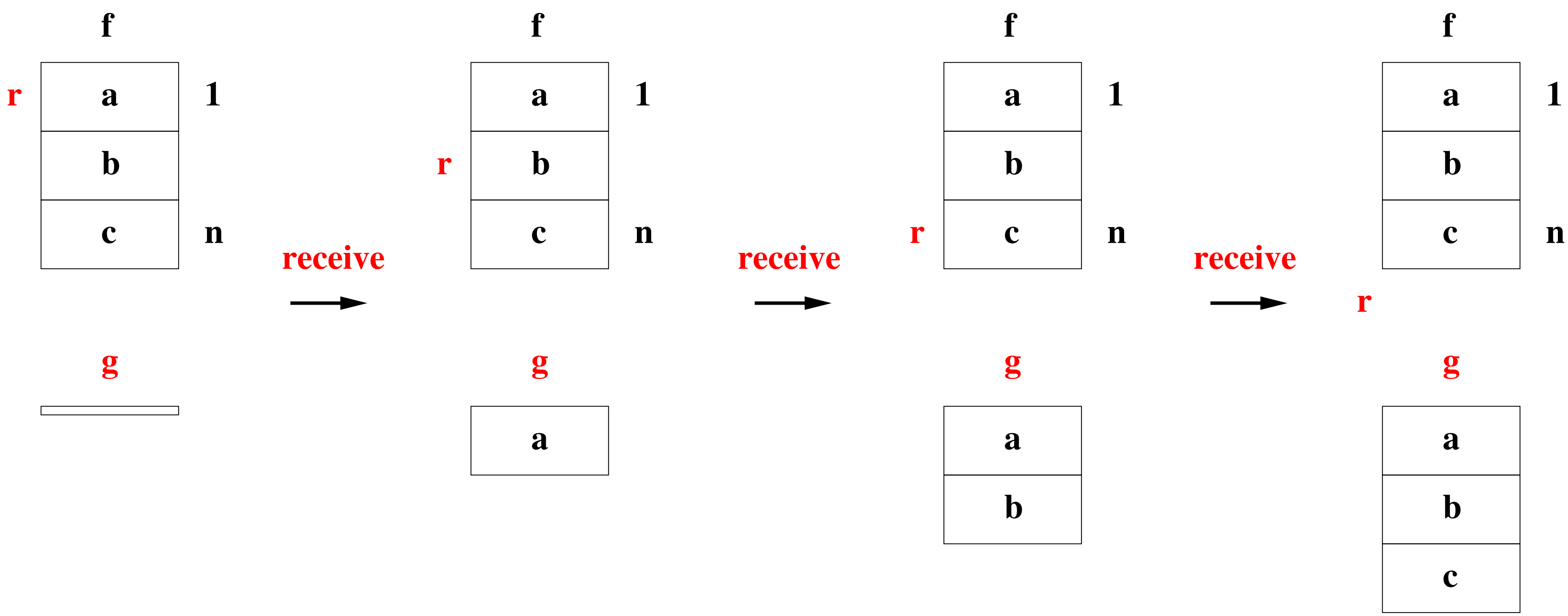


What the Observer will now See



A new event is introduced: receive

File transfer. Event receive



- The new variable r lies within the interval $1 \,..\, n+1$

- The variable g is equal to f restricted to its $r-1$ first values

First Refinement of the Protocol Model: the State

- Introducing additional variable r

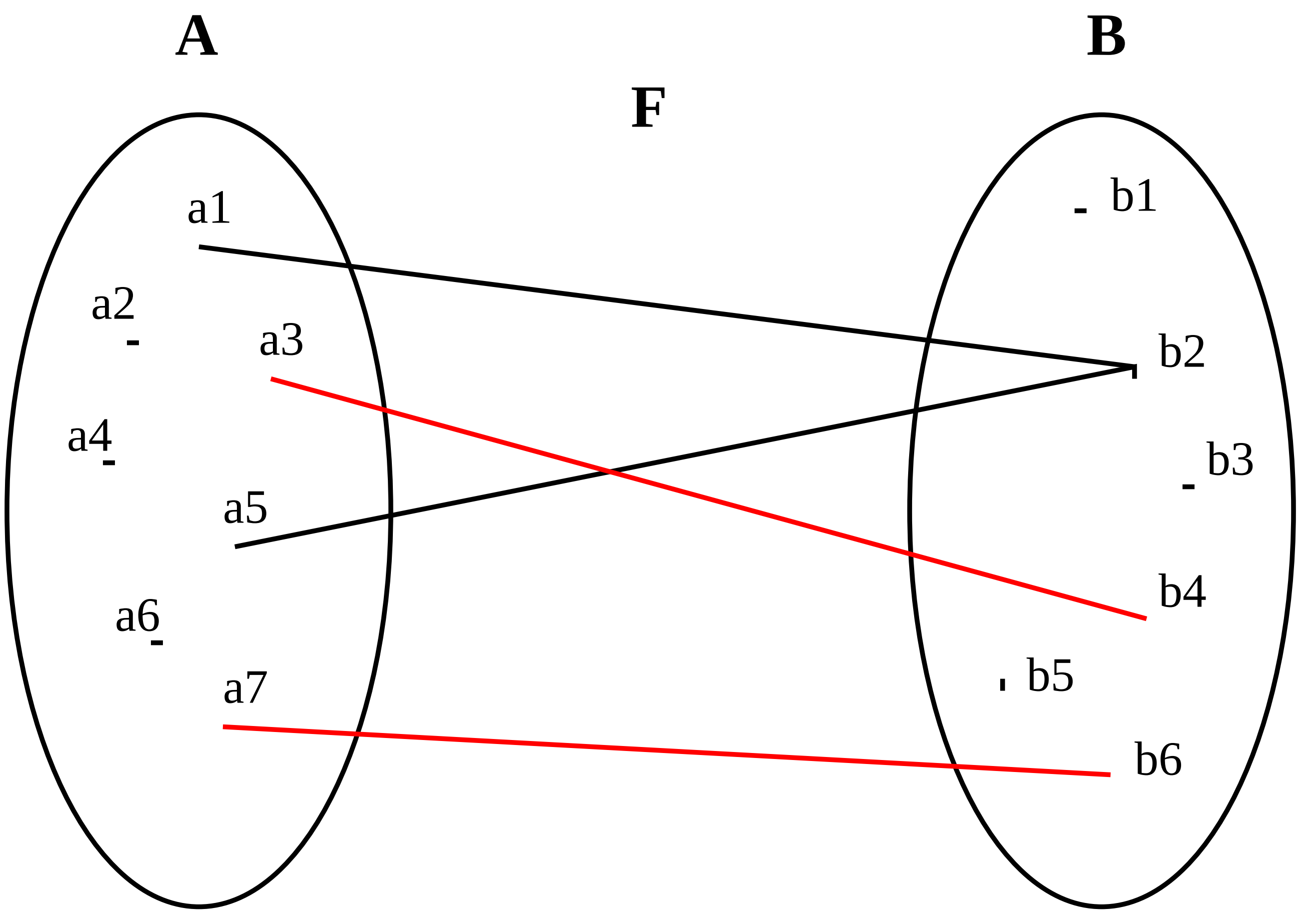
variables: g, r

inv1_1: $r \in 1 \,..\, n+1$

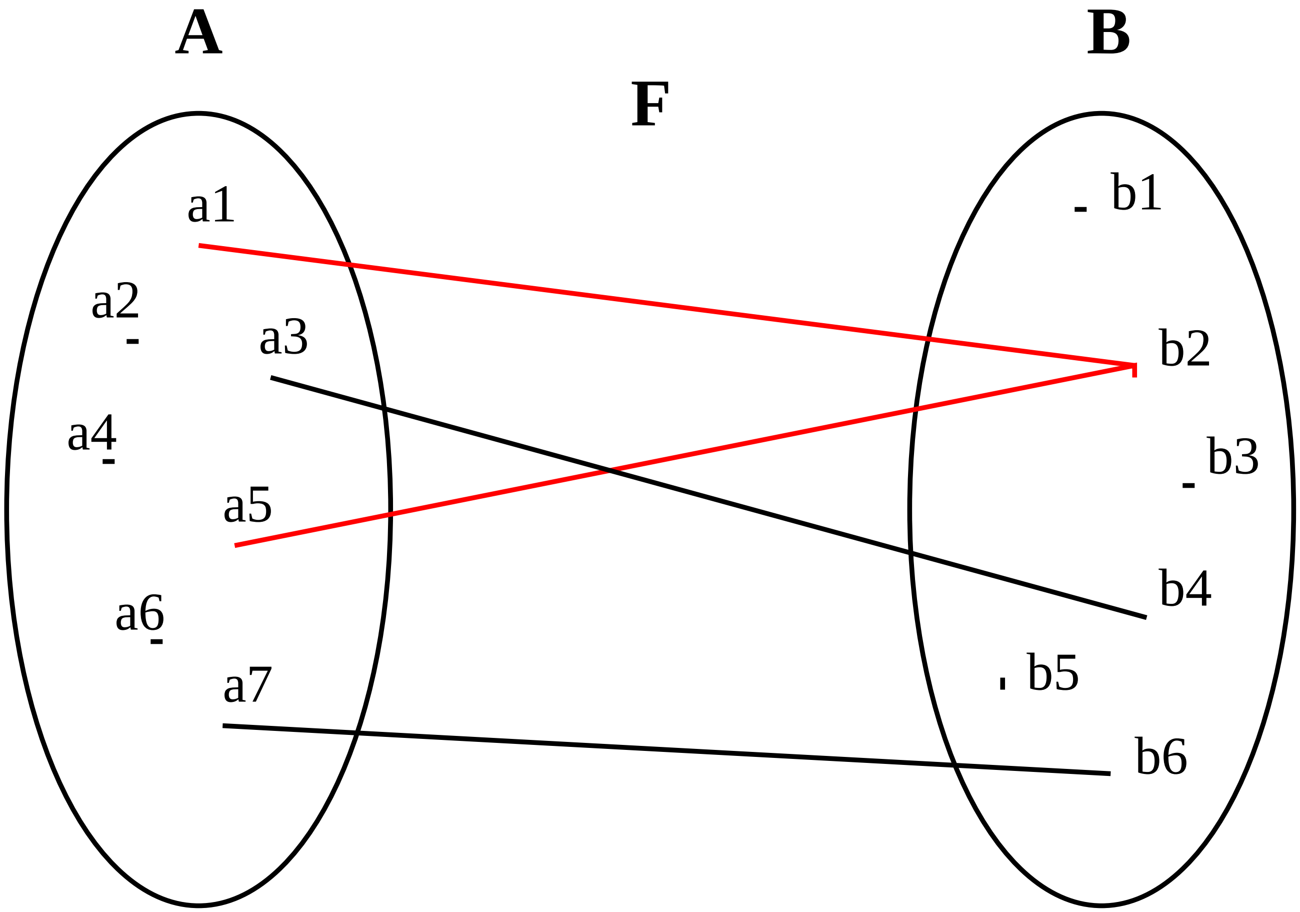
inv1_2: $g = (1 \,..\, r-1) \lhd f$

- g is defined to be the domain restriction of f to $1 \,..\, r-1$

$s \lhd r$	domain restriction operator
$s \ntriangleleft r$	domain subtraction operator
$r \rhd t$	range restriction operator
$r \ntriangleright t$	range subtraction operator

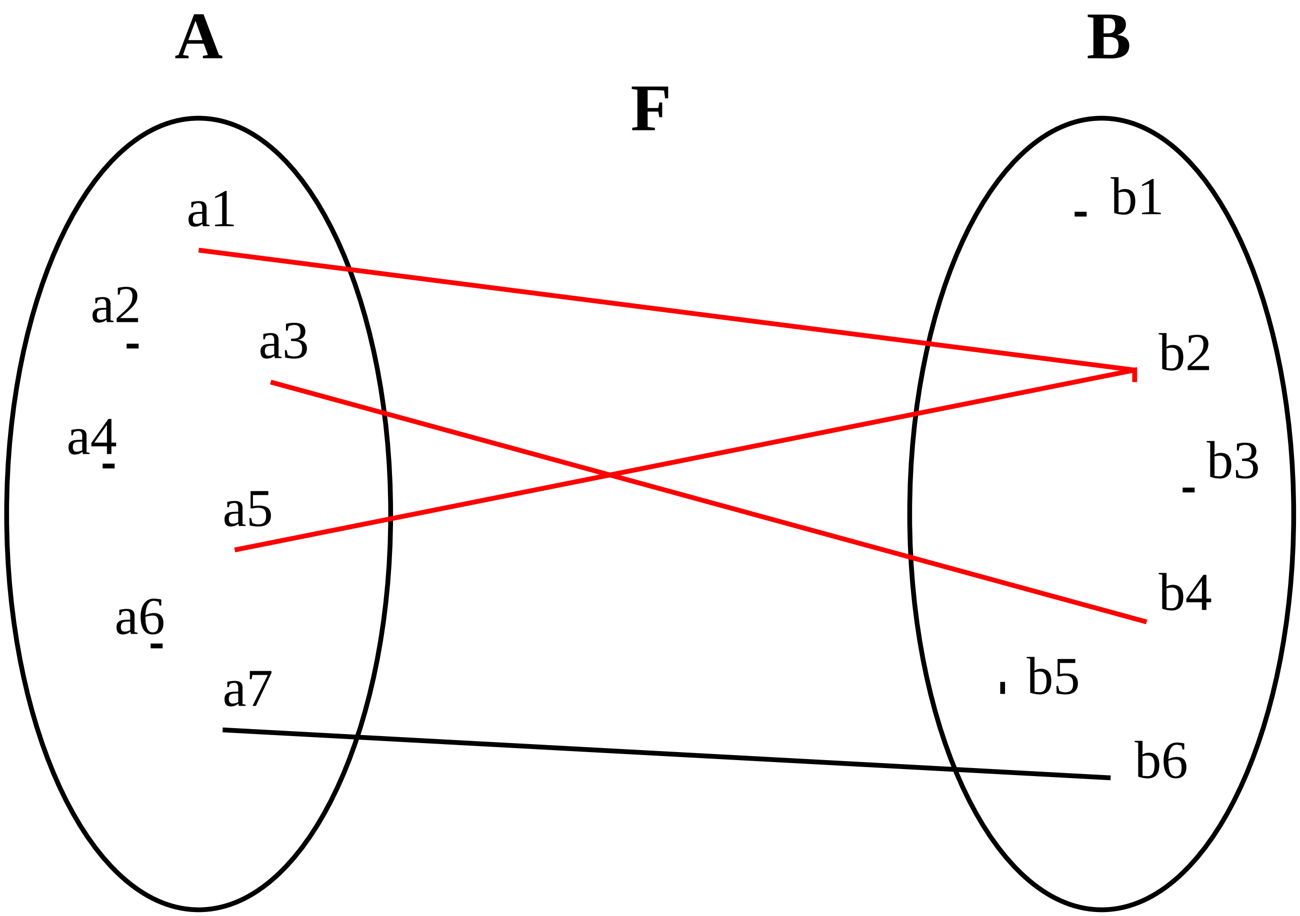


$$\{a3,\ a7\} \lhd F$$

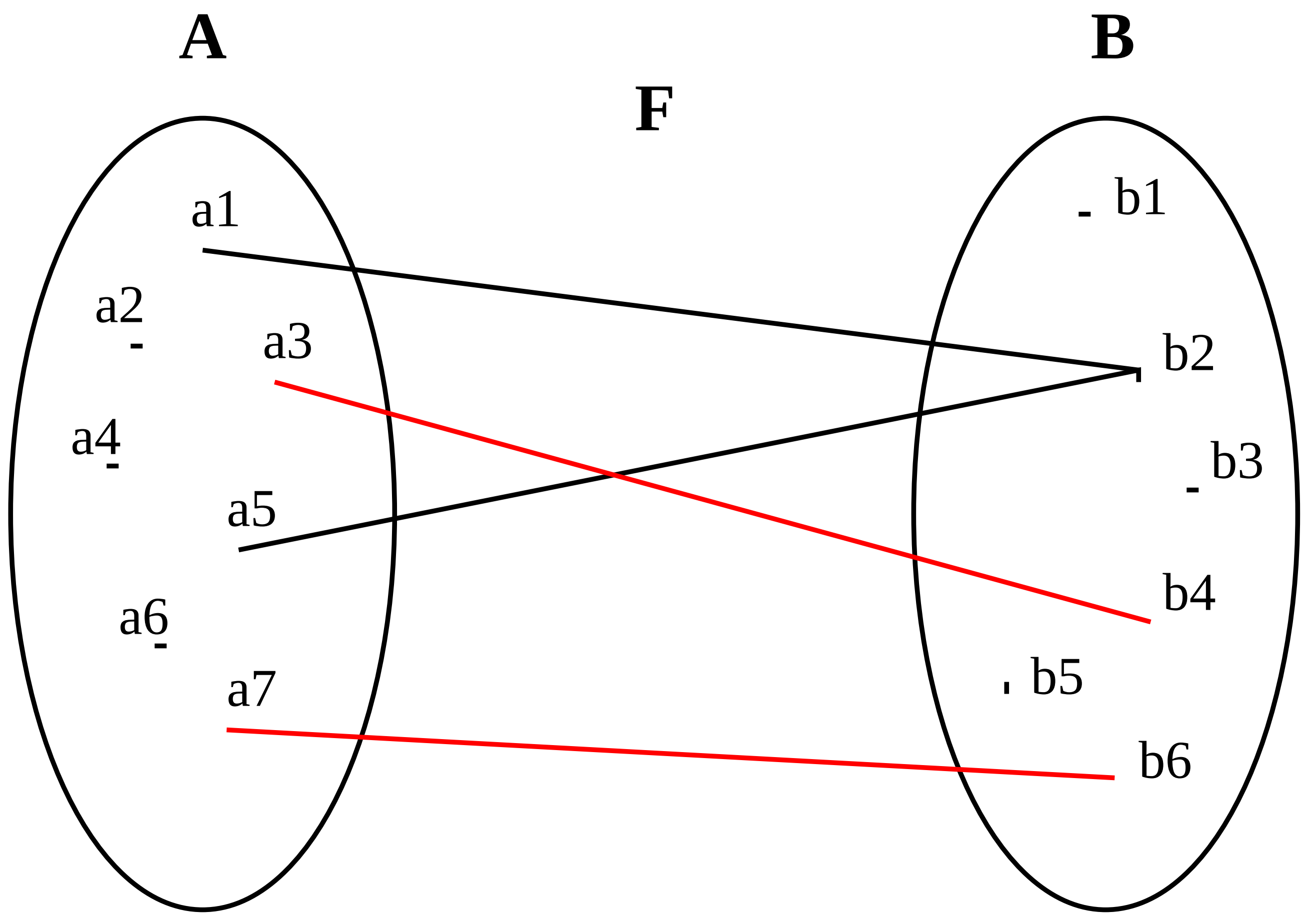


$$\{a3,\ a7\} \lhd F$$

The Range Restriction Operator



$$F \rhd \{b2, b4\}$$



$$F \mathrel{\ominus\!\!\!\!\triangleright}\{b2\}$$

init
$\quad g := \varnothing$
$\quad r := 1$

receive
$\quad$ **refines**
$\quad\quad$ **progress**
$\quad$ **refines**
$\quad\quad$ **convergent**
$\quad$ **when**
$\quad\quad r \leq n$
$\quad$ **then**
$\quad\quad h := h \cup \{r \mapsto f(r)\}$
$\quad\quad r := r + 1$
$\quad$ **end**

final
$\quad$ **when**
$\quad\quad r = n + 1$
$\quad$ **then**
$\quad\quad$ skip
$\quad$ **end**

- The variant is decreased by the convergent event

variant1: $n + 1 - r$

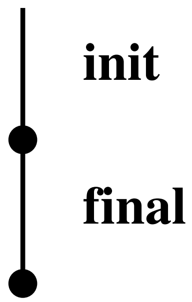
Development Approach

- Initial model: The file is transmitted in one shot (FUN1 and FUN2)

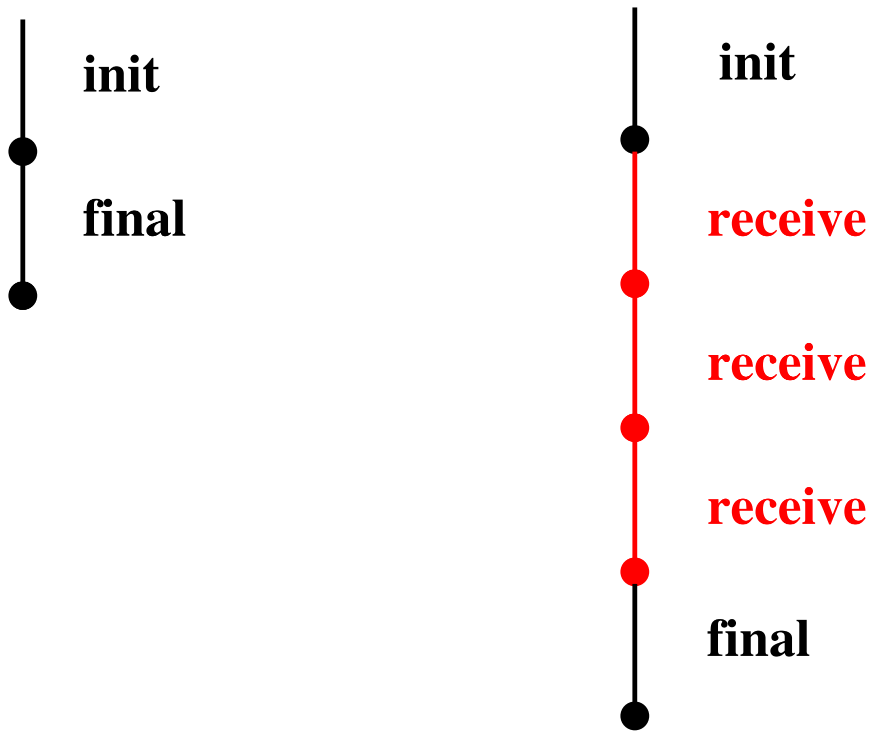
- First refinement: The file is transmitted gradually (FUN3)

- Second refinement: The two agents are separated

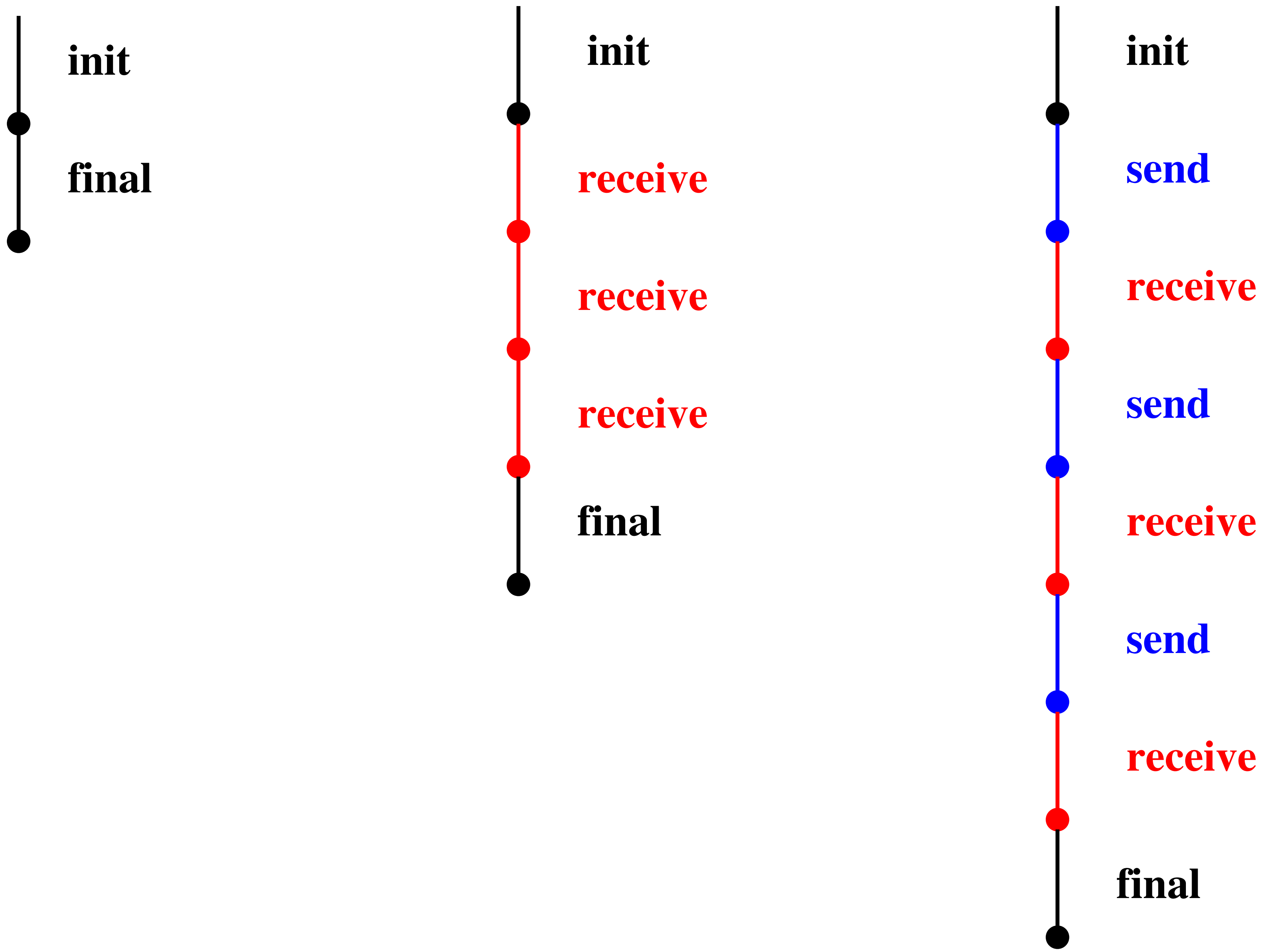
- Third refinement: Towards an implementation


init
final

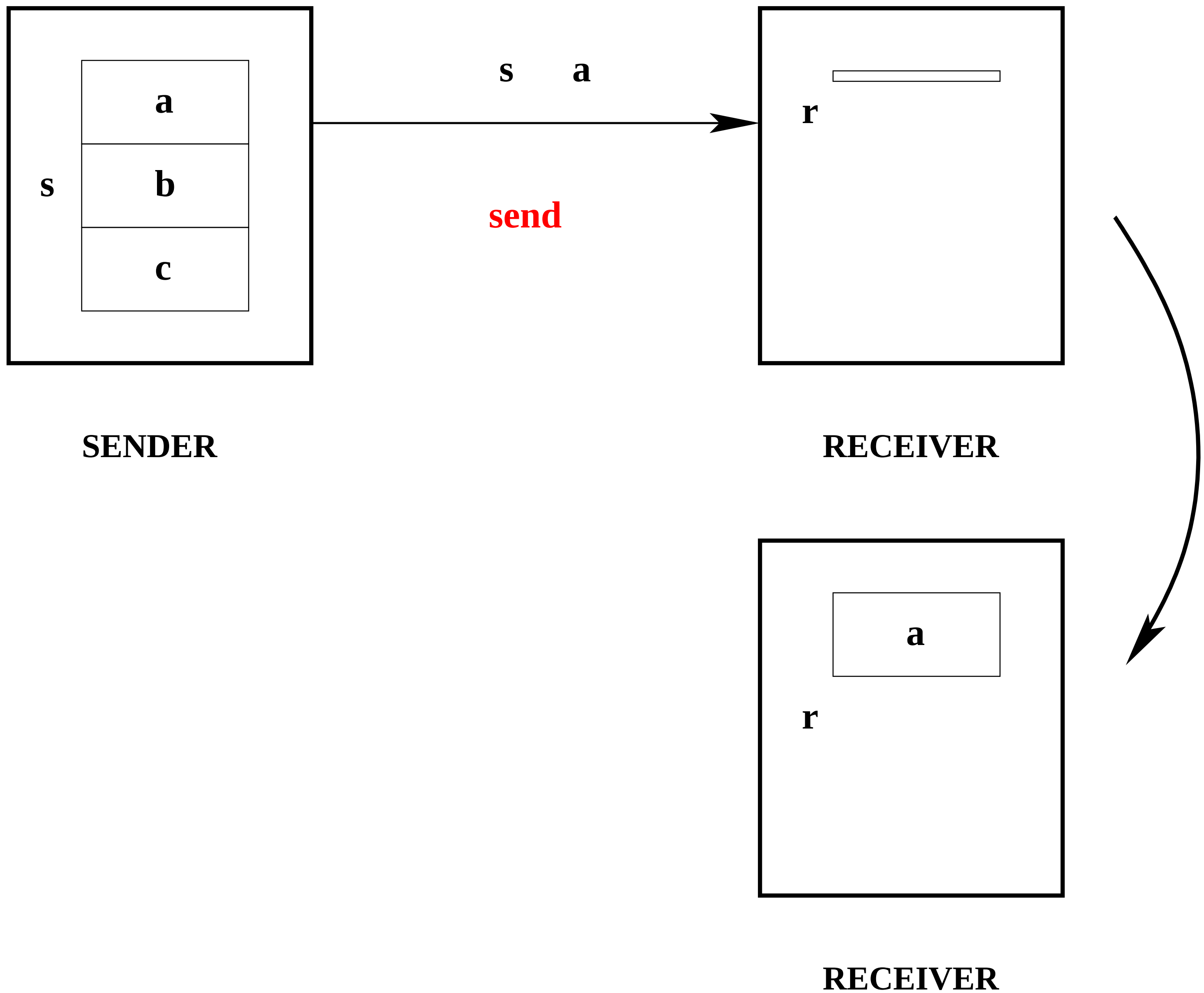
What the Observer could see of first Refinement



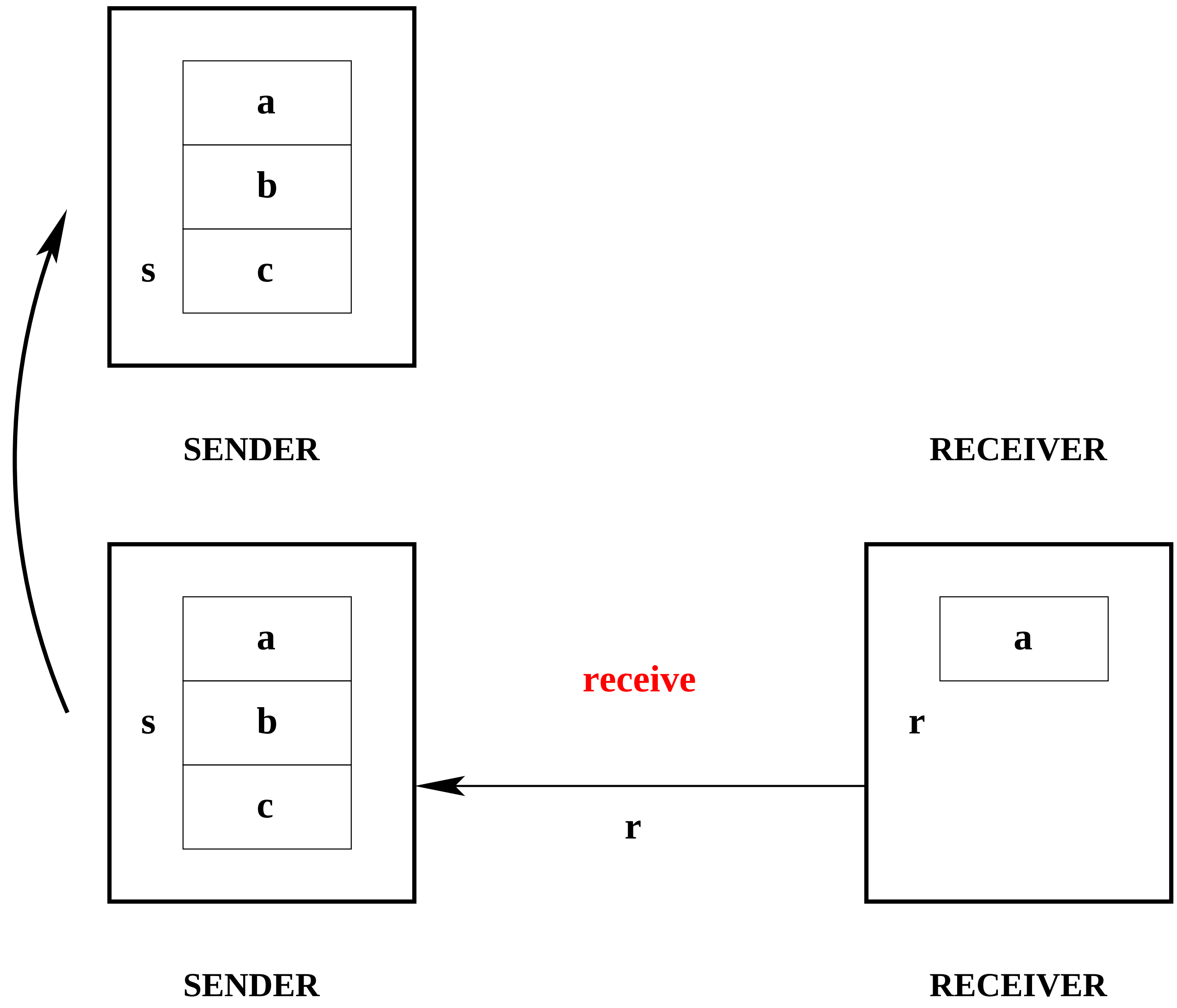
What the Observer will now See



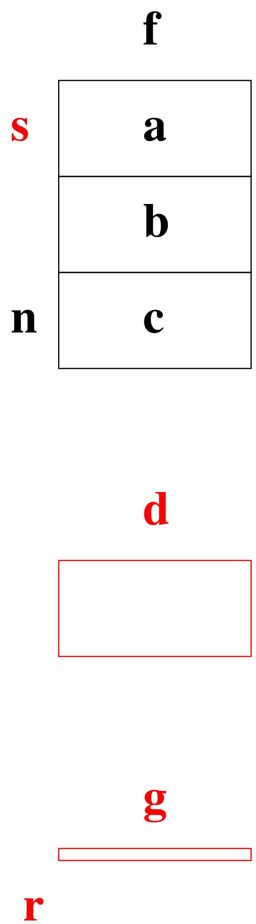
A More Accurate Version (1)

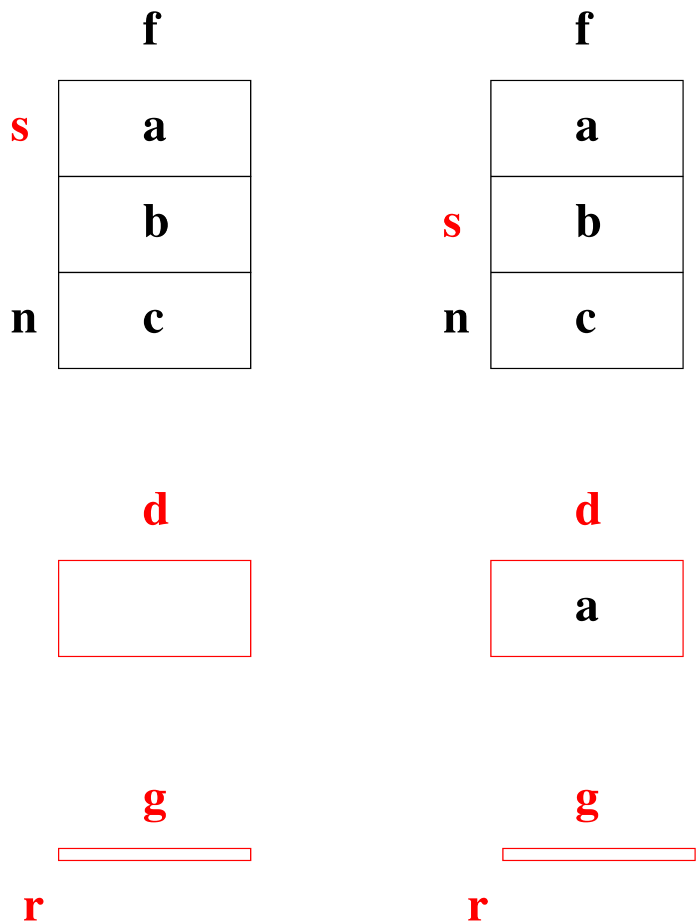


A More Accurate Version (2)

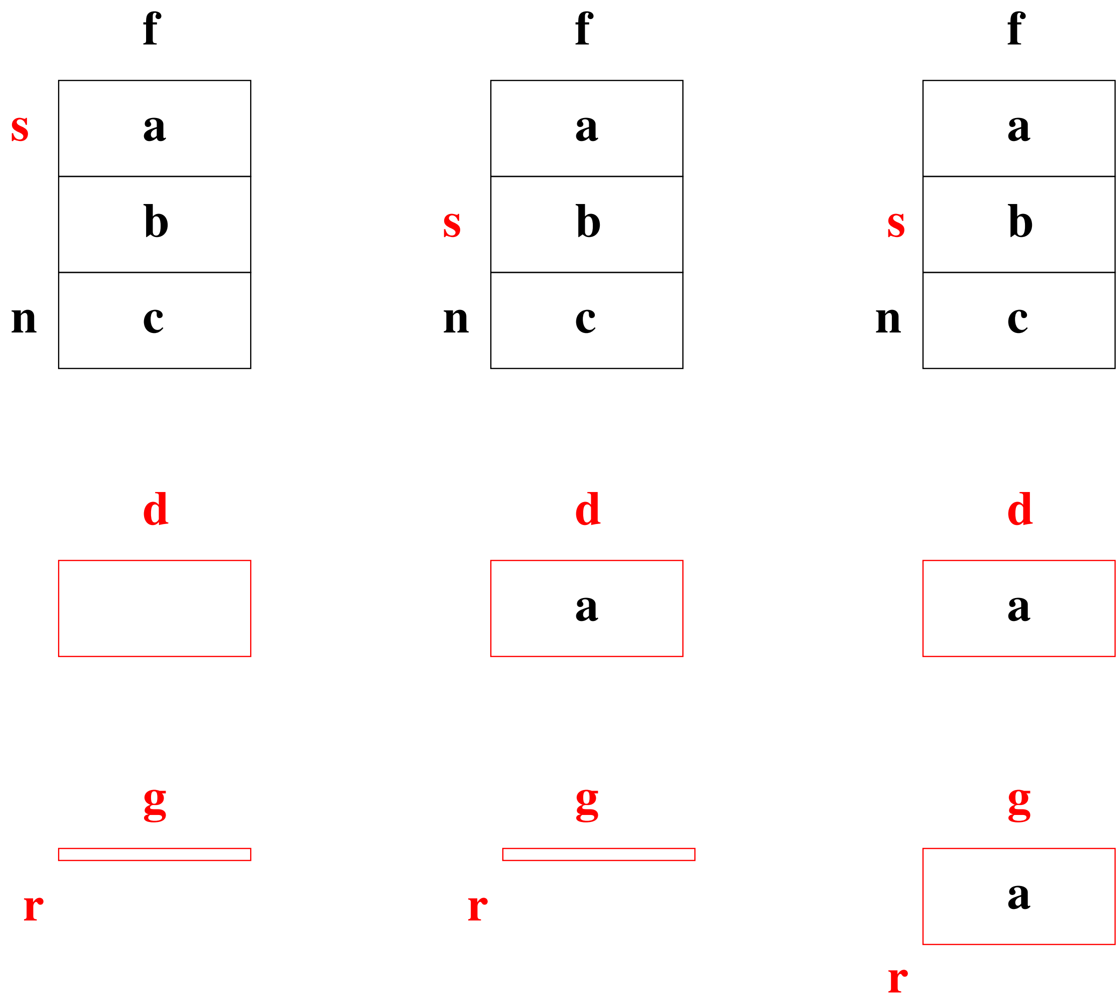


Initial Situation

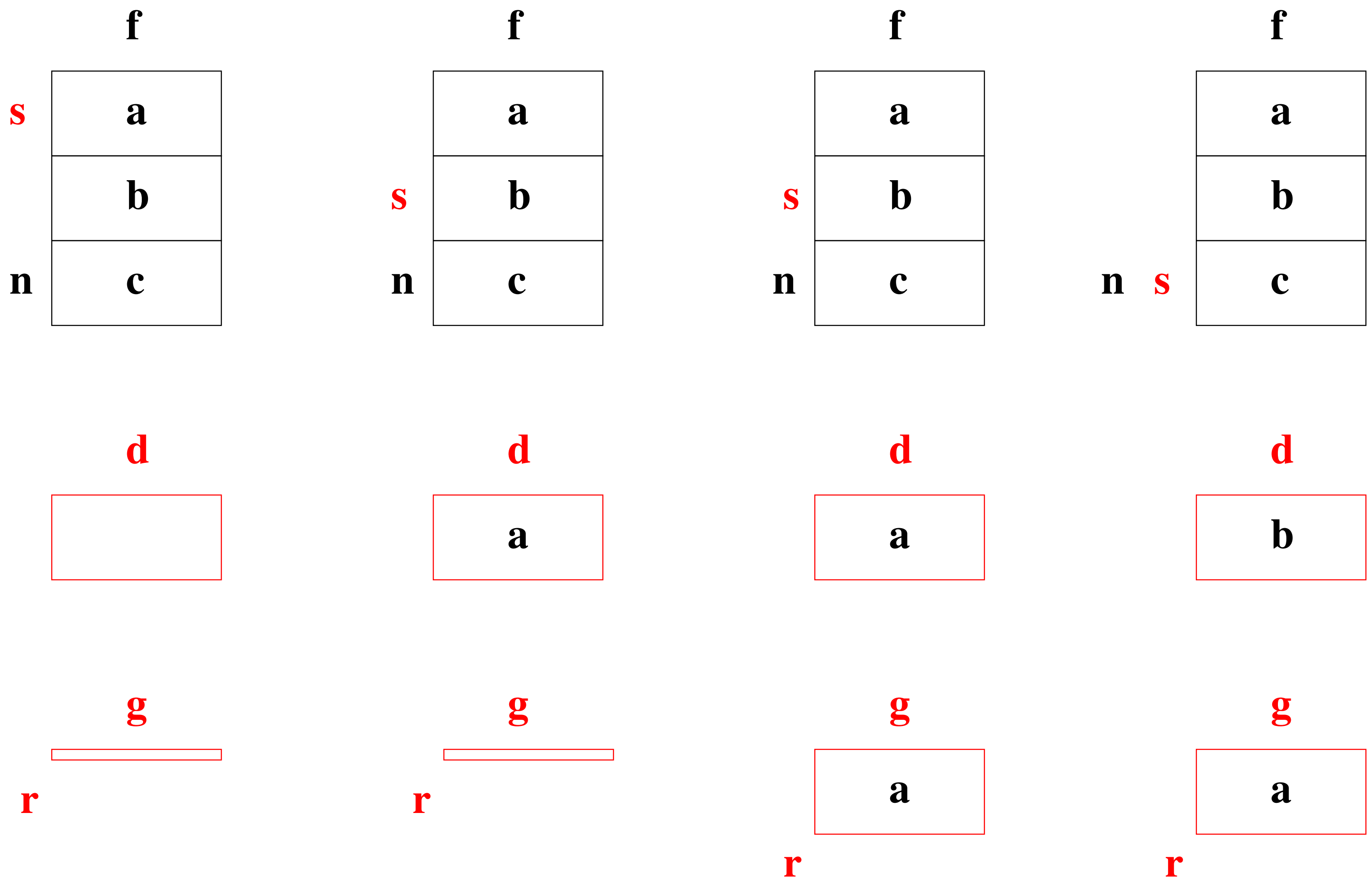



f
s
a
b
n
c
d
g
r
f
a
s
b
n
c
d
a
g
r

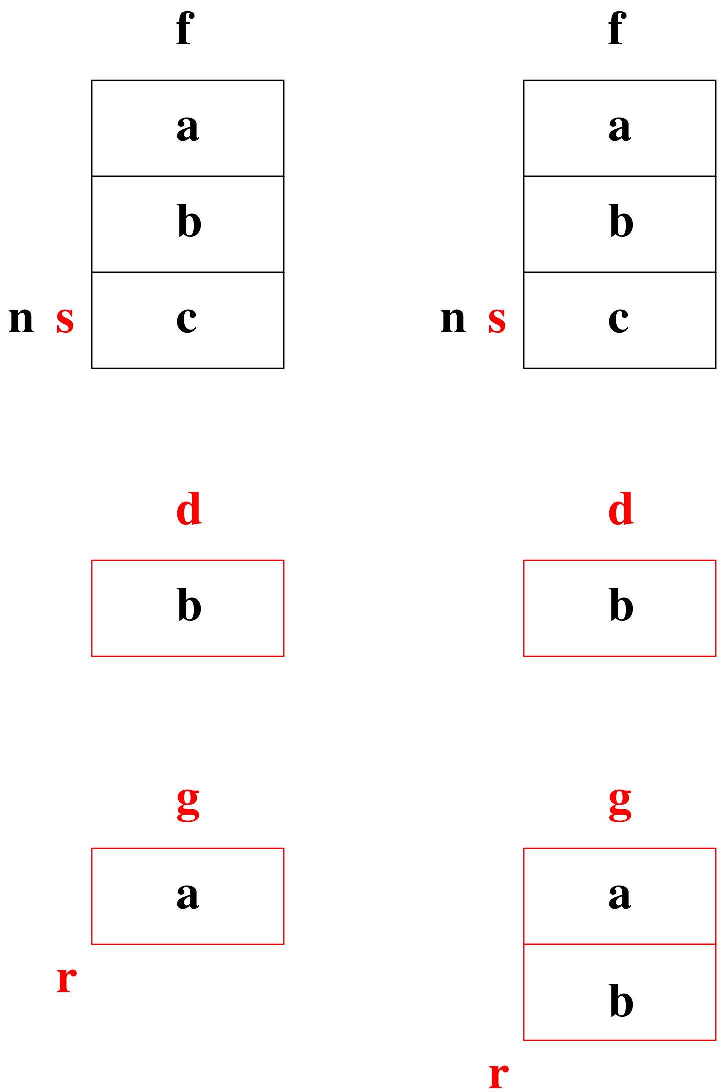
Receive

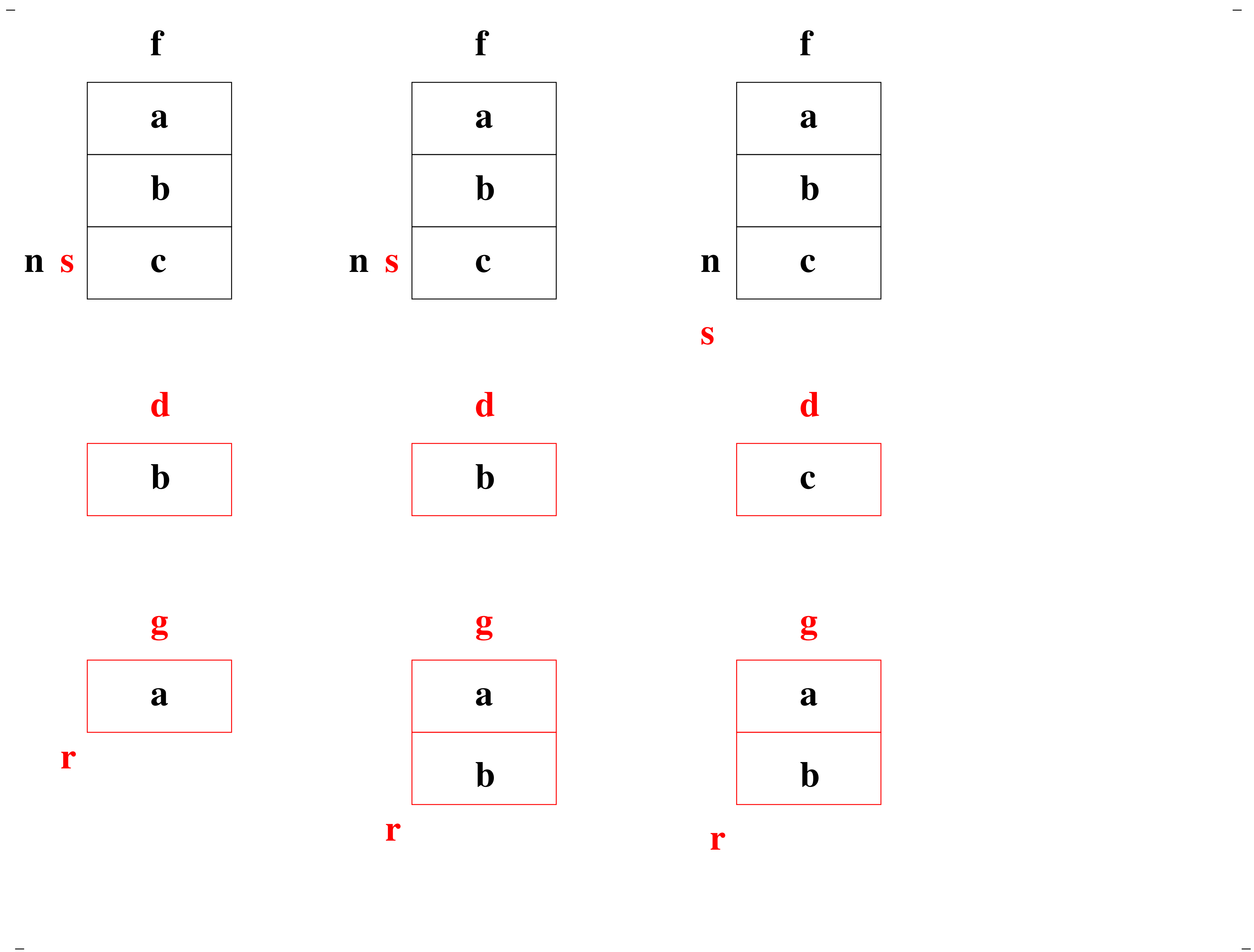


Send

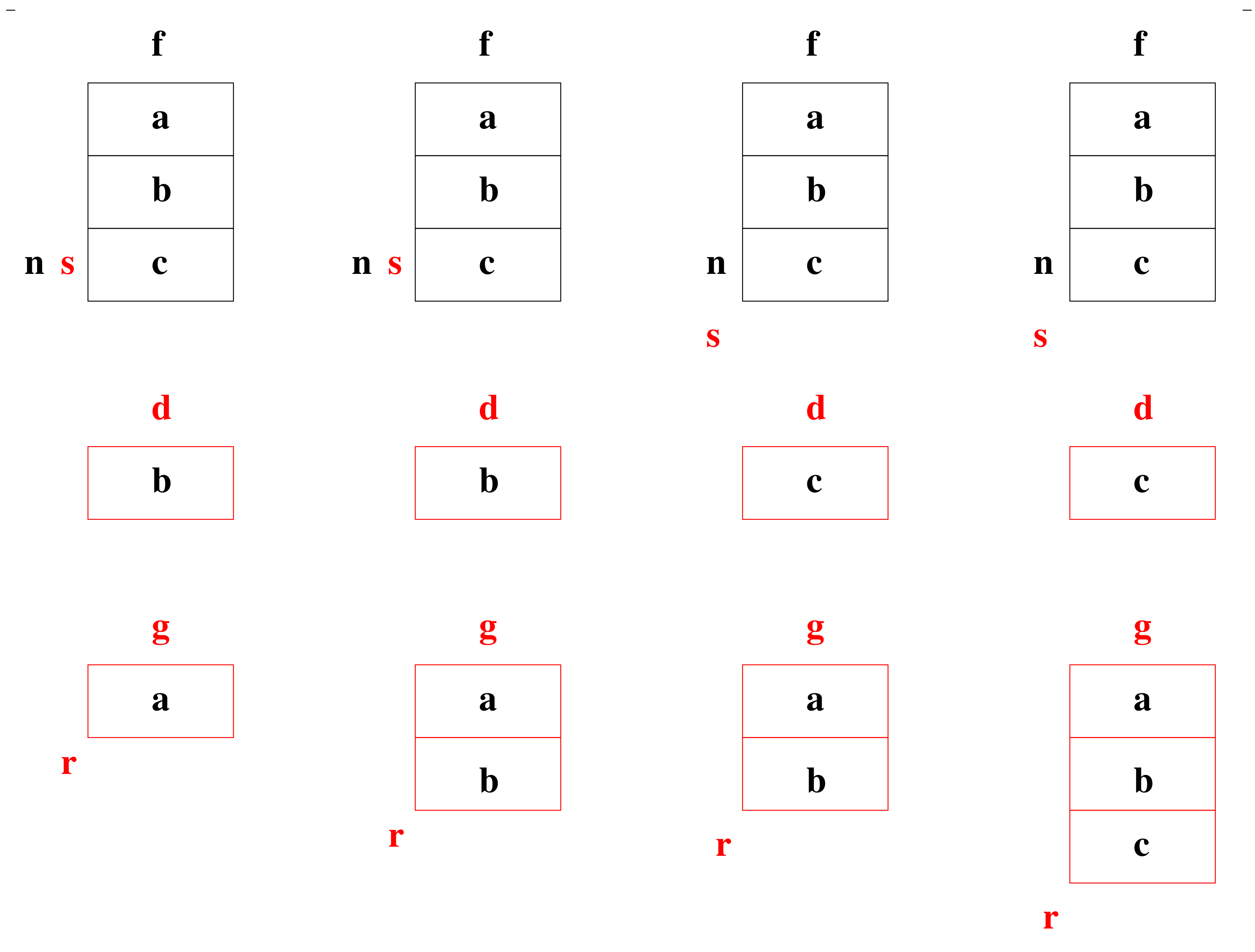


Receive

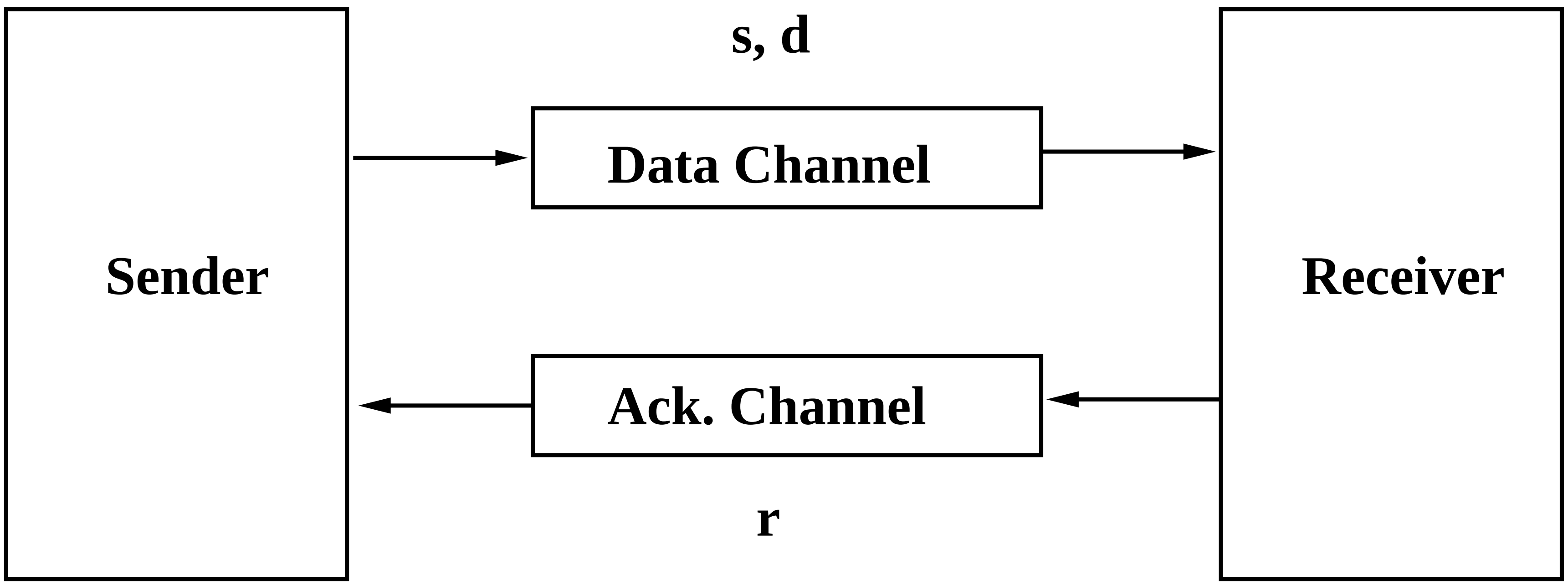



f f f
a a a
b b b
n s c
n s c
n c
s
d d d
b b c
g g g
a a a
r
b b
r r

Receive



Sender and Receiver



- We introduce an additional variable s, and a data item d

carrier sets: D

constants: $n, f, d0$

variables: g, r, s, d

inv2_1: $s \in 1 \,..\, n+1$

inv2_2: $s \in r \,..\, r+1$

inv2_3: $d \in D$

inv2_4: $s = r+1 \;\Rightarrow\; d = f(r)$

axm2_1: $d0 \in D$

Second Refinement: the Events

init
$g := \varnothing$
$s := 1$
$r := 1$
$d := d0$

send
when
$s = r$
$s \neq n + 1$
then
$d, s := f(s), s + 1$
end

receive
when
$s = r + 1$
then
$h := h \cup \{r \mapsto d\}$
$r := r + 1$
end

final
when
$r = n + 1$
then
skip
end

- Initial model: The file is transmitted in one shot (FUN1 and FUN2)

- First refinement: The file is transmitted gradually (FUN3)

- Second refinement: The two agents are separated

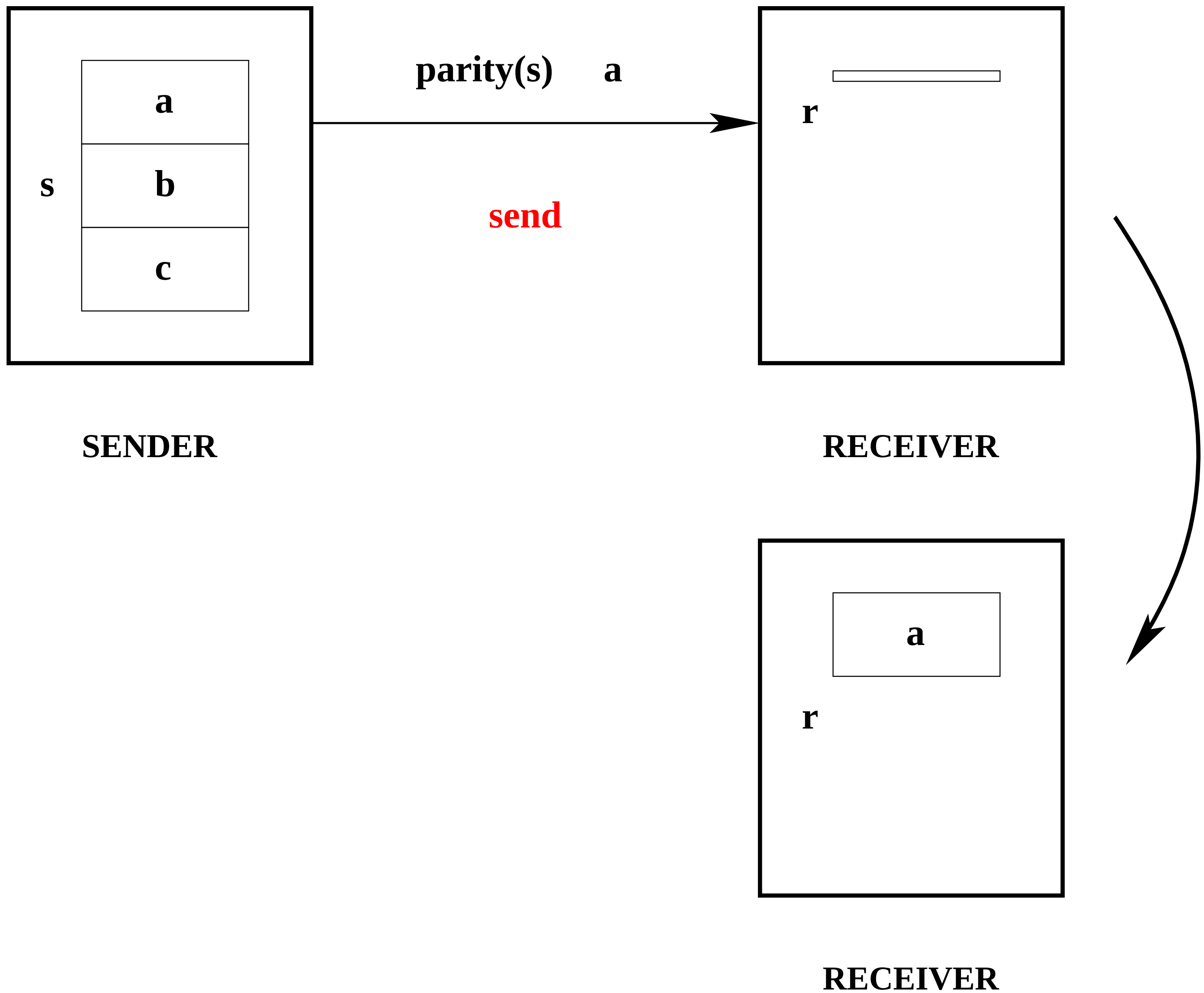
- Third refinement: Towards an implementation

send
when
$s = r$
$s \neq n+1$
then
$d := f(s)$
$s := s+1$
end

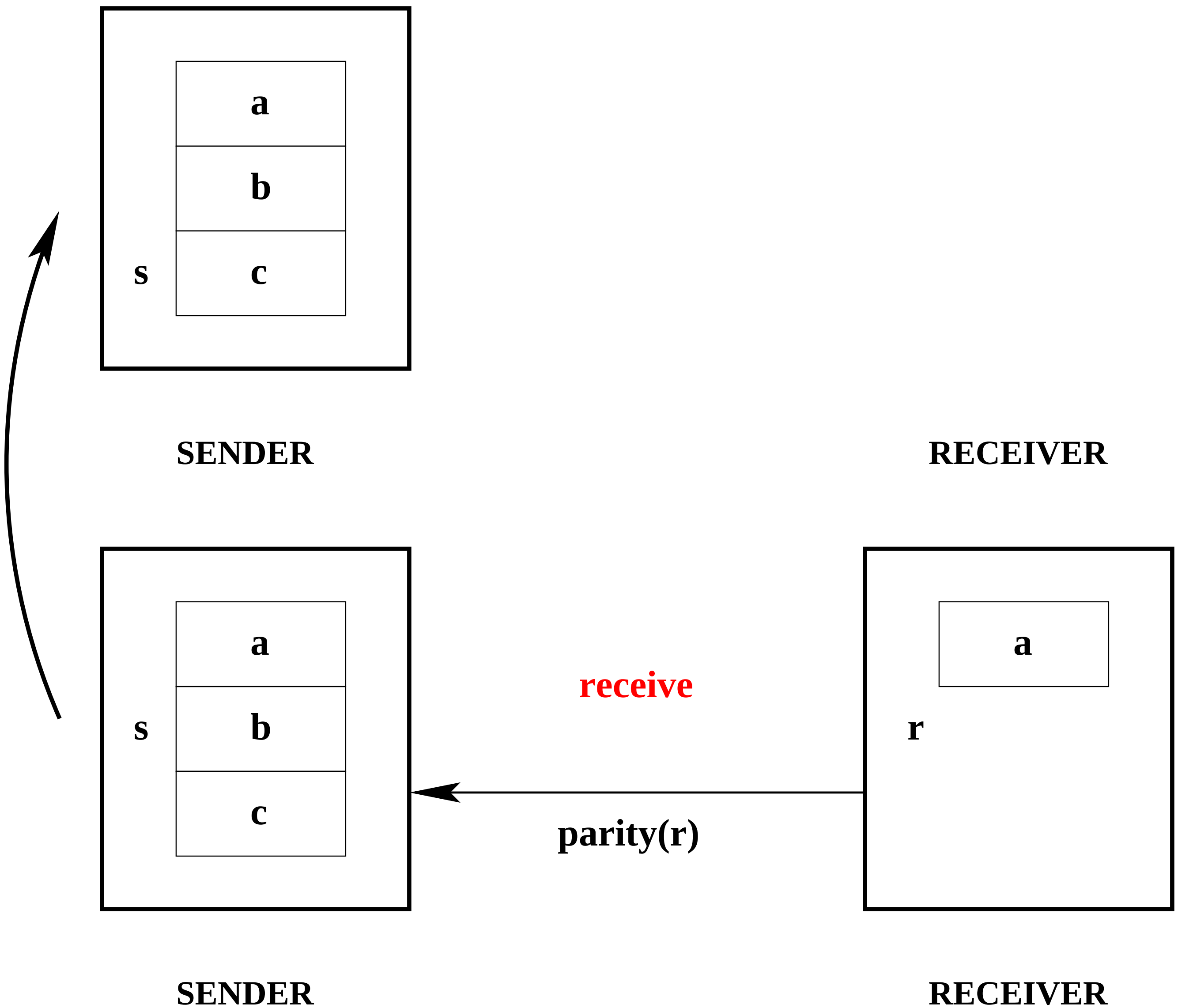
receive
when
$s = r+1$
then
$g := g \;\cup\; \{r \mapsto d\}$
$r := r+1$
end

inv2_2: $s \;\in\; r \,..\, r+1$

Third Refinement: Sending Parities (1)



Third Refinement: Sending Parities (2)



axm3_1: $parity \in \mathbb{N} \rightarrow \{0, 1\}$

axm3_2: $parity(0) = 0$

axm3_3: $\forall x \cdot (\, x \in \mathbb{N} \;\Rightarrow\; parity(x + 1) = 1 - parity(x) \,)$

thm3_1: $$\forall x, y \cdot \left(\begin{array}{l} \;x \in \mathbb{N} \\ \;y \in \mathbb{N} \\ \;x \in y \,..\, y + 1 \\ \;parity(x) = parity(y) \\ \Rightarrow \\ \;x = y \end{array} \right)$$

Third Refinement: the State

carrier sets: D

constants: $n, f, parity$

variables: g, s, r, d, p, q

inv3_1: $p = parity(s)$

inv3_2: $q = parity(r)$

axm3_1: $parity \in \mathbb{N} \rightarrow \{0, 1\}$

axm3_2: $parity(0) = 0$

axm3_3: $\forall x \cdot \left(\begin{array}{l} x \in \mathbb{N} \\ \Rightarrow \\ parity(x+1) = 1 - parity(x) \end{array} \right)$

init
$$g := \varnothing$$
$$s := 1$$
$$r := 1$$
$$p := 1$$
$$q := 1$$
$$d := d0$$

final
when
$$r = n + 1$$
then
skip
end

send
when
$$p = q$$
$$s \neq n + 1$$
then
$$d := f(s)$$
$$s := s + 1$$
$$p := 1 - p$$
end

receive
when
$$p \neq q$$
then
$$g := g \cup \{r \mapsto d\}$$
$$r := r + 1$$
$$q := 1 - q$$
end

What we Have Learned in this Lecture

- More mathematical conventions

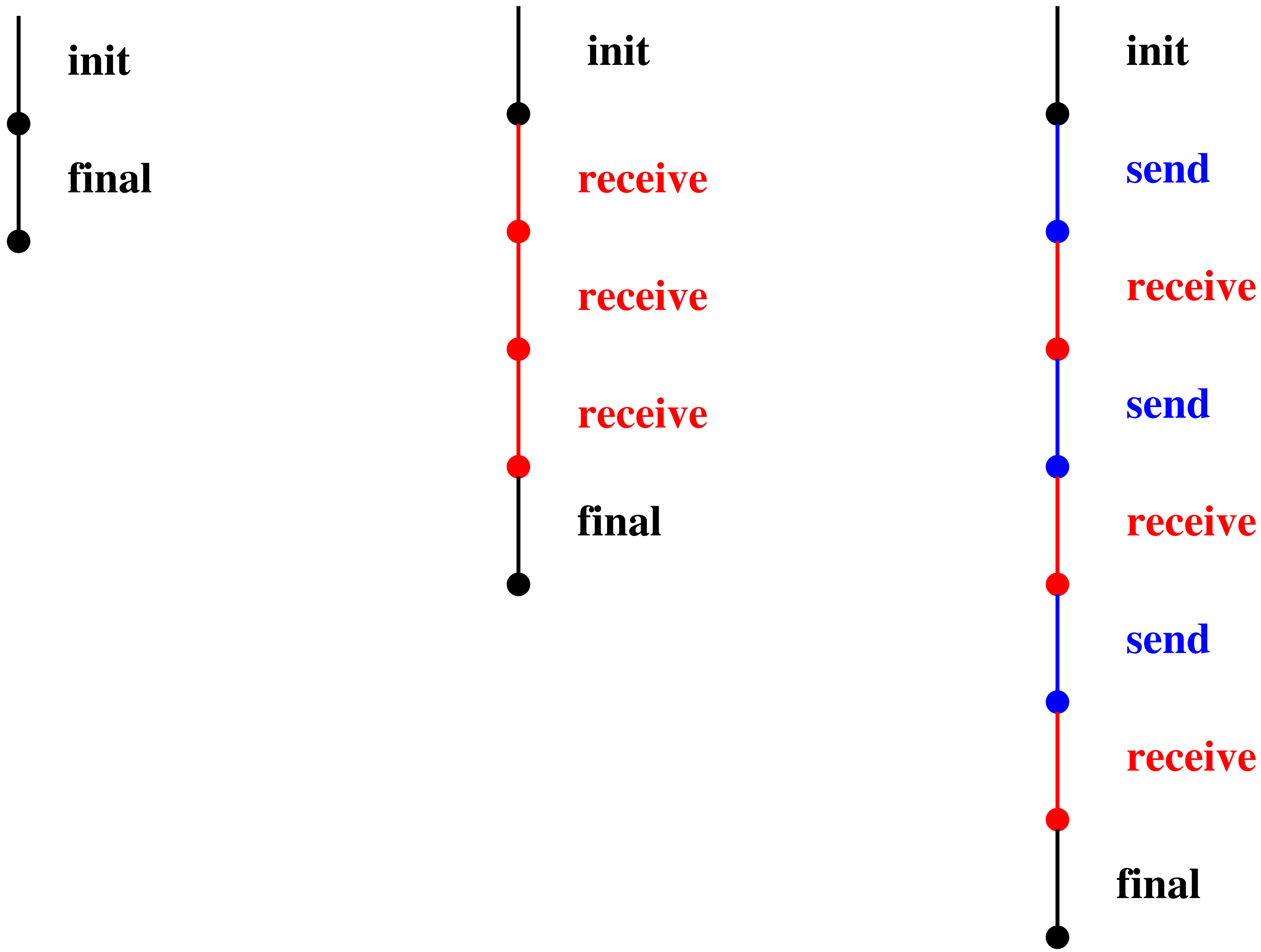
- How to write a model

- What kind of things we have to prove

- How the proof can help finding invariants

- Many things can be done by tools

- A small theory of parities


init
final
init
receive
receive
receive
final
init
send
receive
send
receive
send
receive
final

Reminder of Mathematical Conventions (1)

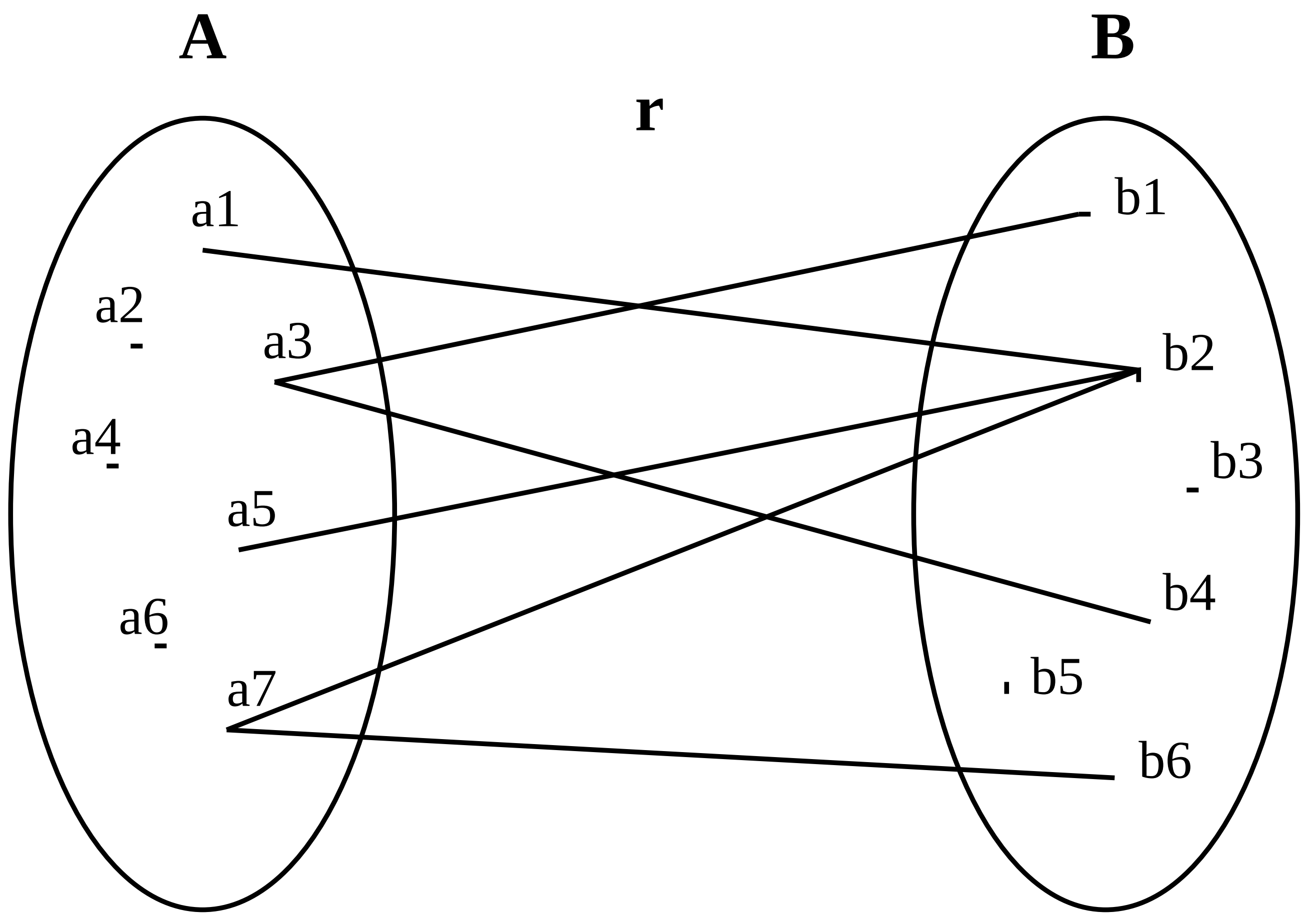
$x \in S$	Set membership operator
$\mathbb{N}$	set of Natural Numbers: $\{0, 1, 2, 3, \ldots\}$
$a \mathinner{..} b$	Interval from a to b: $\{a, a+1, \ldots, b\}$ (empty when $b < a$)
$a \mapsto b$	pair constructing operator
$S \times T$	Cartesian product operator
$S \subseteq T$	set inclusion operator
$\mathbb{P}(S)$	power set operator

$S \leftrightarrow T$	Set of binary relations from S to T
$S \rightarrow T$	Set of total functions from S to T
$S \nrightarrow T$	Set of partial functions from S to T
$\mathrm{dom}(r)$	Domain of a relation r
$\mathrm{ran}(r)$	Range of a relation r

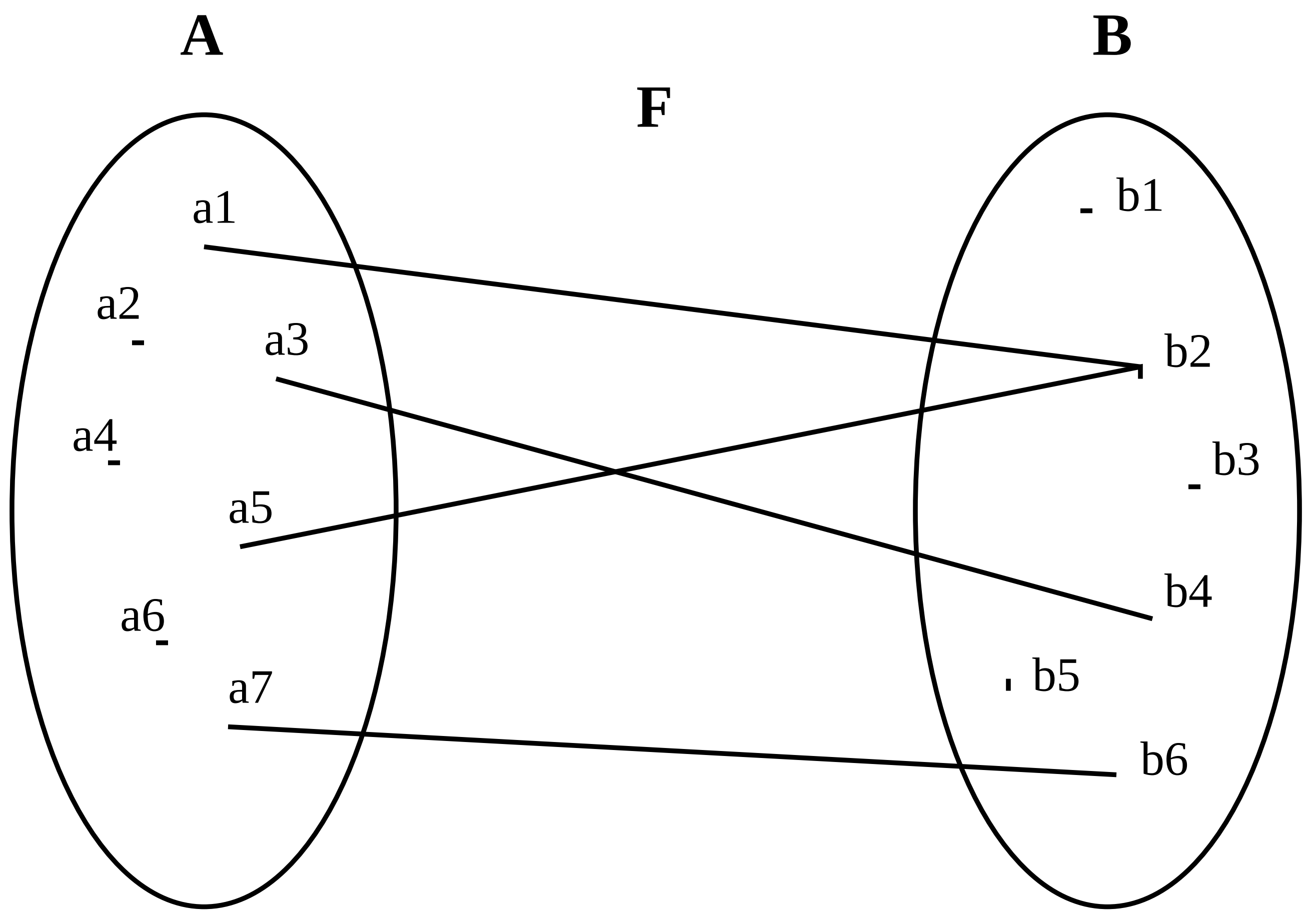
Reminder of Mathematical Conventions (3)

$s \lhd r$	domain restriction operator
$s \ntriangleleft r$	domain subtraction operator
$r \rhd t$	range restriction operator
$r \ntriangleright t$	range subtraction operator

A Binary Relation r from a Set A to a Set B



A Partial Function F from a Set A to a Set B

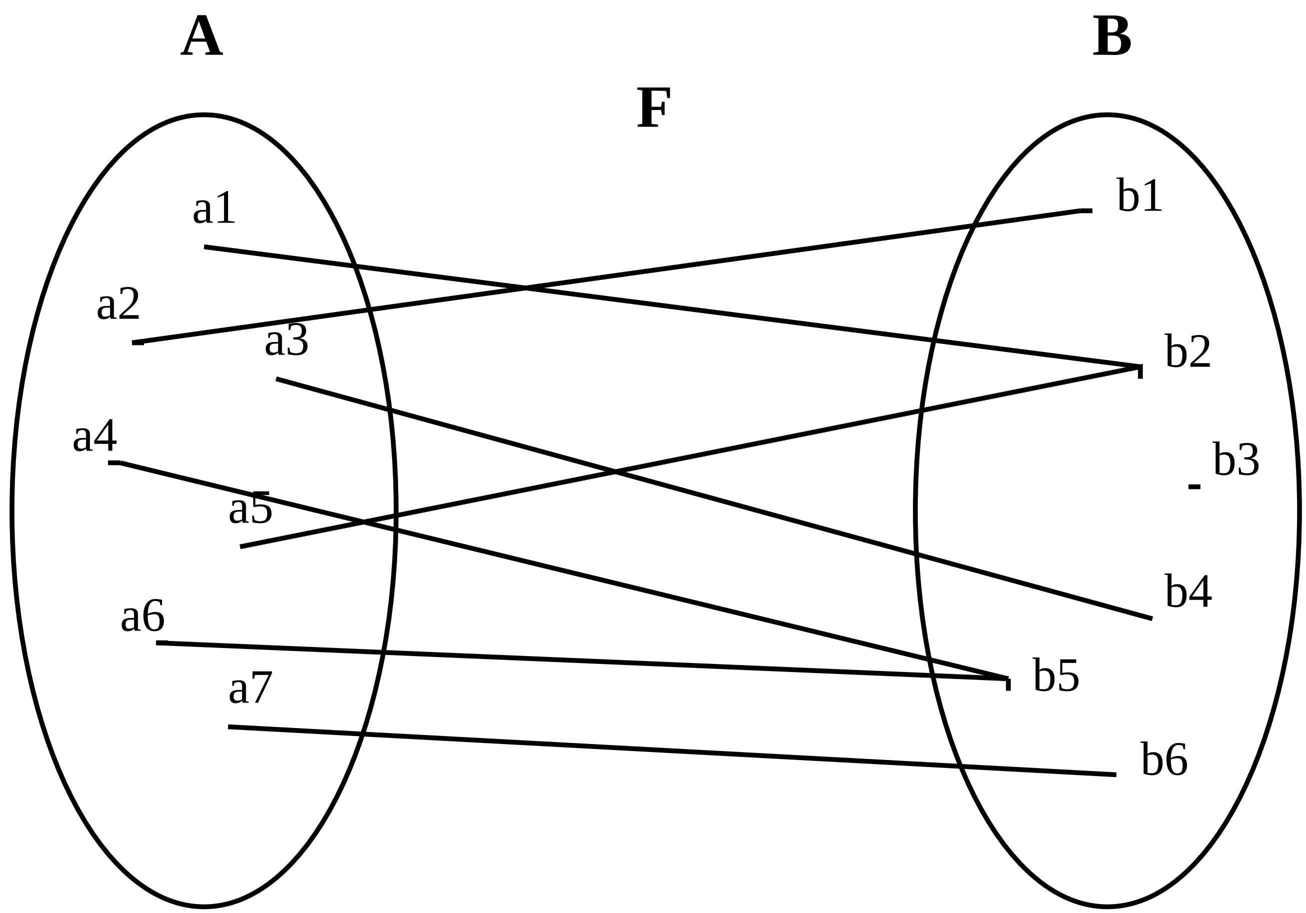


$$F = \{a1 \mapsto b2,\ a3 \mapsto b4,\ a5 \mapsto b2,\ a7 \mapsto b6\}$$

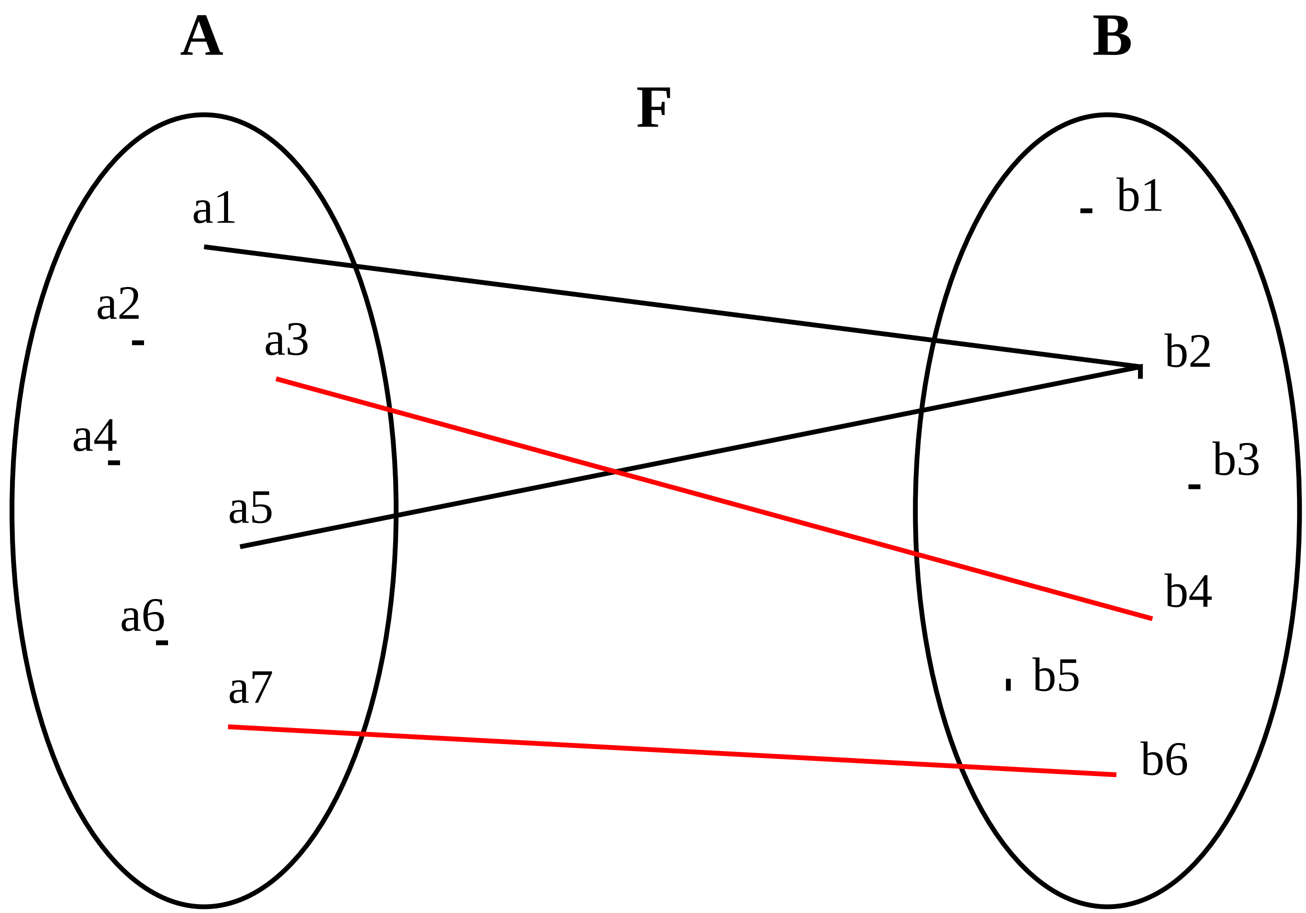
$$\mathrm{dom}\,(F) = \{a1,\ a3,\ a5,\ a7\}$$

$$\mathrm{ran}\,(F) = \{b2,\ b4,\ b6\}$$

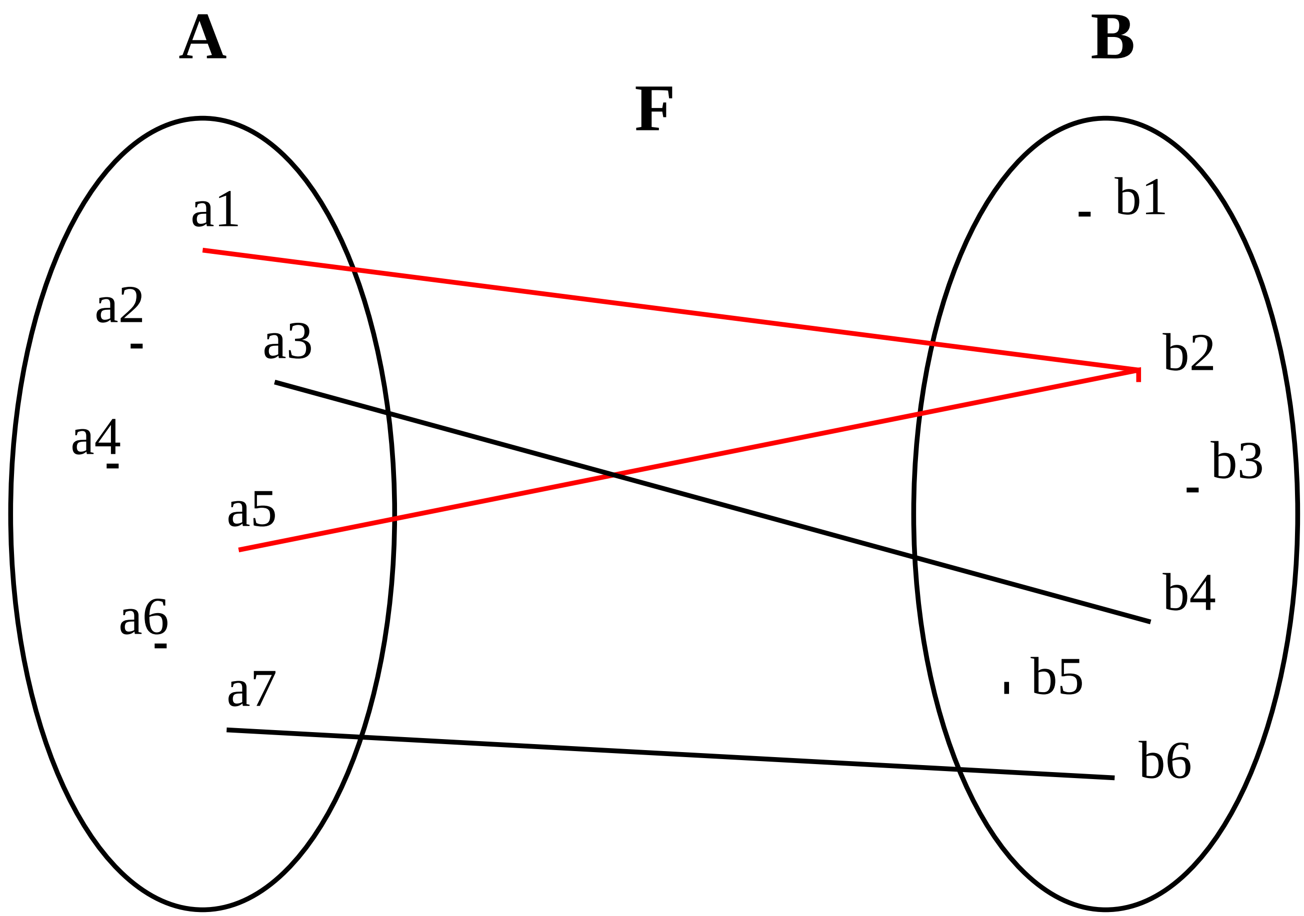
A Total Function F from a Set A to a Set B



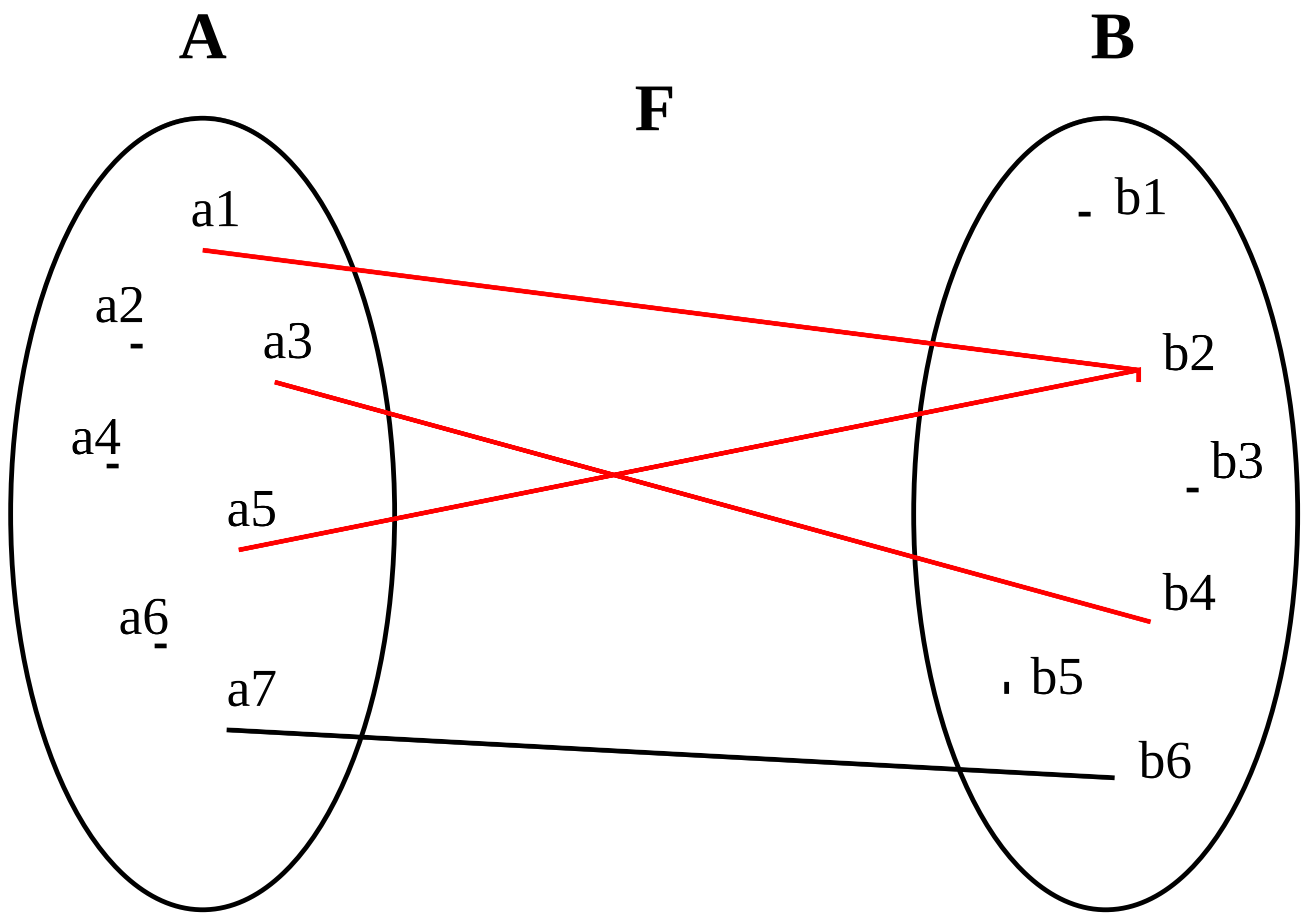
$$\mathsf{dom}\,(F) \;=\; A$$



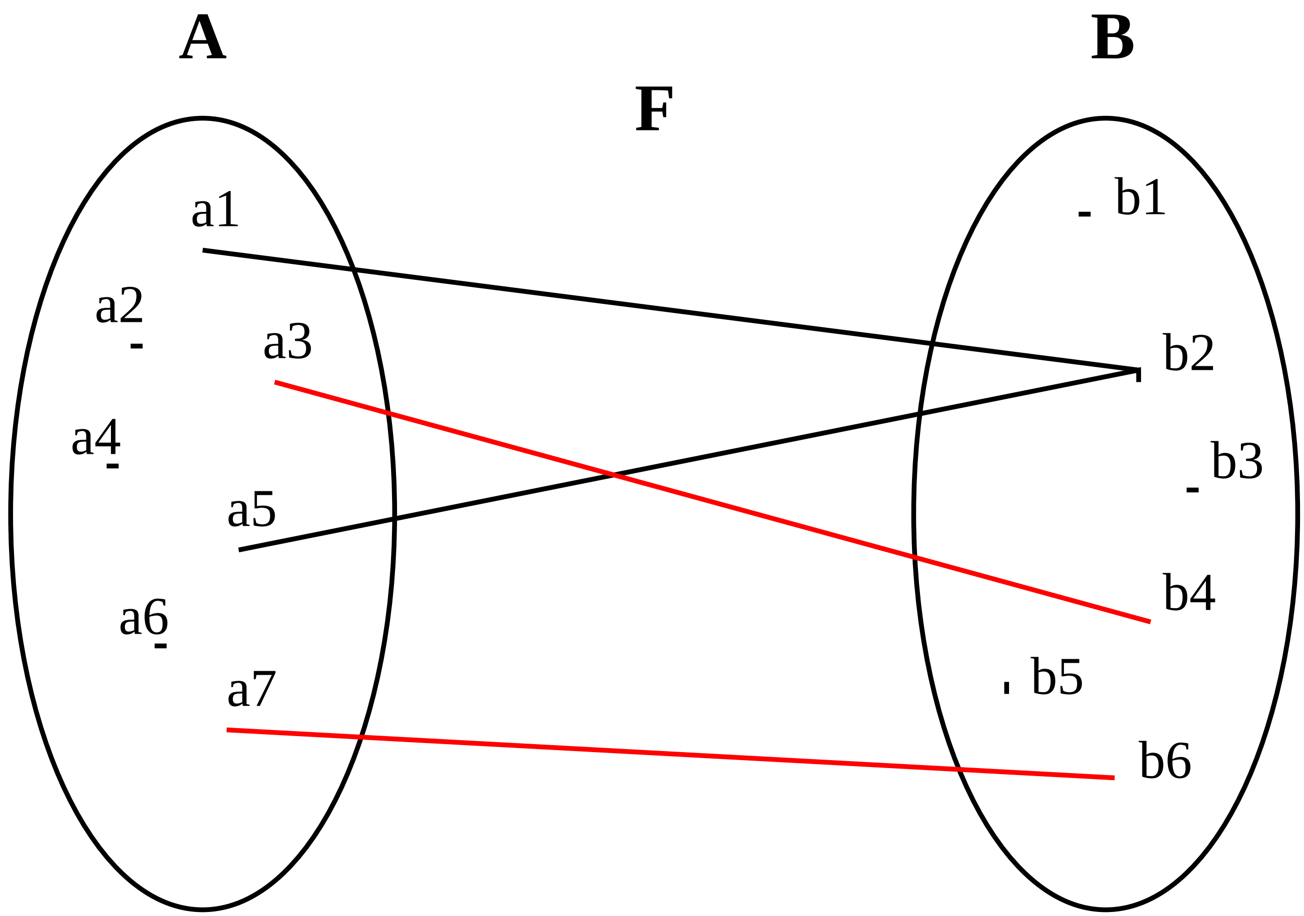
$$\{a3,\ a7\} \lhd F$$



$$\{a3,\ a7\} \lhd F$$



$$\boldsymbol{F \rhd \{b2, b4\}}$$



$$F \mathrel{⩥} \{b2\}$$

Structure of a Model

- List of Carrier Sets (identifiers)

- List of Constants (identifiers)

- List of Axioms (predicates built on sets and constants)

- List of Variables (identifiers)

- List of Invariants (predicates built on sets, constants, and variables)

- List of Events