UNIVERSITY OF Southampton

School of Electronics and Computer Science

Data provenance in a distributed calculus

Vladimiro Sassone

work in progress with A. Francalanza, J. Rathke, and I. Souilah

Motivation

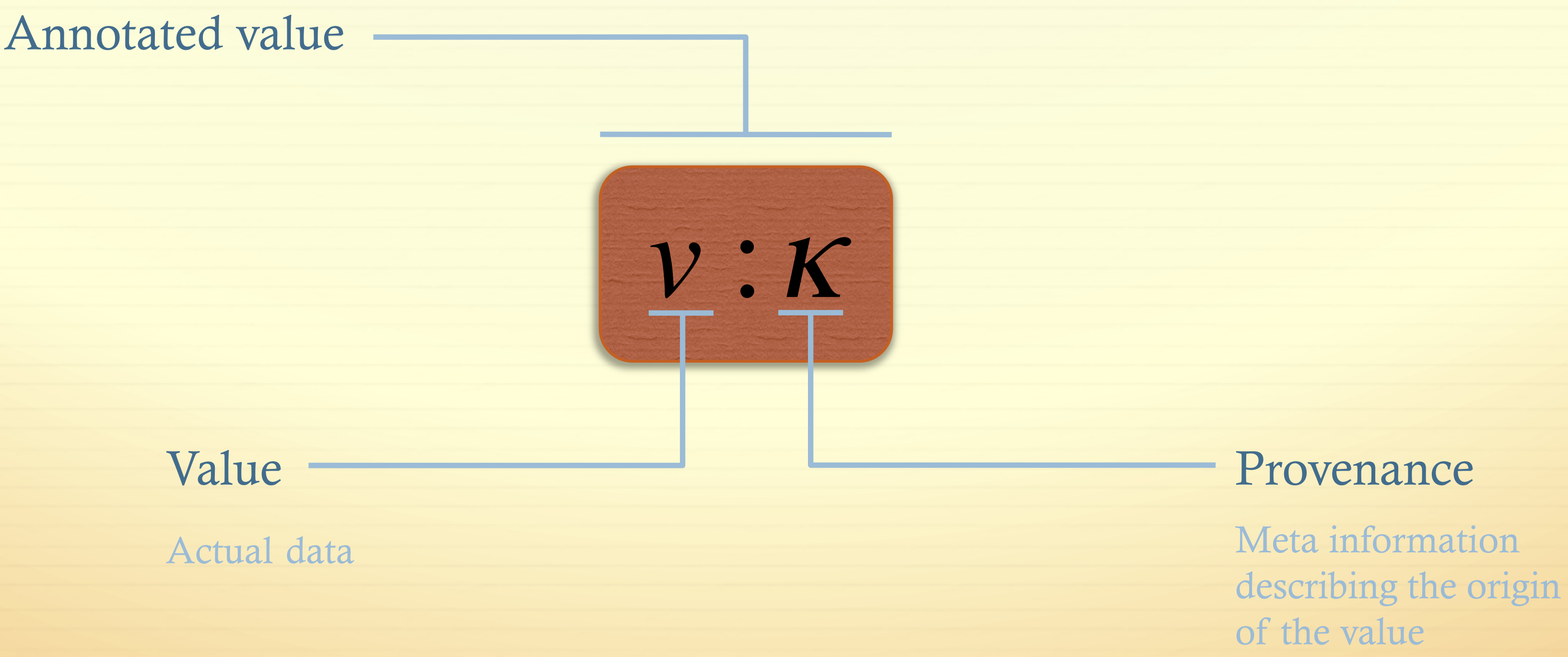
- (Meta)data is almost entirely neglected in the process calculi literature
- Track data provenance both for its important applications and as an challenging exercise in modelling (meta)data. We aim at simplicity:
    - data annotations representing provenance
    - structure, interpretation and management of provenance information
    - provenance tracking
- Provenance-based security (aspects: trust + data confidentiality and privacy)
    - Example: photography competition
- The overall ambition is to underpin practical development, like trust-policy languages and protocols, and provenance-middleware

Model features

- Two the central features of the basic model:
  1. values are annotated with their provenance
  2. provenance is kept up-to-date as computation proceeds
- Focus on one particular kind of provenance information:
  - The principals that influenced a value, and how they did it

Provenance model

Annotated data



Annotated value

$v : \kappa$

Value

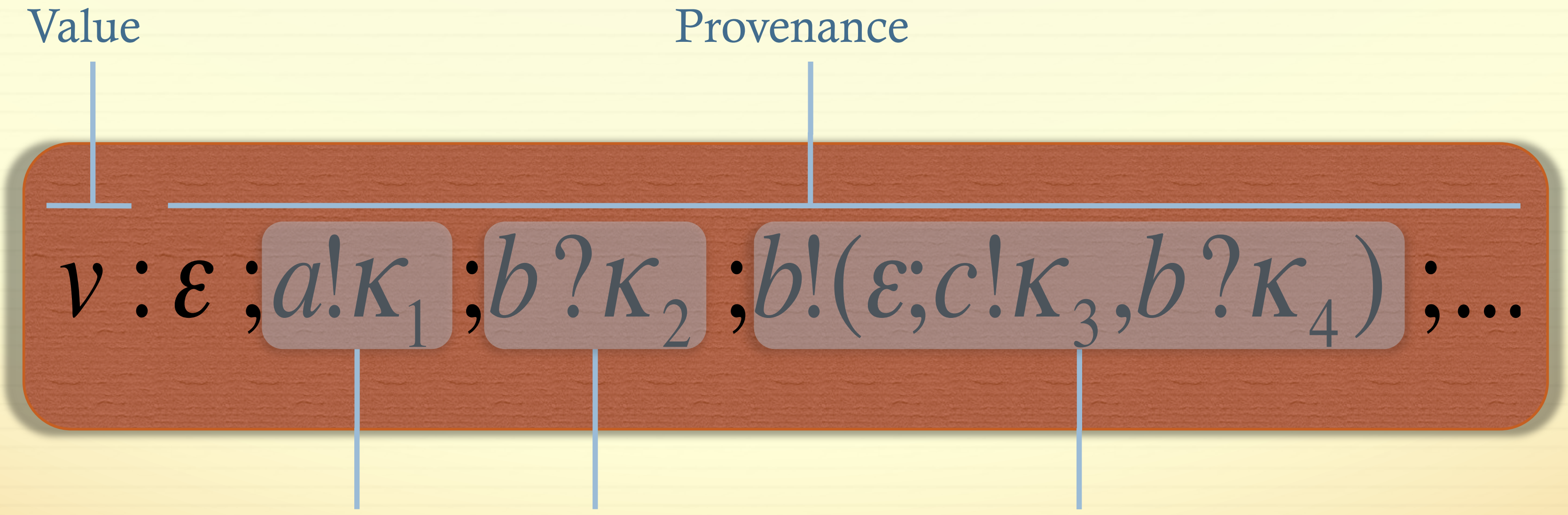
Actual data

Provenance

Meta information describing the origin of the value

Provenance model

Structure and interpretation of provenance



Value

Provenance

$$v : \varepsilon\ ; a!\kappa_1\ ; b?\kappa_2\ ; b!(\varepsilon ; c!\kappa_3, b?\kappa_4)\ ; \ldots$$

“Operations” that were performed on the value. They record the principals that “influenced” the value and how.

Provenance model

Structure and interpretation of provenance

ε (empty provenance) denotes value v originated here

$$v : \varepsilon$$

Provenance model

Structure and interpretation of provenance

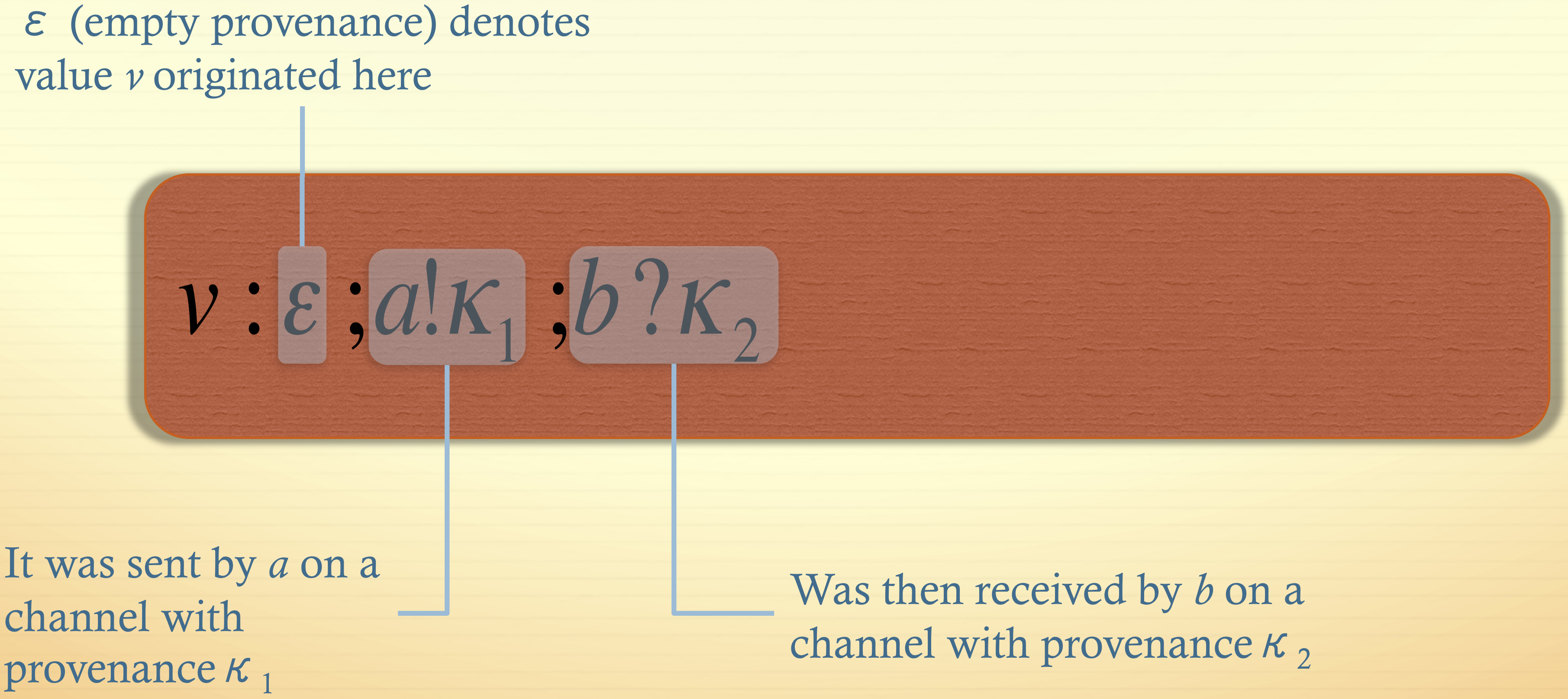
ε (empty provenance) denotes value v originated here

$$v : \varepsilon ; a!\kappa_1$$

It was sent by a on a channel with provenance κ_1

Provenance model

Structure and interpretation of provenance



ε (empty provenance) denotes value v originated here

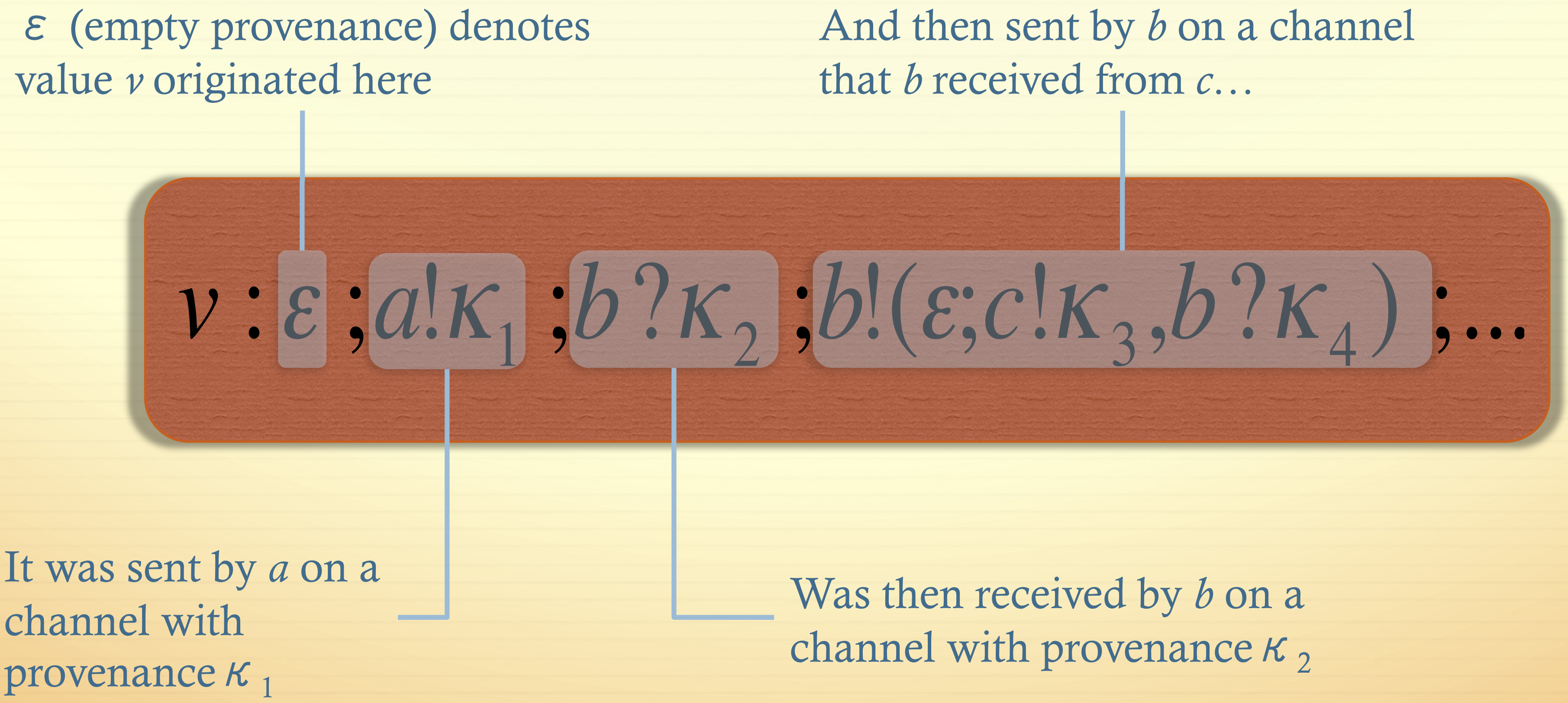
$$v : \varepsilon ; a!\kappa_1 ; b?\kappa_2$$

It was sent by a on a channel with provenance κ_1

Was then received by b on a channel with provenance κ_2

Provenance model

Structure and interpretation of provenance



ε (empty provenance) denotes value *v* originated here

And then sent by *b* on a channel that *b* received from *c*…

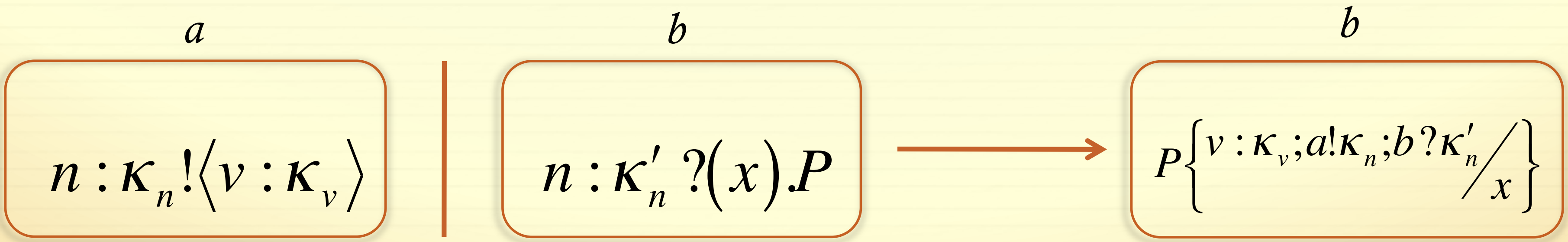
$$v : \varepsilon ; a!\kappa_1 ; b?\kappa_2 ; b!(\varepsilon ; c!\kappa_3 , b?\kappa_4) ; \ldots$$

It was sent by *a* on a channel with provenance κ_1

Was then received by *b* on a channel with provenance κ_2

Provenance model

Provenance tracking



a

$n:\kappa_n!\langle v:\kappa_v\rangle$

b

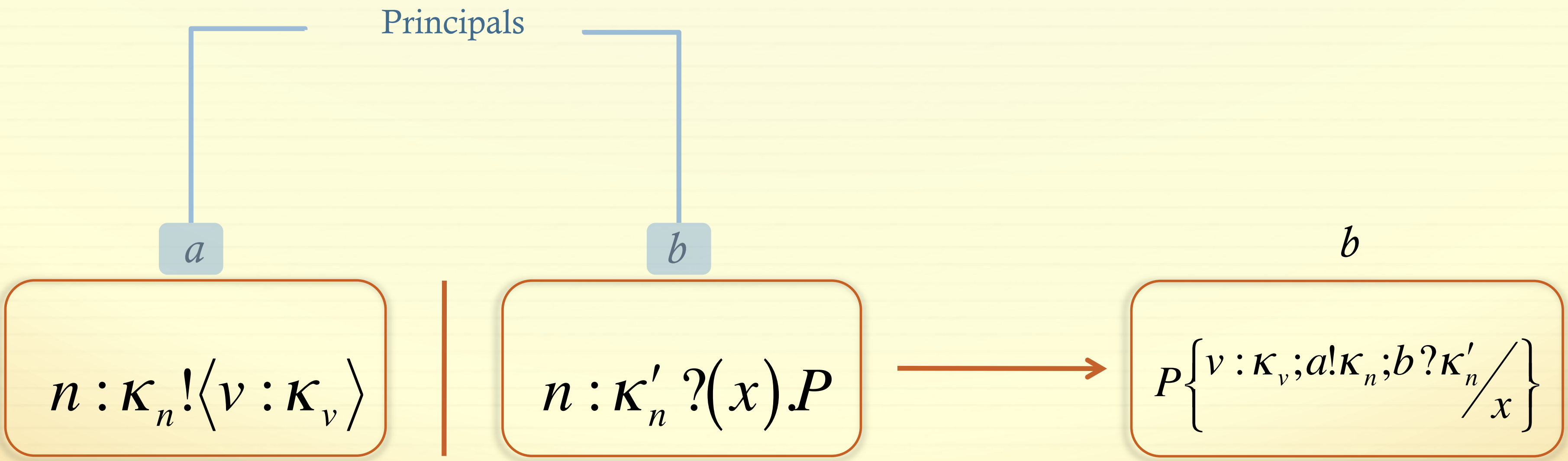
$n:\kappa'_n?(x).P$

b

$P\left\{{}^{v:\kappa_v;a!\kappa_n;b?\kappa'_n}\!/_{x}\right\}$

Provenance model

Provenance tracking



Principals

a

$n : \kappa_n ! \langle v : \kappa_v \rangle$

$\mid$

b

$n : \kappa'_n ?(x).P$

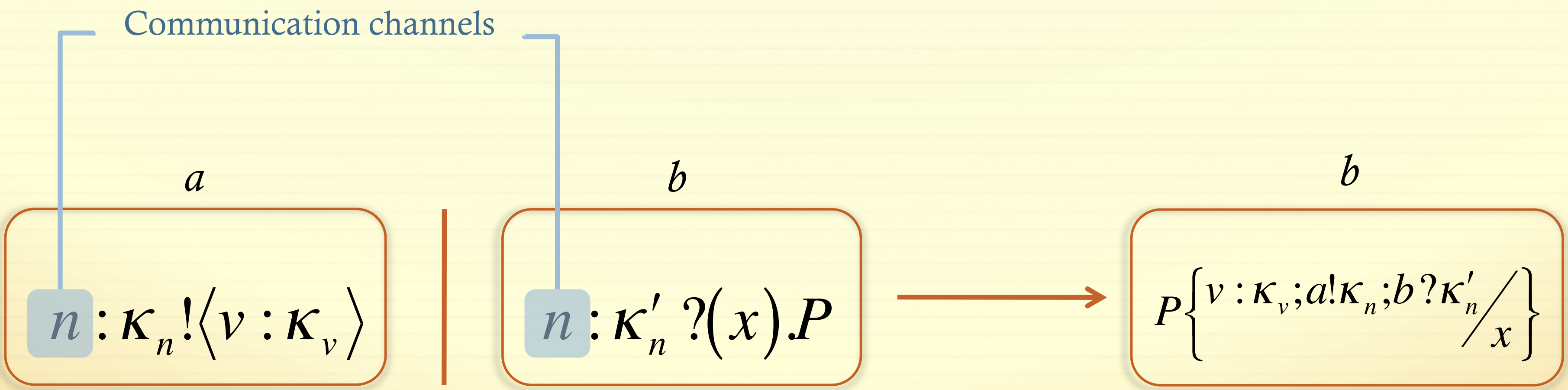
$\longrightarrow$

b

$P\left\{ {}^{v : \kappa_v ; a!\kappa_n ; b?\kappa'_n} \big/ {}_{x} \right\}$

Provenance model

Provenance tracking



Communication channels

a

$n : \kappa_n ! \langle v : \kappa_v \rangle$

$\mid$

b

$n : \kappa'_n ?(x).P$

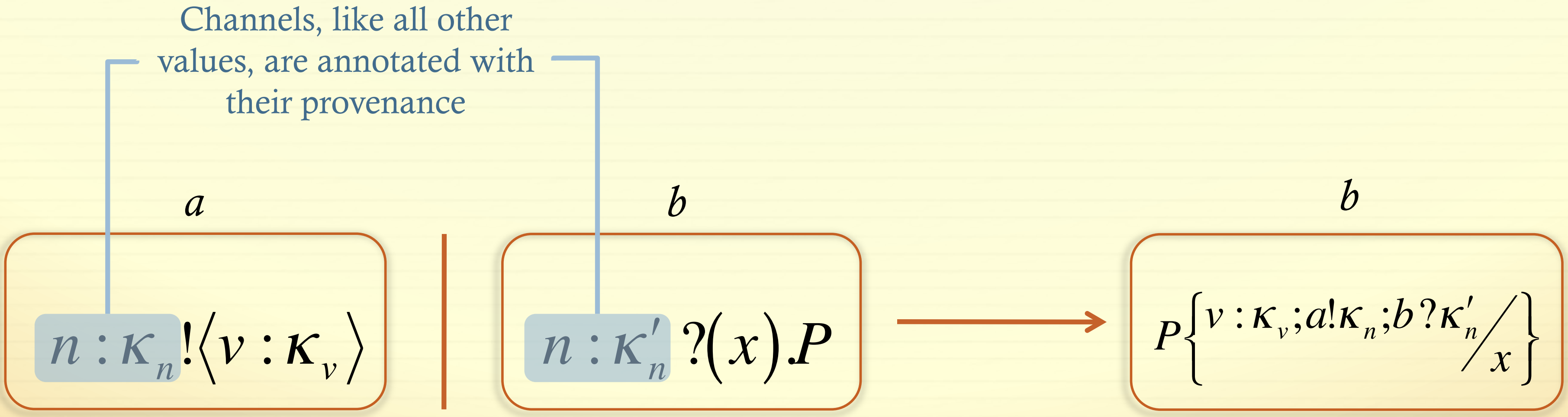
$\longrightarrow$

b

$P\left\{ {}^{v : \kappa_v ; a!\kappa_n ; b?\kappa'_n} \big/ {}_{x} \right\}$

Provenance model

Provenance tracking

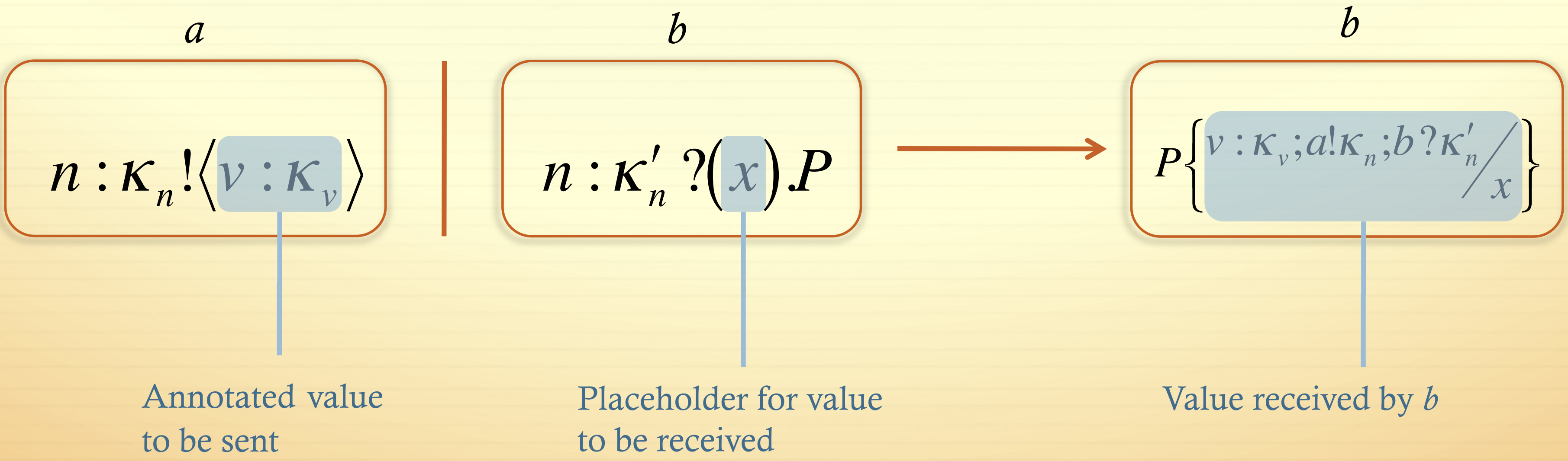


Channels, like all other values, are annotated with their provenance

$$n:\kappa_n!\langle v:\kappa_v\rangle \quad \Big| \quad n:\kappa'_n?(x).P \longrightarrow P\left\{{}^{v:\kappa_v;a!\kappa_n;b?\kappa'_n}\!\big/_{x}\right\}$$

Provenance model

Provenance tracking



$$a \quad n:\kappa_n!\langle v:\kappa_v\rangle \;\Big|\; b \quad n:\kappa'_n?(x).P \longrightarrow b \quad P\left\{{}^{v:\kappa_v;a!\kappa_n;b?\kappa'_n}/_{x}\right\}$$

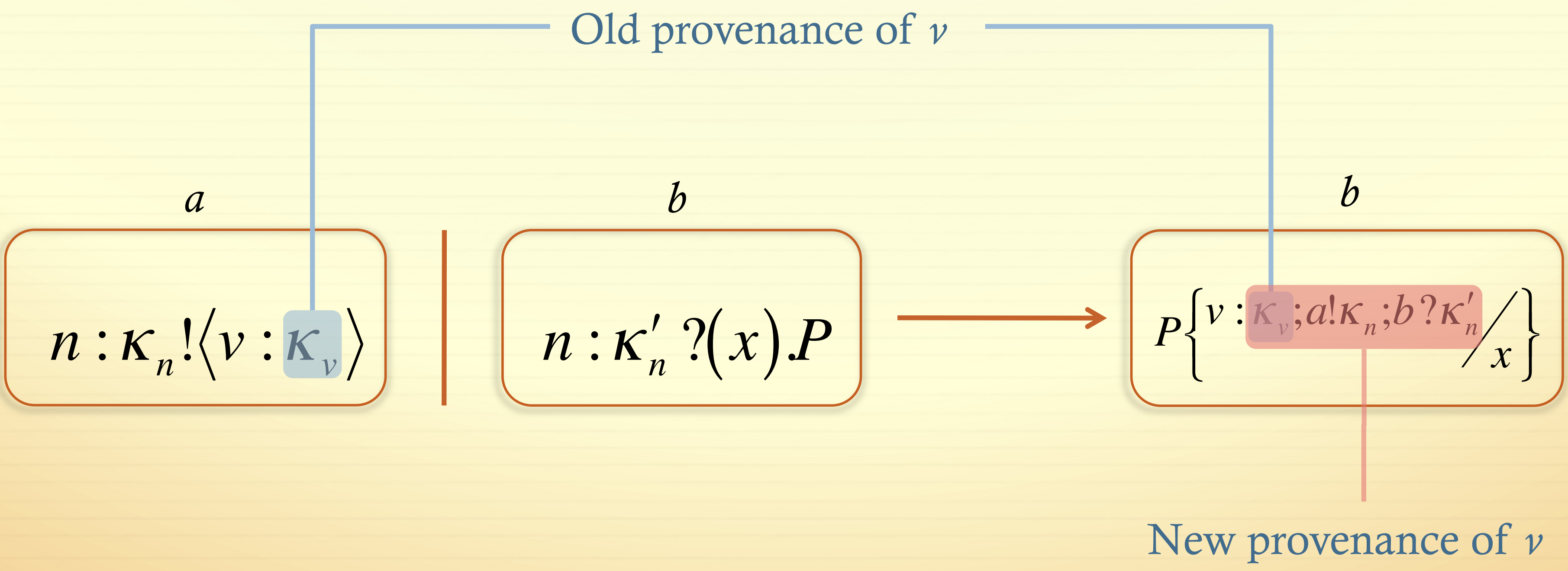
Annotated value to be sent

Placeholder for value to be received

Value received by *b*

Provenance model

Provenance tracking



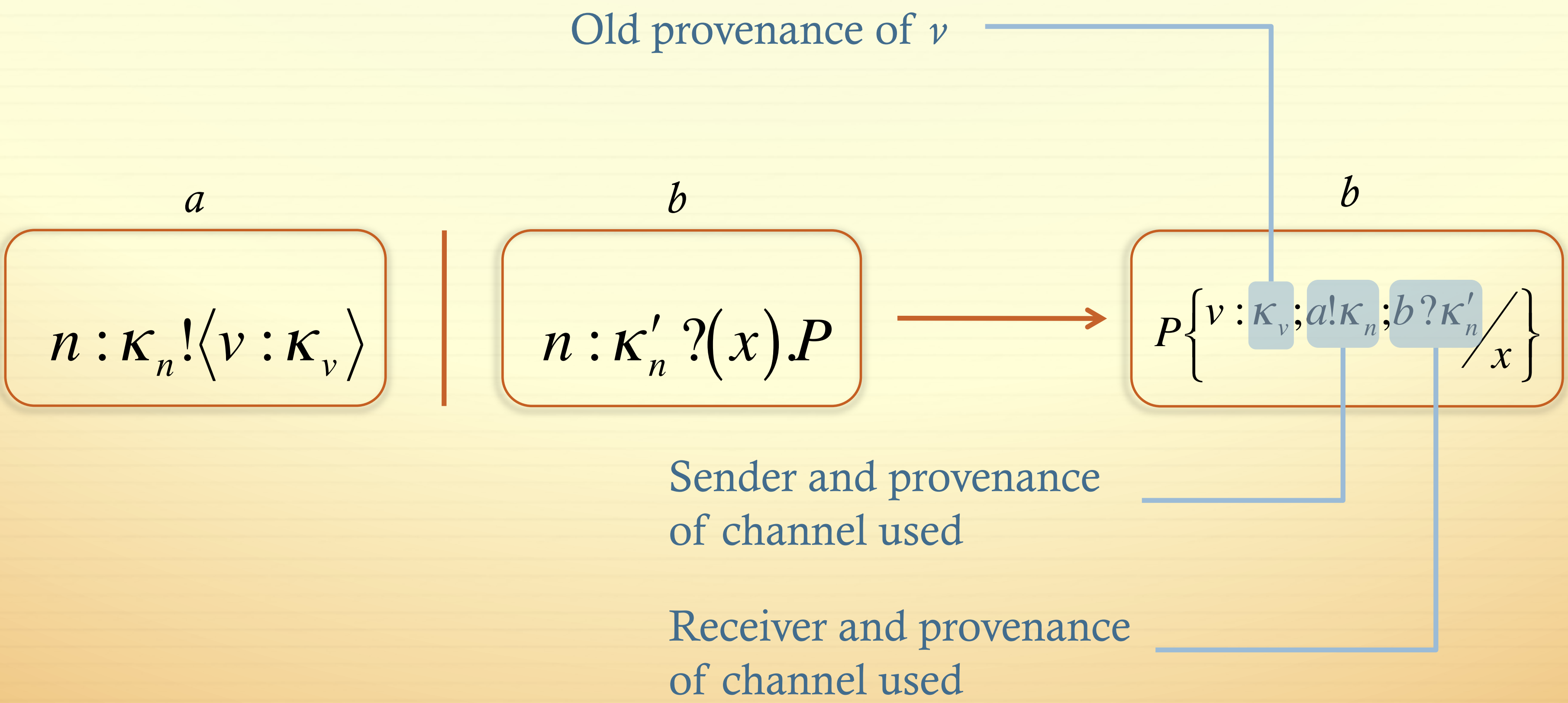
Old provenance of v

$$\overset{a}{\boxed{n : \kappa_n!\langle v : \kappa_v \rangle}} \;\Big|\; \overset{b}{\boxed{n : \kappa'_n ?(x).P}} \longrightarrow \overset{b}{\boxed{P\left\{{}^{v : \kappa_v; a!\kappa_n; b?\kappa'_n}\!\big/_{x}\right\}}}$$

New provenance of v

Provenance model

Provenance tracking



a | b → b

$n:\kappa_n!\langle v:\kappa_v\rangle \mid n:\kappa'_n?(x).P \longrightarrow P\{v:\kappa_v;a!\kappa_n;b?\kappa'_n/x\}$

Old provenance of v

Sender and provenance of channel used

Receiver and provenance of channel used

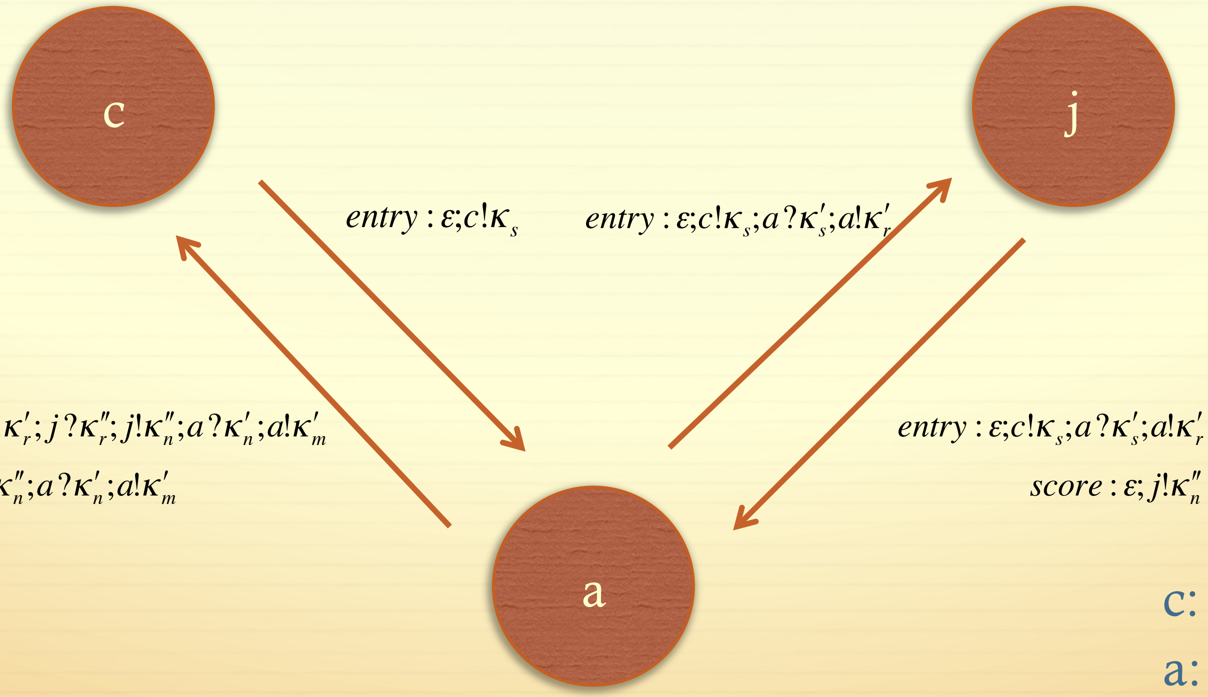
Confidentiality in provenance systems

- Data may be public, yet its provenance confidential, or vice versa
- Principals who may access data are not necessarily the same as those who may access its provenance
- In general, fine grained access control over provenance “histories” is needed as different parts of it have different sensitivity

Security requirements of data ≠ Security requirements of its provenance

Hiding provenance trees

Example: photography competition



$entry : \varepsilon; c!\kappa_s$

$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r$

$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r; j?\kappa''_r; j!\kappa''_n; a?\kappa'_n; a!\kappa'_m$

$score : \varepsilon; j!\kappa''_n; a?\kappa'_n; a!\kappa'_m$

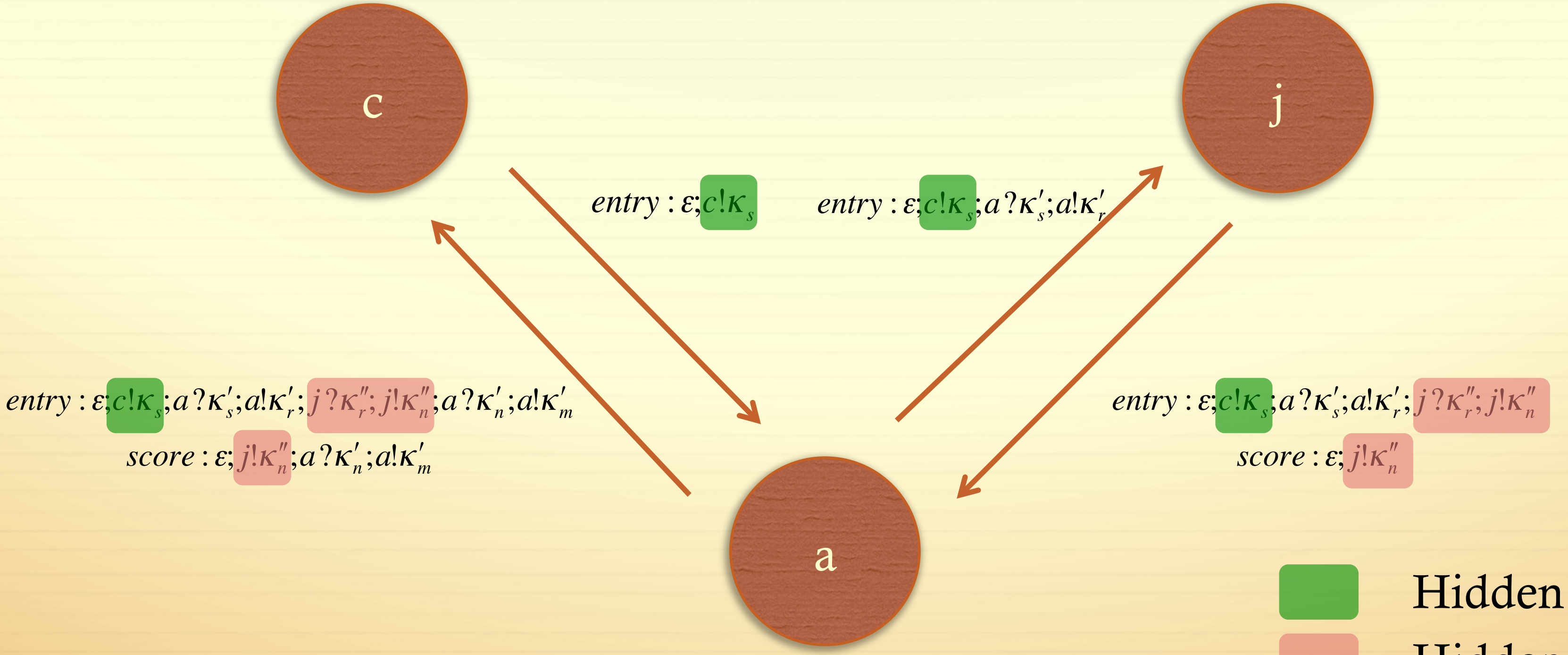
$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r; j?\kappa''_r; j!\kappa''_n$

$score : \varepsilon; j!\kappa''_n$

c:	Contestant
a:	Administrator
j:	Judge

Hiding provenance trees

Example: photography competition



$entry : \varepsilon; c!\kappa_s$

$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r$

$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r; j?\kappa''_r; j!\kappa''_n; a?\kappa'_n; a!\kappa'_m$

$score : \varepsilon; j!\kappa''_n; a?\kappa'_n; a!\kappa'_m$

$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r; j?\kappa''_r; j!\kappa''_n$

$score : \varepsilon; j!\kappa''_n$

🟩 Hidden from j

🟥 Hidden from c

Confidentiality in provenance systems

a promising approach

- One value, multiple **views**
  - Different principals have different views of the same provenance list based on their privileges

$$entry : \varepsilon; c!\kappa_s; a?\kappa'_s; a!\kappa'_r; j?\kappa''_r; j!\kappa''_n; a?\kappa'_n; a!\kappa'_m$$

Confidentiality in provenance systems

a promising approach

- One value, multiple **views**
  - Different principals have different views of the same provenance list based on their privileges

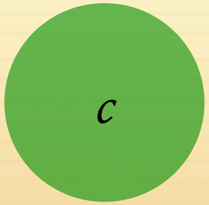
$$\mathit{entry} : \varepsilon;c!\kappa_s;a?\kappa'_s;a!\kappa'_r;j?\kappa''_r;j!\kappa''_n;a?\kappa'_n;a!\kappa'_m$$

Confidentiality in provenance systems

a promising approach

- One value, multiple **views**
  - Different principals have different views of the same provenance list based on their privileges

$$entry : \varepsilon;c!\kappa_s;a?\kappa'_s;a!\kappa'_r;j?\kappa''_r;j!\kappa''_n;a?\kappa'_n;a!\kappa'_m$$



Confidentiality in provenance systems

a promising approach

- One value, multiple **views**
  - Different principals have different views of the same provenance list based on their privileges

$$entry : \varepsilon;c!\kappa_s;a?\kappa'_s;a!\kappa'_r;j?\kappa''_r;j!\kappa''_n;a?\kappa'_n;a!\kappa'_m$$

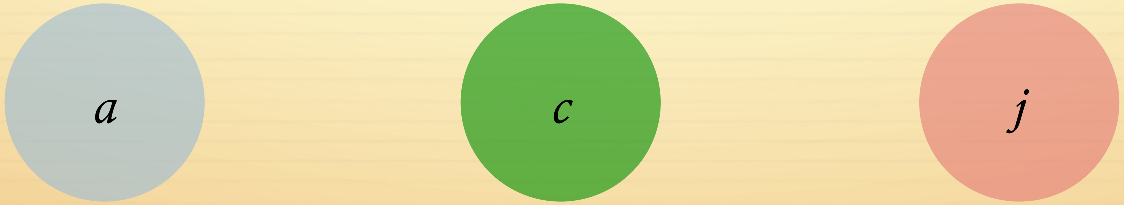
j

Confidentiality in provenance systems

a promising approach

- One value, multiple **views**
  - Different principals have different views of the same provenance list based on their privileges

$$entry : \varepsilon;c!\kappa_s;a?\kappa'_s;a!\kappa'_r;j?\kappa''_r;j!\kappa''_n;a?\kappa'_n;a!\kappa'_m$$



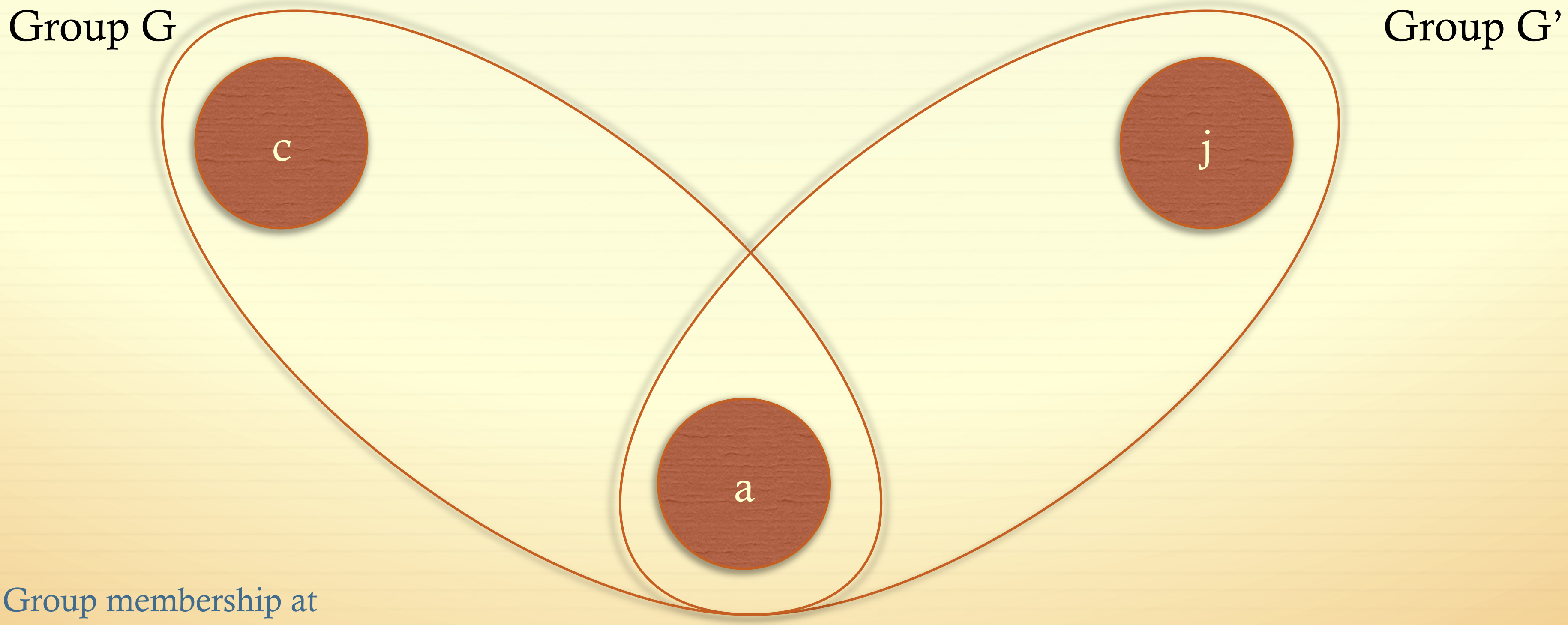
Confidentiality in provenance systems

a promising approach

- To achieve this, we use ***groups***
- Different principals belong to different groups
- Group membership determines what parts of a provenance list a principal has access to
- Principals
  - Can create new groups: $\text{new } G$
  - Can add other principals to their groups: $\text{add}(a,G)$
  - Can **restrict** access to particular parts of a provenance tree to a particular group: $\text{hide}(v:\kappa,G)$

Hiding provenance trees

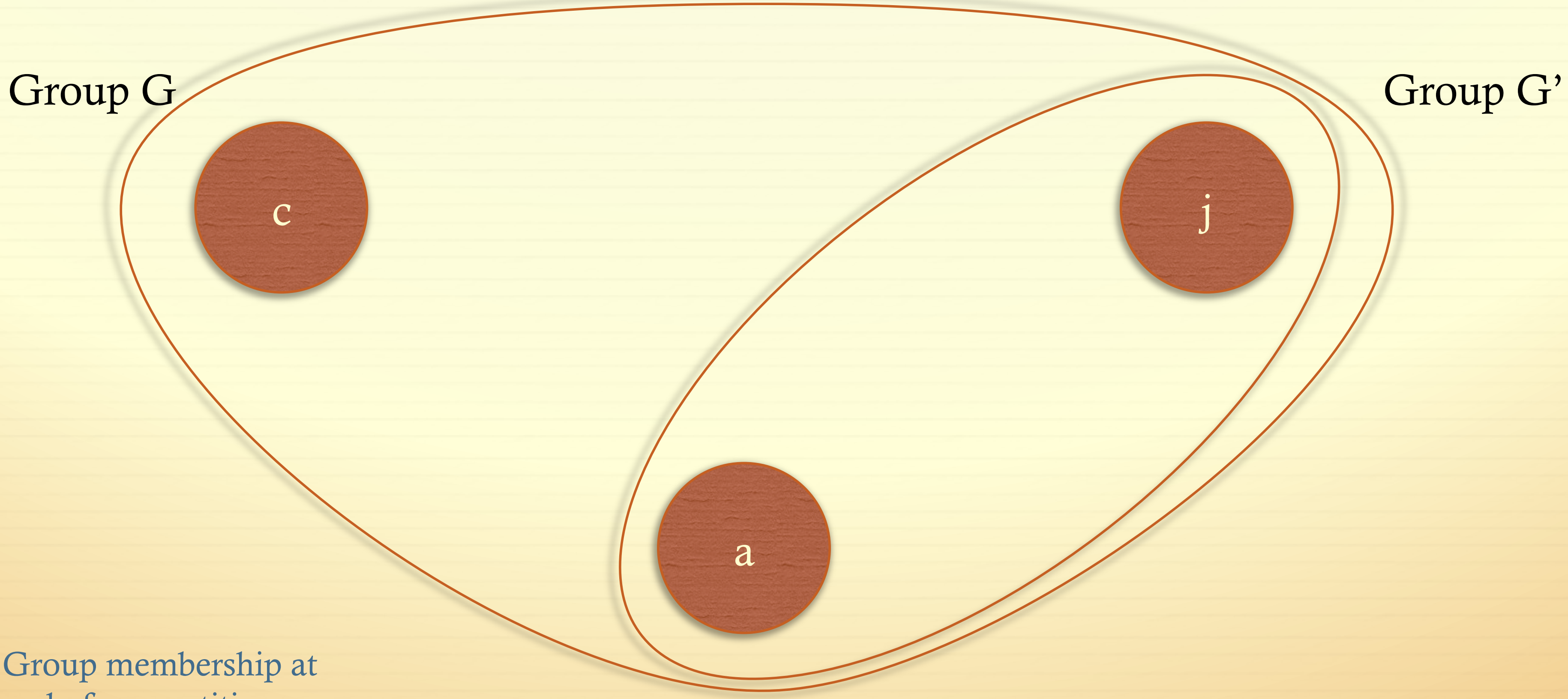
photography competition



Group membership at
start of competition

Hiding provenance trees

photography competition



Group membership at
end of competition

Current work

- Correctness of provenance tracking: the provenance information determines the history of each piece of data accurately “enough”
  - Express this as a form of testing (on traces):

$$\forall S (\forall t \in [\![S]\!] (t \rightarrow^* v : \kappa \implies \forall s \in [\![v : \kappa]\!] (s | t \rightarrow^* \checkmark)))$$

- Using provenance:
  - Provenance queries vs pattern restricted input
  - Trust in quality of data based on trust in principals and provenance of data
- Policies and types