Hyper Separation Logic

Trayan Gospodinov, Peter Müller, Thibault Dardinier

INSAIT ETH zürich EPFL

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$

For any two executions:
same public inputs ⇒ same public outputs

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$
Generalized NI	$\forall\forall\exists$

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$
Generalized NI	$\forall\forall\exists$
Non-determinism	$\exists\exists$
Existence of a minimum	$\exists\forall$
Functional correctness	$\forall$
Reachability	$\exists$

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$
Generalized NI	$\forall\forall\exists$
Non-determinism	$\exists\exists$
Existence of a minimum	$\exists\forall$
Functional correctness	$\forall$
Reachability	$\exists$

Separation logic

$$\frac{\{p\}\ C\ \{q\}}{\{p * f\}\ C\ \{q * f\}}$$

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	∀∀
Generalized NI	∀∀∃
Non-determinism	∃∃
Existence of a minimum	∃∀
Functional correctness	∀
Reachability	∃

Separation logic

Local spec

$$\frac{\{p\}\ C\ \{q\}}{\{p * f\}\ C\ \{q * f\}}$$

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$
Generalized NI	$\forall\forall\exists$
Non-determinism	$\exists\exists$
Existence of a minimum	$\exists\forall$
Functional correctness	$\forall$
Reachability	$\exists$

Separation logic

Local spec

$$\frac{\{p\}\ C\ \{q\}}{\{p * f\}\ C\ \{q * f\}}$$

Augmented with additional heap

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	$\forall\forall$
Generalized NI	$\forall\forall\exists$
Non-determinism	$\exists\exists$
Existence of a minimum	$\exists\forall$
Functional correctness	$\forall$
Reachability	$\exists$

Separation logic

Local spec

$$\frac{\{p\}\ C\ \{q\}}{\{p * f\}\ C\ \{q * f\}}$$

Augmented with additional heap

Goal: Modular Reasoning for Hyperproperties

Hyperproperties

Hyperproperty	Type
Non-interference (NI)	∀∀
Generalized NI	∀∀∃
Non-determinism	∃∃
Existence of a minimum	∃∀
Functional correctness	∀
Reachability	∃

+

Separation logic

Local spec

$$\frac{\{p\}\ C\ \{q\}}{\{p * f\}\ C\ \{q * f\}}$$

Augmented with additional heap

Existing Program Logics for Hyperproperties

Hyperproperty	No heap
$\forall$	Hoare Logic

Existing Program Logics for Hyperproperties

Hyperproperty	No heap	With heap
∀	Hoare Logic	Separation Logic

Existing Program Logics for Hyperproperties

Hyperproperty	No heap	With heap
∀	HL	SL
∃	IncorrectnessL	IncorrectnessSL

Existing Program Logics for Hyperproperties

Hyperproperty	No heap	With heap
$\forall$	HL, OL	SL, OSL
$\exists$	IL, SIL, OL	ISL, SSIL, OSL
$\forall\forall$	RHL	RSL
$\forall^*$	CHL, LHC	LGTM
$\exists\exists$	InsecL	InsecSL
$\forall^*\exists^*$	RHLE	
$\exists^*\forall^*$		
Other combinations		

Existing Program Logics for Hyperproperties

Tracks **sets** of states

Hyperproperty	No heap	With heap
$\forall$	HL, OL, **HHL**	SL, OSL
$\exists$	IL, SIL, OL, **HHL**	ISL, SSIL, OSL
$\forall\forall$	RHL, **HHL**	RSL
$\forall^*$	CHL, LHC, **HHL**	LGTM
$\exists\exists$	InsecL, **HHL**	InsecSL
$\forall^*\exists^*$	RHLE, **HHL**	
$\exists^*\forall^*$	**HHL**	
Other combinations	**HHL**	

Existing Program Logics for Hyperproperties

Hyperproperty	No heap	With heap
∀	HL, OL, **HHL**	SL, OSL
∃	IL, SIL, OL, **HHL**	ISL, SSIL, OSL
∀∀	RHL, **HHL**	RSL
$\forall^*$	CHL, LHC, **HHL**	LGTM
∃∃	InsecL, **HHL**	InsecSL
∀*∃*	RHLE, **HHL**	
∃*∀*	**HHL**	
Other combinations	**HHL**	

Big gap

Existing Program Logics for Hyperproperties

Hyperproperty	No heap	With heap
$\forall$	HL, OL, **HHL**	SL, OSL, **HSL**
$\exists$	IL, SIL, OL, **HHL**	ISL, SSIL, OSL, **HSL**
$\forall\forall$	RHL, **HHL**	RSL, **HSL**
$\forall^*$	CHL, LHC, **HHL**	LGTM, **HSL**
$\exists\exists$	InsecL, **HHL**	InsecSL, **HSL**
$\forall^*\exists^*$	RHLE, **HHL**	**HSL**
$\exists^*\forall^*$	**HHL**	**HSL**
Other combinations	**HHL**	**HSL**

Tracks **sets** of states

Quantification Over Executions

$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}$ [y]:=[x]+1 $\{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$

Quantification Over Executions

$\{\forall\langle\sigma\rangle.\ \sigma(x{\mapsto}5 * y{\mapsto}\_)\}$ [y]:=[x]+1 $\{\forall\langle\sigma'\rangle.\ \sigma'(x{\mapsto}5 * y{\mapsto}6)\}$

Initial state

Quantification Over Executions

$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}$ [y]:=[x]+1 $\{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$

Initial state

Reachable state

Quantification Over Executions

$$\underbrace{\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}\ [y]:=[x]+1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}}_{\{x\mapsto 5 * y\mapsto \_\}\ [y]:=[x]+1\ \{x\mapsto 5 * y\mapsto 6\}}$$

Quantification Over Executions

Separation Logic (SL)

$$\underbrace{\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}\ [y]{:=}[x]+1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}}$$

$$\{x\mapsto 5 * y\mapsto \_\}\ [y]{:=}[x]+1\ \{x\mapsto 5 * y\mapsto 6\}$$

Quantification Over Executions

Separation Logic (SL)

$$\{\forall\langle\sigma\rangle.\ \sigma(x \mapsto 5 * y \mapsto \_)\}\ [y] := [x]+1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x \mapsto 5 * y \mapsto 6)\}$$

$$\{\exists\langle\sigma\rangle.\ \sigma(r \mapsto \_)\}\ [r] := \mathrm{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r \mapsto 1)\}$$

Quantification Over Executions

Separation Logic (SL)

$\{\forall\langle\sigma\rangle.\ \sigma(x \mapsto 5 * y \mapsto \_)\}$ [y]:=[x]+1 $\{\forall\langle\sigma'\rangle.\ \sigma'(x \mapsto 5 * y \mapsto 6)\}$

$$\underbrace{\{\exists\langle\sigma\rangle.\ \sigma(r \mapsto \_)\}\ [r]:=\text{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r \mapsto 1)\}}_{\{r \mapsto \_\}\ [r]:=\text{rand}(1,4)\ \{r \mapsto 1\}}$$

Quantification Over Executions

Separation Logic (SL)

$$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}\ [y]:=[x]+1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$$

Separation Sufficient Incorrectness Logic (SSIL)

$$\underbrace{\{\exists\langle\sigma\rangle.\ \sigma(r\mapsto \_)\}\ [r]:=\mathrm{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r\mapsto 1)\}}$$

$$\{r\mapsto \_\}\ [r]:=\mathrm{rand}(1,4)\ \{r\mapsto 1\}$$

Quantification Over Executions

Separation Logic (SL)

$$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}\ [y]:=[x]+1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$$

Separation Sufficient Incorrectness Logic (SSIL)

$$\{\exists\langle\sigma\rangle.\ \sigma(r\mapsto \_)\}\ [r]:=\mathrm{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r\mapsto 1)\}$$

Relational Separation Logic (RSL)

$$\left\{\exists n.\begin{bmatrix} l_1 \mapsto n \\ l_2 \mapsto n \end{bmatrix}\right\}\ o:=f(\underbrace{l}_{\text{low}},\ \underbrace{h}_{\text{high}})\ \left\{\exists n.\begin{bmatrix} o_1 \mapsto n \\ o_2 \mapsto n \end{bmatrix}\right\}$$

Quantification Over Executions

Separation Logic (SL)

$$\{\forall\langle\sigma\rangle.\ \sigma(x \mapsto 5 * y \mapsto \_)\}\ [y] := [x] + 1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x \mapsto 5 * y \mapsto 6)\}$$

Separation Sufficient Incorrectness Logic (SSIL)

$$\{\exists\langle\sigma\rangle.\ \sigma(r \mapsto \_)\}\ [r] := \mathrm{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r \mapsto 1)\}$$

Relational Separation Logic (RSL)

$$\left\{\exists n.\begin{bmatrix} l_1 & \mapsto & n \\ l_2 & \mapsto & n \end{bmatrix}\right\}\ o := f(l, h)\ \left\{\exists n.\begin{bmatrix} o_1 & \mapsto & n \\ o_2 & \mapsto & n \end{bmatrix}\right\}$$

Start with same low

Quantification Over Executions

Separation Logic (SL)

$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}$ [y]:=[x]+1 $\{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$

Separation Sufficient Incorrectness Logic (SSIL)

$\{\exists\langle\sigma\rangle.\ \sigma(r\mapsto \_)\}$ [r]:=rand(1,4) $\{\exists\langle\sigma'\rangle.\ \sigma'(r\mapsto 1)\}$

Relational Separation Logic (RSL)

Ends with same output

$$\left\{\exists n.\begin{bmatrix} l_1 \mapsto n \\ l_2 \mapsto n \end{bmatrix}\right\} \text{ o:=f(l, h) } \left\{\exists n.\begin{bmatrix} o_1 \mapsto n \\ o_2 \mapsto n \end{bmatrix}\right\}$$

Start with same low

Quantification Over Executions

Separation Logic (SL)

$$\{\forall\langle\sigma\rangle.\ \sigma(x\mapsto 5 * y\mapsto \_)\}\ [y]{:=}[x]{+}1\ \{\forall\langle\sigma'\rangle.\ \sigma'(x\mapsto 5 * y\mapsto 6)\}$$

Separation Sufficient Incorrectness Logic (SSIL)

$$\{\exists\langle\sigma\rangle.\ \sigma(r\mapsto \_)\}\ [r]{:=}\mathrm{rand}(1,4)\ \{\exists\langle\sigma'\rangle.\ \sigma'(r\mapsto 1)\}$$

Relational Separation Logic (RSL)

$$\{\forall\langle\sigma_1\rangle.\ \forall\langle\sigma_2\rangle.\ \exists n.\ \sigma_1(l\mapsto n) \land \sigma_2(l\mapsto n)\}$$
$$o{:=}f(l, h)$$
$$\{\forall\langle\sigma_1'\rangle.\ \forall\langle\sigma_2'\rangle.\ \exists n.\ \sigma_1'\ (o\mapsto n) \land \sigma_2'\ (o\mapsto n)\}$$

Quantification Over Executions

Separation Logic (SL)

$\{\forall\langle\sigma\rangle$ … $\mapsto 6)\}$

Separatio…

Relational…

Logic	HSL encoding
SL ($\forall$)	$\{\forall\}$ C $\{\forall\}$
SSIL ($\exists$)	$\{\exists\}$ C $\{\exists\}$
RSL ($\forall\forall$)	$\{\forall\forall\}$ C $\{\forall\forall\}$
…	…

$\{\forall\langle\sigma_1'\rangle.\ \forall\langle\sigma_2'\rangle.\ \exists n.\ \sigma_1'(o\mapsto n) \wedge \sigma_2'(o\mapsto n)\}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

New: **Hyper** separating conjunction

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^+}{\{P \star F\}\ C\ \{Q \star F\}}$$

∃ forces termination

New: **Hyper** separating conjunction

Key Result: $\boxed{\text{Generalized}}$ Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

∃ forces termination

New: **Hyper** separating conjunction

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

$$\{\forall\langle\sigma\rangle.\, \sigma(x \mapsto 5)\}\ [x] := [x]+1\ \{\forall\langle\sigma\rangle.\, \sigma(x \mapsto 6)\}$$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^+}{\{P \star F\}\ C\ \{Q \star F\}}$$

$$\frac{\{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)\}\ [x]{:=}[x]+1\ \{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)\}}{\{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)) \star (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))\}\ [x]{:=}[x]+1\ \{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)) \star (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))\}}$$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

$$\frac{\{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)\}\ [x] \coloneqq [x]+1\ \{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)\}}{\{\underbrace{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)) \bigstar (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))}_{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5 * y \mapsto \_)}\}\ [x] \coloneqq [x]+1\ \{\underbrace{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)) \bigstar (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))}_{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6 * y \mapsto \_)}\}}$$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

$$\{x \mapsto 5\}\ [x]{:=}[x]{+}1\ \{x \mapsto 6\}$$

$$\frac{\{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)\}\ [x]{:=}[x]{+}1\ \{\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)\}}{\{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 5)) \bigstar (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))\}\ [x]{:=}[x]{+}1\ \{(\forall\langle\sigma\rangle.\,\sigma(x \mapsto 6)) \bigstar (\forall\langle\sigma\rangle.\,\sigma(y \mapsto \_))\}}$$

$$\{x \mapsto 5 * y \mapsto \_\}\ [x]{:=}[x]{+}1\ \{x \mapsto 6 * y \mapsto \_\}$$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \star \forall\}\ C\ \{\forall \star \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \star \forall\forall\}\ C\ \{\forall\forall \star \forall\forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \bigstar \forall\forall\}\ C\ \{\forall\forall \bigstar \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \bigstar \exists\}\ C\ \{\exists \bigstar \exists\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \star \forall\}\ C\ \{\forall \star \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \star \forall\forall\}\ C\ \{\forall\forall \star \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \cancel{\star \exists}\}\ C\ \{\exists \cancel{\star \exists}\}}$

The two witness executions might be unrelated

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL (∀)	$\frac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$
RSL (∀∀)	$\frac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \bigstar \forall\forall\}\ C\ \{\forall\forall \bigstar \forall\forall\}}$
SSIL (∃)	$\frac{\{\exists\}\ C\ \{\exists\}}{\{\exists \bigstar \forall\}\ C\ \{\exists \bigstar \forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad \boxed{F \text{ is } \forall^{+}}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \bigstar \forall\forall\}\ C\ \{\forall\forall \bigstar \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \bigstar \forall\}\ C\ \{\exists \bigstar \forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \star \forall\}\ C\ \{\forall \star \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \star \forall\forall\}\ C\ \{\forall\forall \star \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \star \forall\}\ C\ \{\exists \star \forall\}}$

Other combinations
$\dfrac{\{\exists\forall\}\ C\ \{\exists\forall\}}{\{\exists\forall \star \forall\forall\}\ C\ \{\exists\forall \star \forall\forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \bigstar \forall\forall\}\ C\ \{\forall\forall \bigstar \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \bigstar \forall\}\ C\ \{\exists \bigstar \forall\}}$

Other combinations
$\dfrac{\{\exists\forall\}\ C\ \{\exists\forall\}}{\{\exists\forall \bigstar \forall\forall\}\ C\ \{\exists\forall \bigstar \forall\forall\}}$
$\dfrac{\{\forall\forall\exists\}\ C\ \{\forall\forall\exists\}}{\{\forall\forall\exists \bigstar \forall\forall\}\ C\ \{\forall\forall\exists \bigstar \forall\forall\}}$

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \bigstar F\}\ C\ \{Q \bigstar F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \bigstar \forall\}\ C\ \{\forall \bigstar \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \bigstar \forall\forall\}\ C\ \{\forall\forall \bigstar \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \bigstar \forall\}\ C\ \{\exists \bigstar \forall\}}$

Other combinations
$\dfrac{\{\exists\forall\}\ C\ \{\exists\forall\}}{\{\exists\forall \bigstar \forall\forall\}\ C\ \{\exists\forall \bigstar \forall\forall\}}$
$\dfrac{\{\forall\forall\exists\}\ C\ \{\forall\forall\exists\}}{\{\forall\forall\exists \bigstar \forall\forall\}\ C\ \{\forall\forall\exists \bigstar \forall\forall\}}$

Generalized non-interference

Non-interference

Key Result: Generalized Frame Rule

$$\frac{\{P\}\ C\ \{Q\} \qquad F \text{ is } \forall^{+}}{\{P \star F\}\ C\ \{Q \star F\}}$$

Logic	Frame rule
SL ($\forall$)	$\dfrac{\{\forall\}\ C\ \{\forall\}}{\{\forall \star \forall\}\ C\ \{\forall \star \forall\}}$
RSL ($\forall\forall$)	$\dfrac{\{\forall\forall\}\ C\ \{\forall\forall\}}{\{\forall\forall \star \forall\forall\}\ C\ \{\forall\forall \star \forall\forall\}}$
SSIL ($\exists$)	$\dfrac{\{\exists\}\ C\ \{\exists\}}{\{\exists \star \forall\}\ C\ \{\exists \star \forall\}}$

Other combinations
$\dfrac{\{\exists\forall\}\ C\ \{\exists\forall\}}{\{\exists\forall \star \forall\forall\}\ C\ \{\exists\forall \star \forall\forall\}}$
$\dfrac{\{\forall\forall\exists\}\ C\ \{\forall\forall\exists\}}{\{\forall\forall\exists \star \forall\forall\}\ C\ \{\forall\forall\exists \star \forall\forall\}}$
…

Key Challenge: Defining ★

$$\boxed{p * q \triangleq \{\sigma_p \oplus \sigma_q \mid \sigma_p \vDash p \land \sigma_q \vDash q\}}$$

Key Challenge: Defining ★

$$\boxed{p * q \triangleq \{\sigma_p \oplus \sigma_q \mid \sigma_p \vDash p \land \sigma_q \vDash q\}}$$

Key Challenge: Defining ★

$$\boxed{p * q \triangleq \{\sigma_p \bigoplus \sigma_q \mid \sigma_p \vDash p \wedge \sigma_q \vDash q\}}$$

Key Challenge: Defining ★

$$\boxed{p * q \triangleq \{\sigma_p \oplus \sigma_q \mid \sigma_p \vDash p \land \sigma_q \vDash q\}}$$

Logic	**Combination predicate $\oplus$**
SL, ISL, SSIL	$\sigma_p \oplus \sigma_q \triangleq \sigma_p \uplus \sigma_q$

Disjoint union

Key Challenge: Defining ★

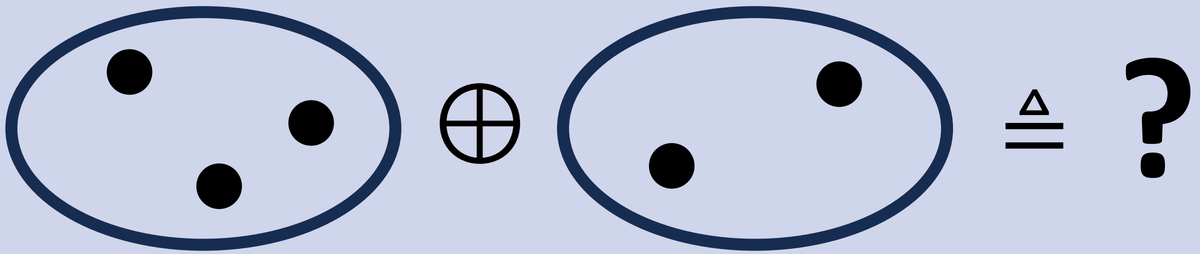
$$\boxed{p * q \triangleq \{\sigma_p \bigoplus \sigma_q \mid \sigma_p \vDash p \land \sigma_q \vDash q\}}$$

Logic	Combination predicate $\bigoplus$
SL, ISL, SSIL	$\sigma_p \bigoplus \sigma_q \triangleq \sigma_p \uplus \sigma_q$
RSL, LGTM, InsecSL	$\begin{bmatrix}\sigma_p^1 \\ \sigma_p^2\end{bmatrix} \bigoplus \begin{bmatrix}\sigma_q^1 \\ \sigma_q^2\end{bmatrix} \triangleq \begin{bmatrix}\sigma_p^1 \uplus \sigma_q^1 \\ \sigma_p^2 \uplus \sigma_q^2\end{bmatrix}$

Pointwise lifted $\uplus$

Key Challenge: Defining ★

$$p * q \triangleq \{\sigma_p \oplus \sigma_q \mid \sigma_p \vDash p \land \sigma_q \vDash q\}$$

Logic	**Combination predicate $\oplus$**
SL, ISL, SSIL	$\sigma_p \oplus \sigma_q \triangleq \sigma_p \uplus \sigma_q$
RSL, LGTM, InsecSL	$\begin{bmatrix}\sigma_p^1 \\ \sigma_p^2\end{bmatrix} \oplus \begin{bmatrix}\sigma_q^1 \\ \sigma_q^2\end{bmatrix} \triangleq \begin{bmatrix}\sigma_p^1 \uplus \sigma_q^1 \\ \sigma_p^2 \uplus \sigma_q^2\end{bmatrix}$
HSL	

Hyper Separating Conjunction

$\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)$

Hyper Separating Conjunction

$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar$

Hyper Separating Conjunction

$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_))$

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x\mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y\mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x\mapsto \_ * y\mapsto \_))$$

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x\mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y\mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x\mapsto \_ * y\mapsto \_))$$

$$\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_)$$

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_))$$

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Longrightarrow (\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Longrightarrow (\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

All satisfy y↦_

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_))\ \bigstar\ (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\ \_))\Leftrightarrow(\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_ * y\mapsto\ \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_))\ \bigstar\ (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_))\Longrightarrow(\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_ * y\mapsto\_))$$

All satisfy y↦_

The witness satisfies y↦_

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)) \Longrightarrow (\exists\langle\sigma\rangle.\ \sigma(x \mapsto \_ * y \mapsto \_))$$

Goal:

★ should respect

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\ \_)) \Leftrightarrow (\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_*y\mapsto\ \_))$$

$$(\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_)) \bigstar (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)) \Longrightarrow (\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_*y\mapsto\_))$$

Goal:
★ should respect

∀ quantifiers
(upper bounds)

Hyper Separating Conjunction

$$(\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_))\ \bigstar\ (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\ \_))\Leftrightarrow(\forall\langle\sigma\rangle.\ \sigma(x\mapsto\ \_ * y\mapsto\ \_))$$

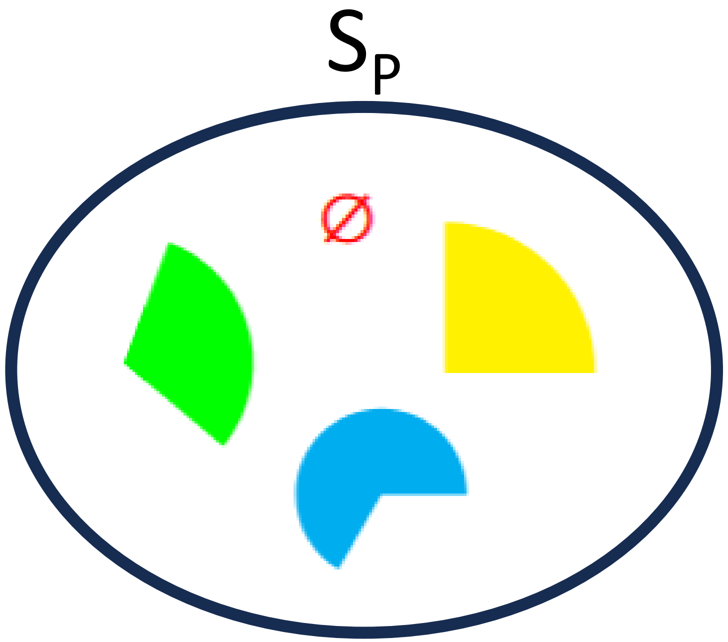
$$(\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_))\ \bigstar\ (\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_))\Longrightarrow(\exists\langle\sigma\rangle.\ \sigma(x\mapsto\_ * y\mapsto\_))$$

Goal:
★ should respect

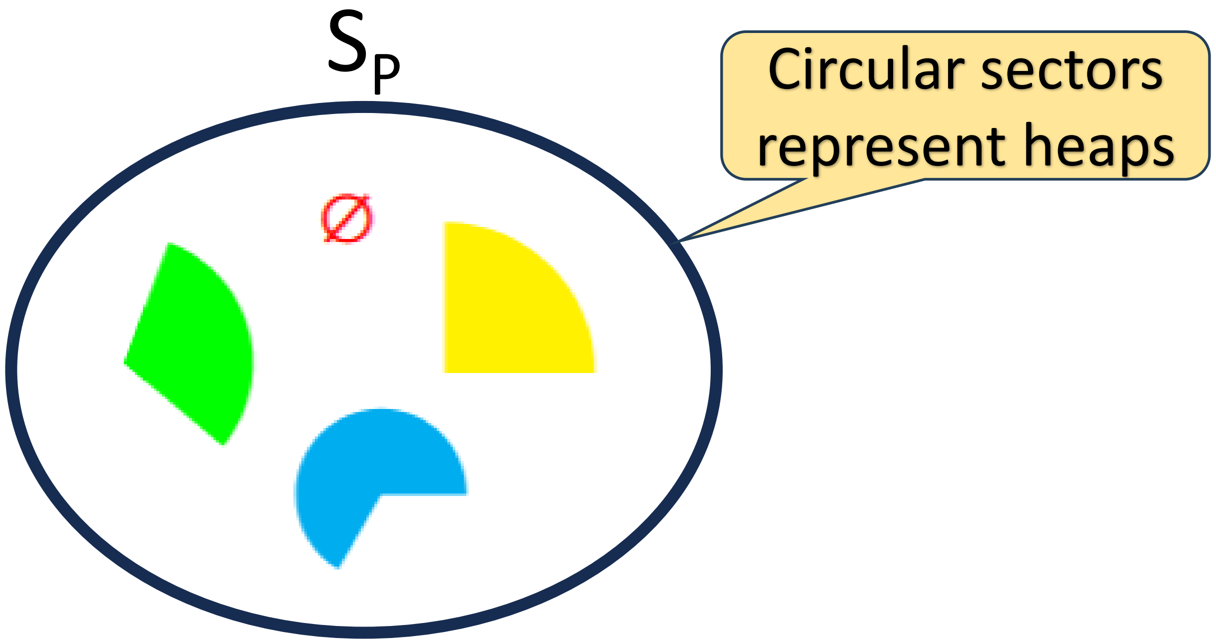
∀ quantifiers
(upper bounds)

∃ quantifiers
(lower bounds)

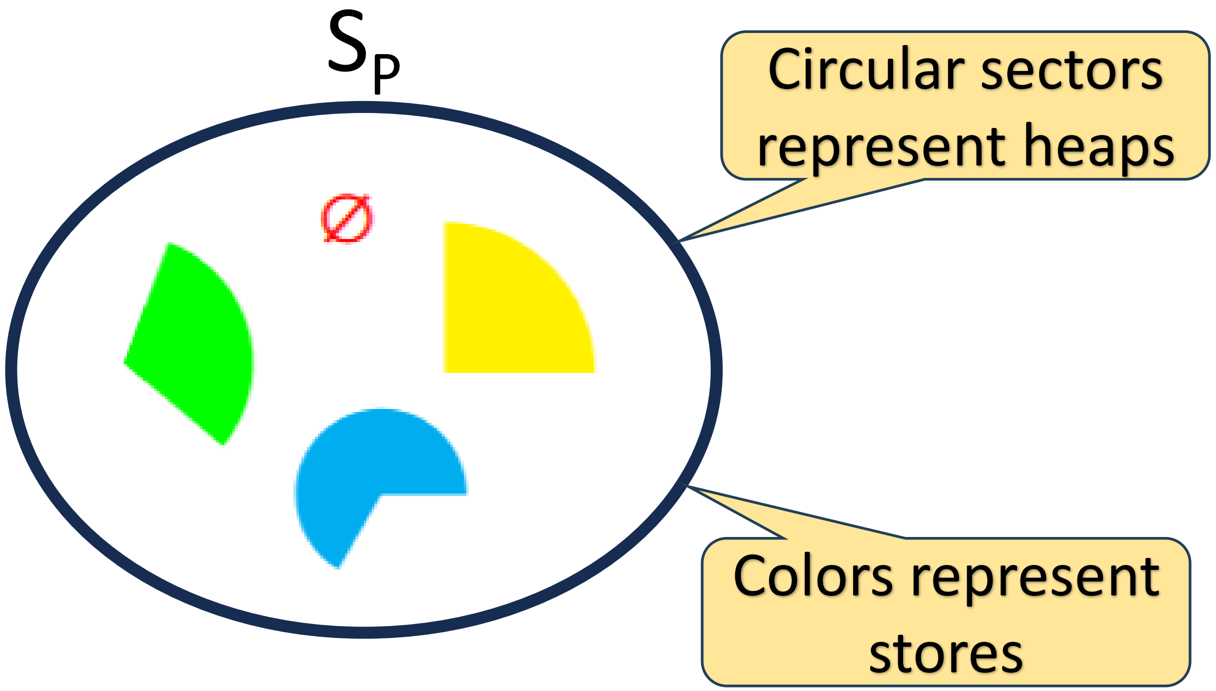
Defining $\oplus$



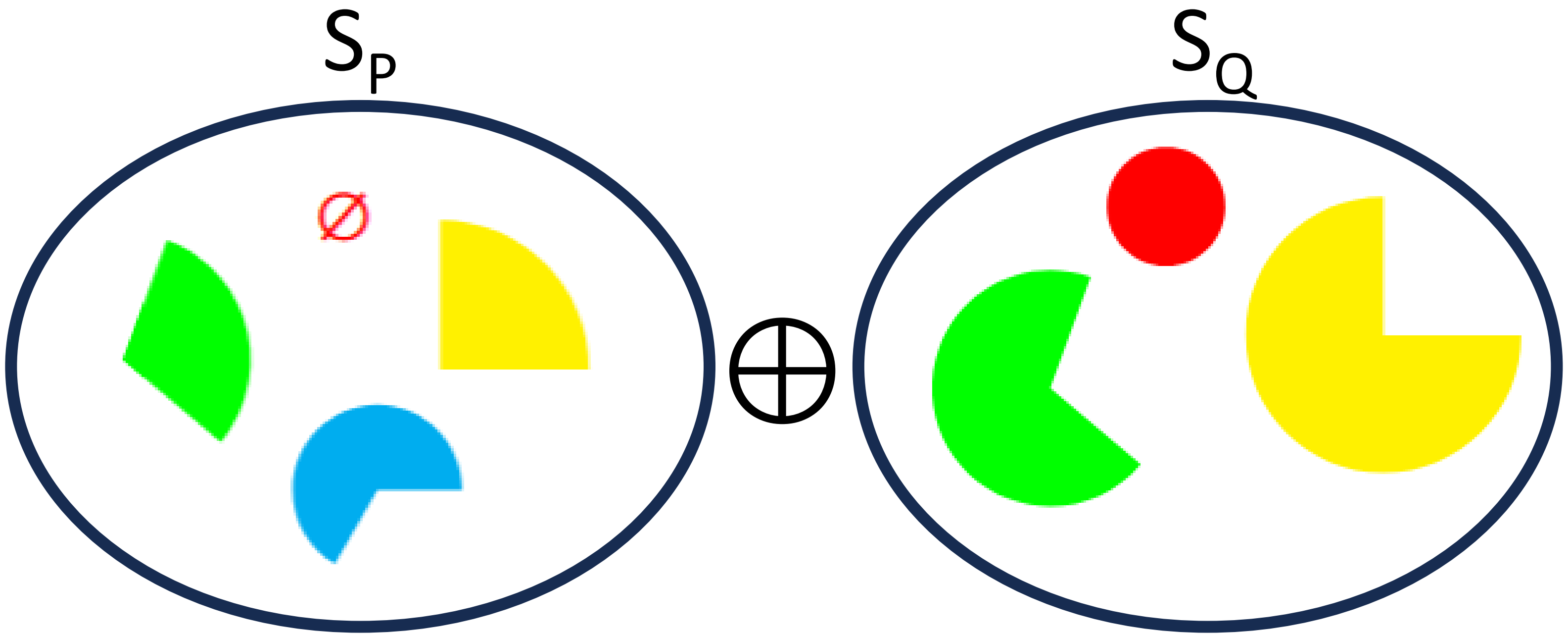
Defining ⊕



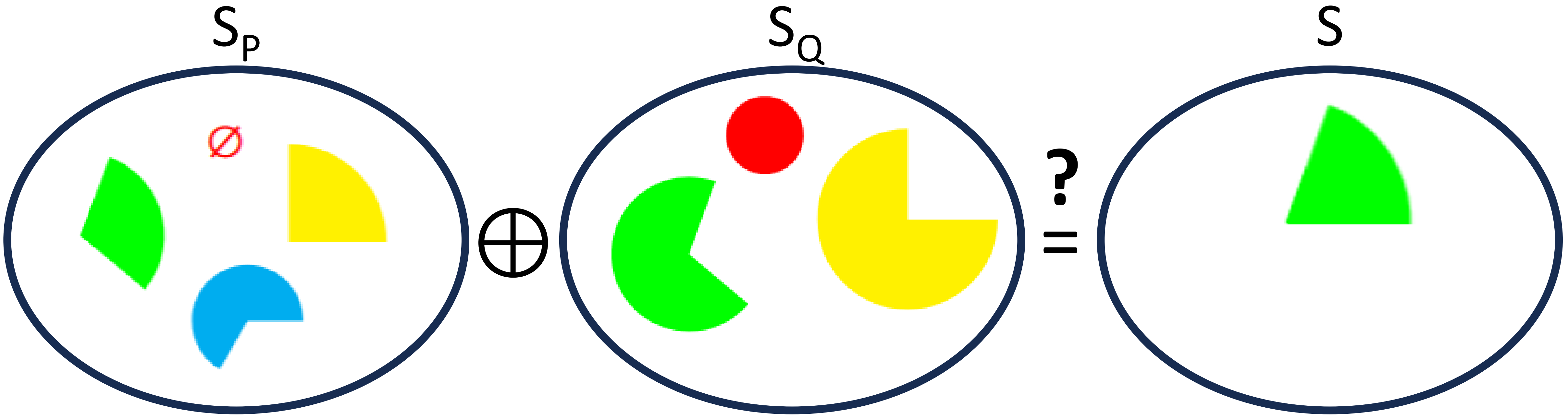
Defining $\oplus$



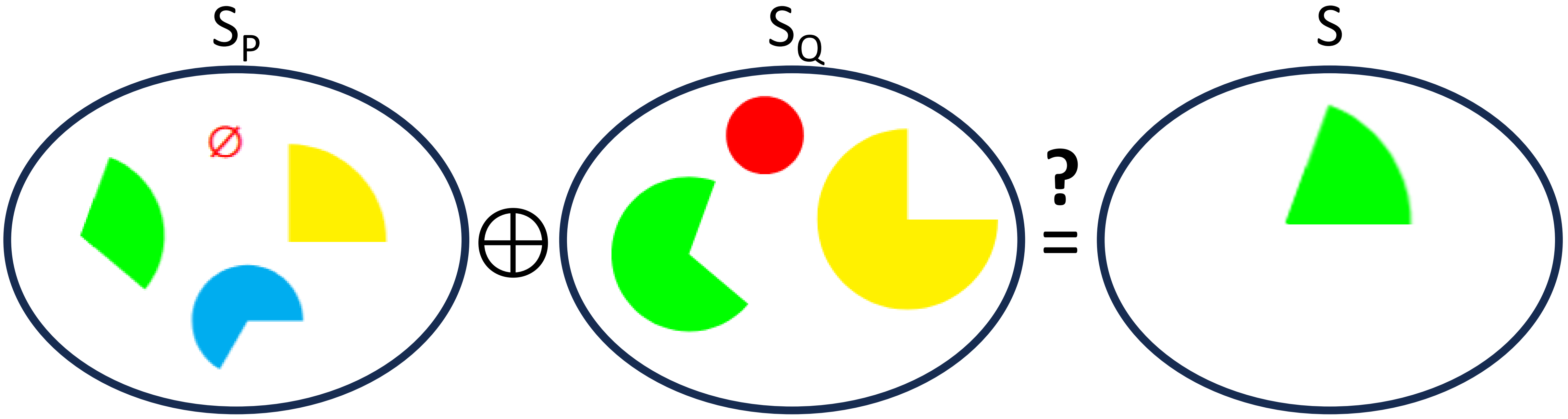
Defining $\oplus$



Defining ⊕

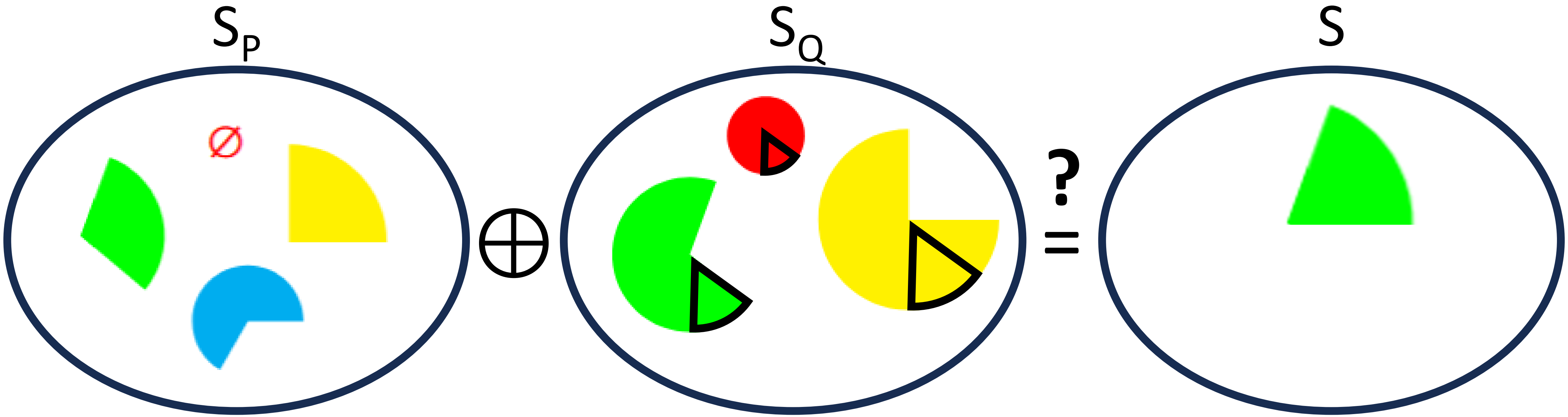


Defining ⊕



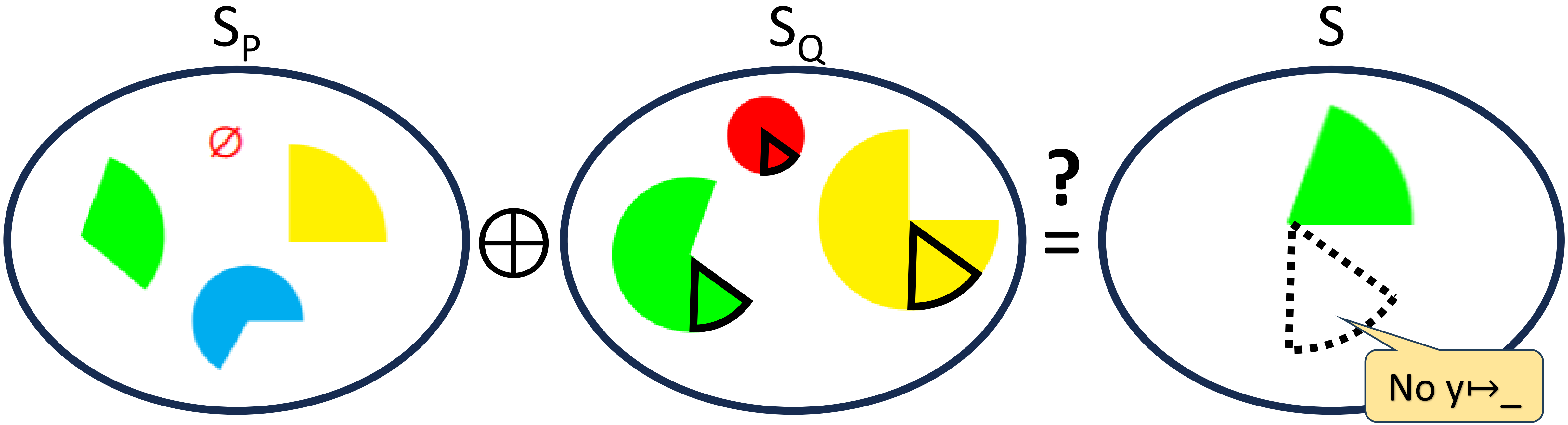
$\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)$

Defining $\oplus$



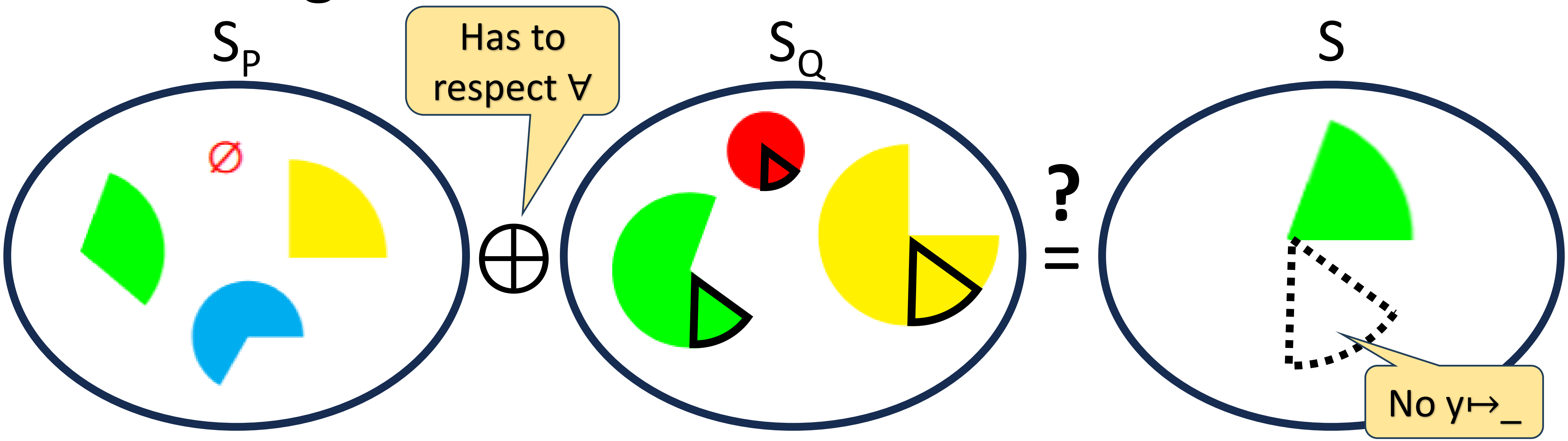
$\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)$

Defining ⊕



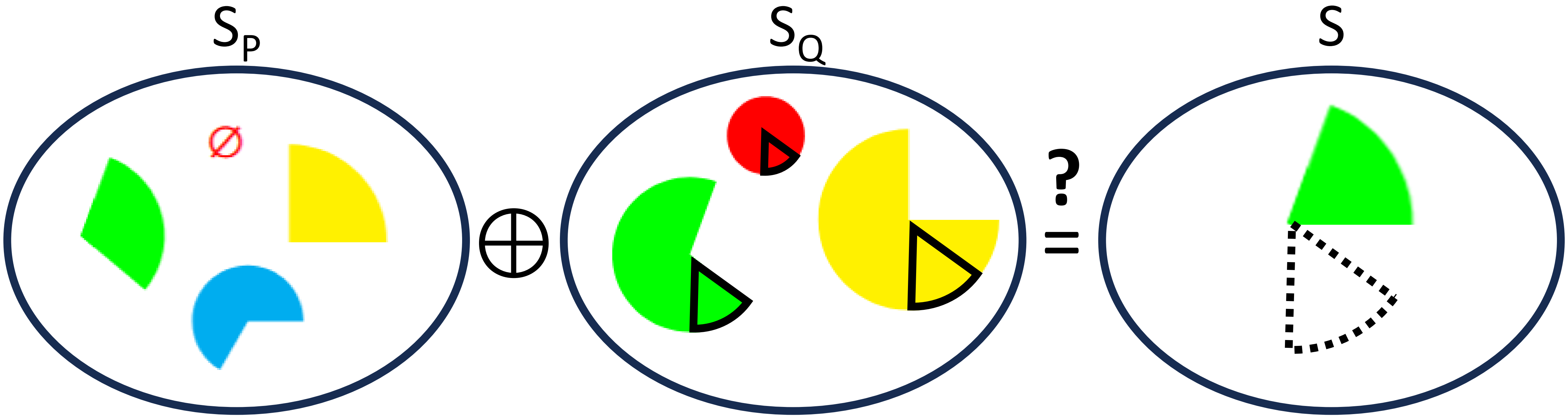
∀⟨σ⟩. σ(y↦_)

Defining ⊕



∀⟨σ⟩. σ(y↦_)

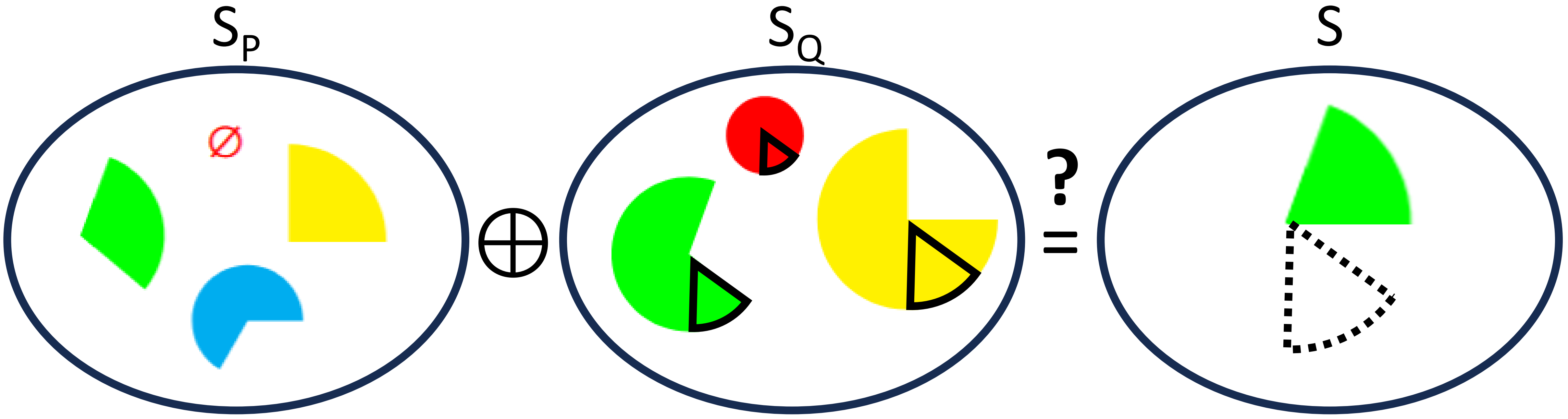
Defining $\oplus$



$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

Defining ⊕

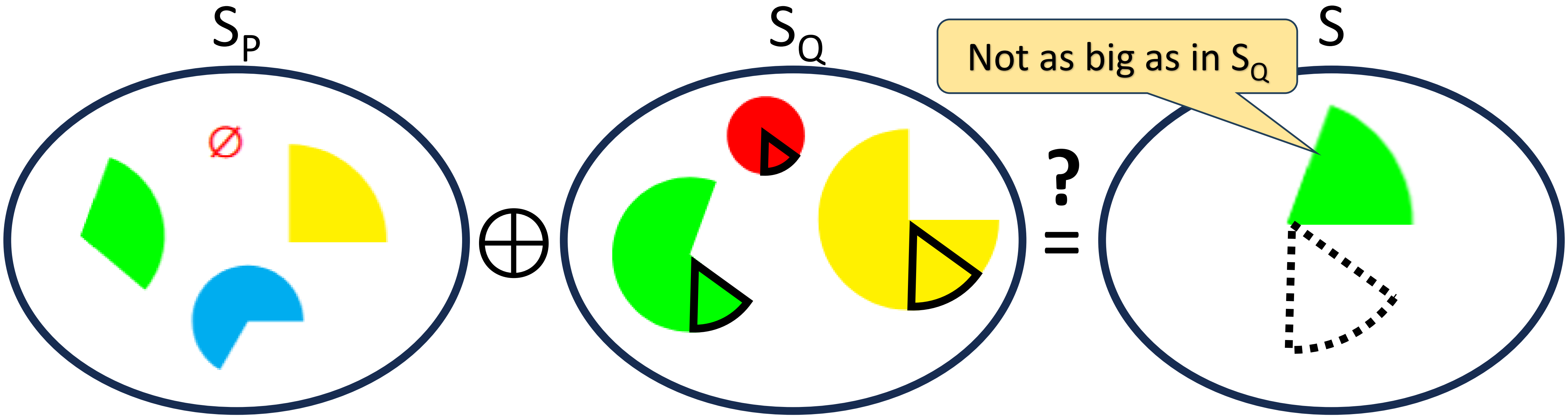


All states in S should be at least as big as S_Q

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

Defining $\oplus$

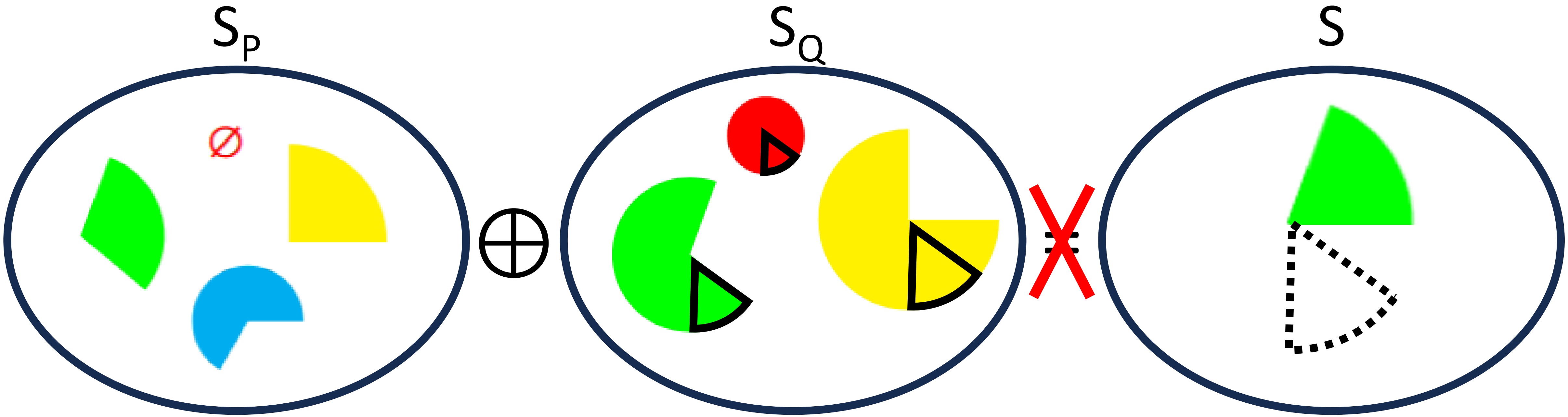


All states in S should be at least as big as S_Q

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

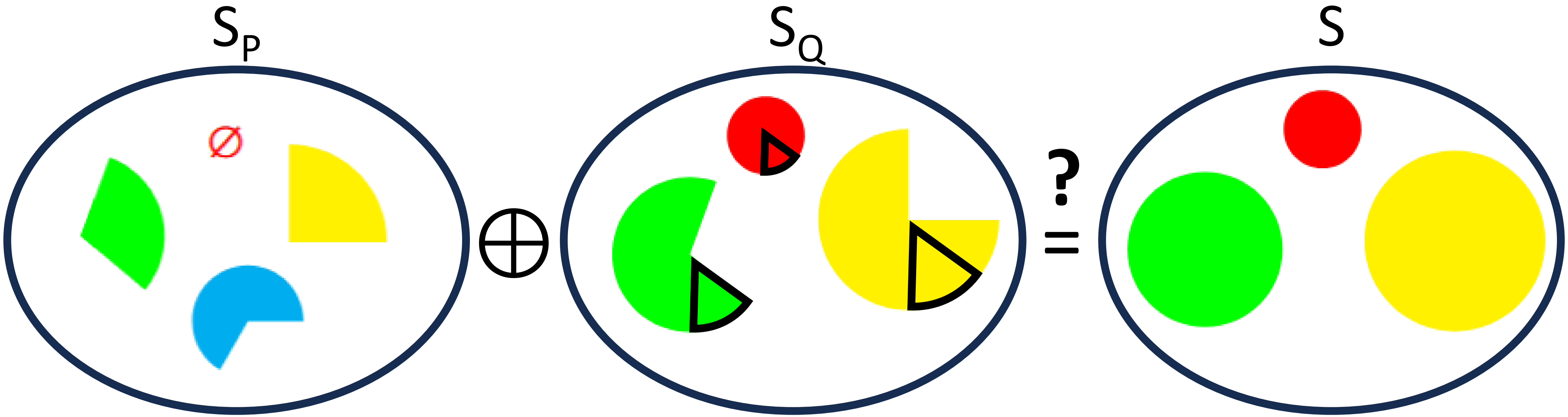
Defining $\oplus$



$\forall\langle\sigma\rangle.\ \sigma(y \mapsto \_)$

$S \subseteq S_P * S_Q$

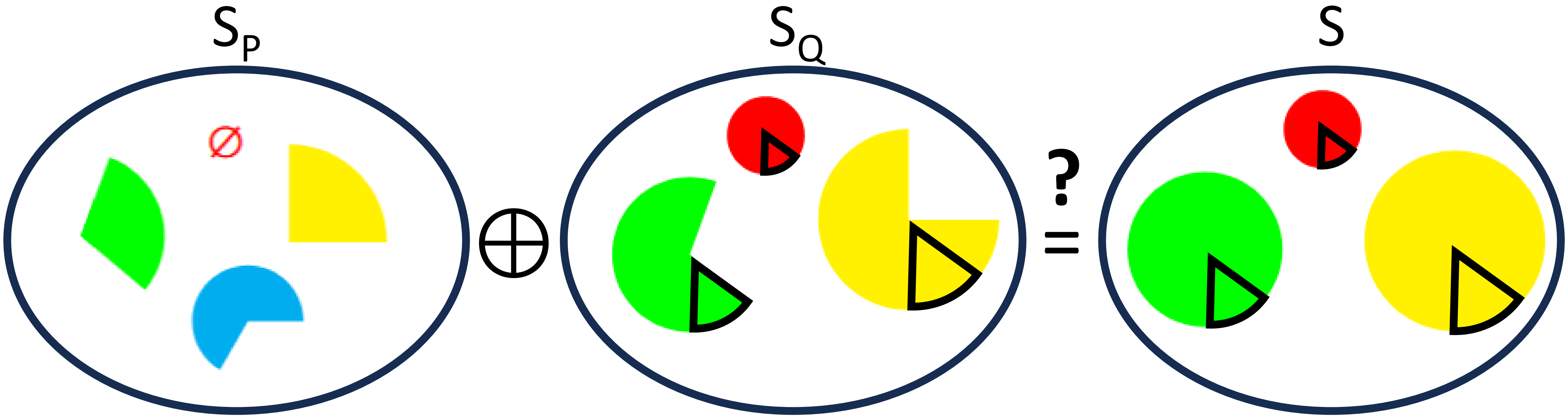
Defining $\oplus$



$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

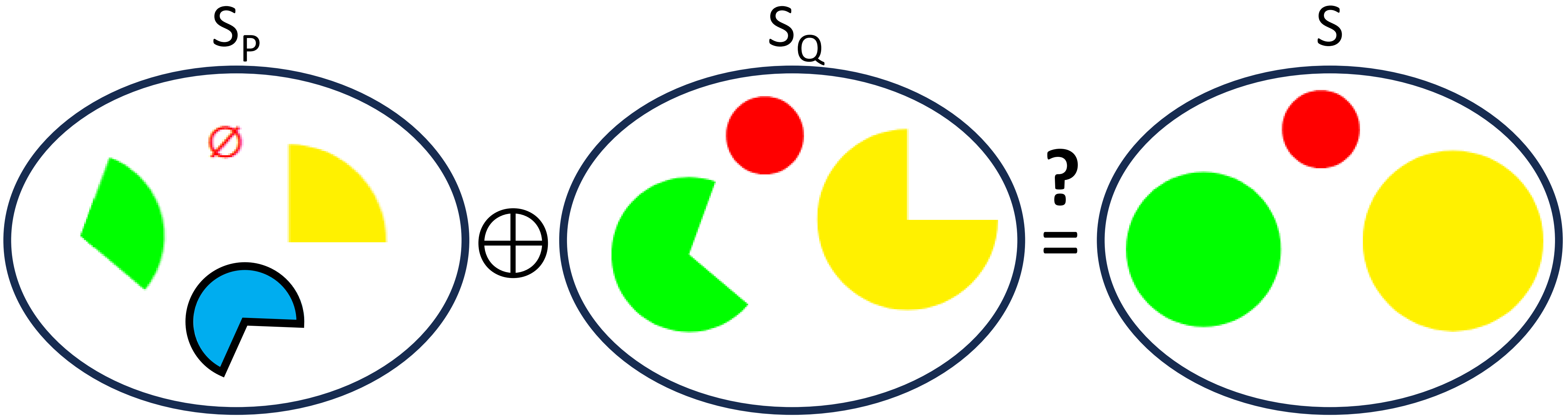
Defining $\oplus$



$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

Defining $\oplus$

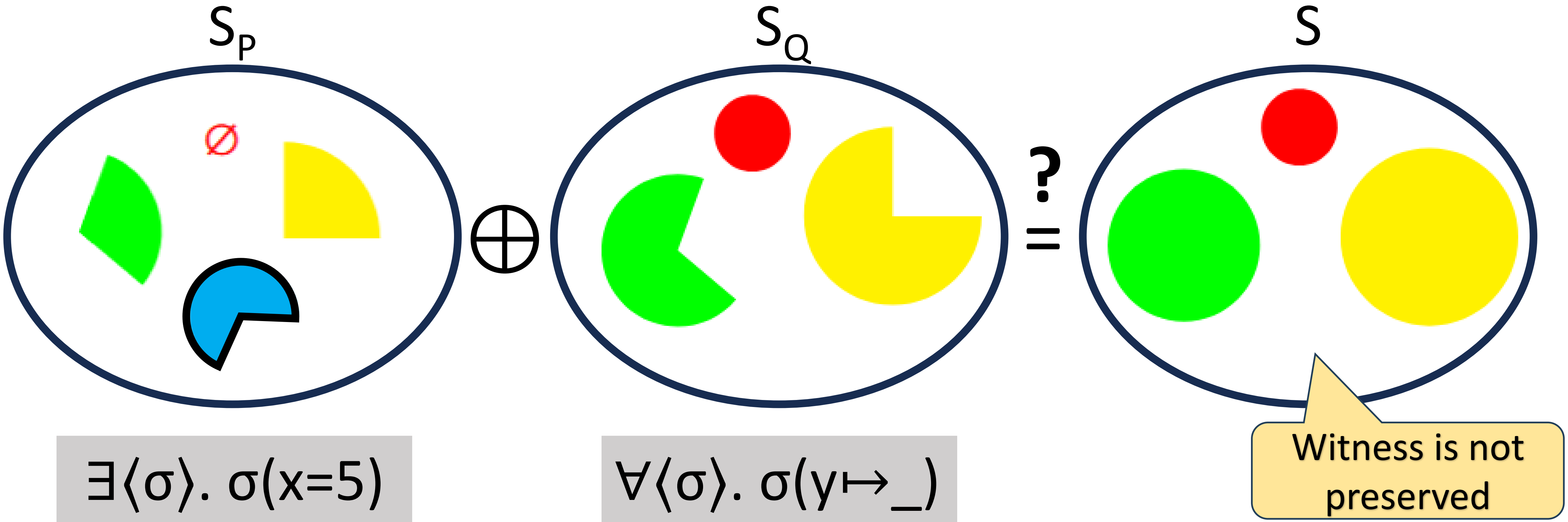


$\exists\langle\sigma\rangle.\ \sigma(x=5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

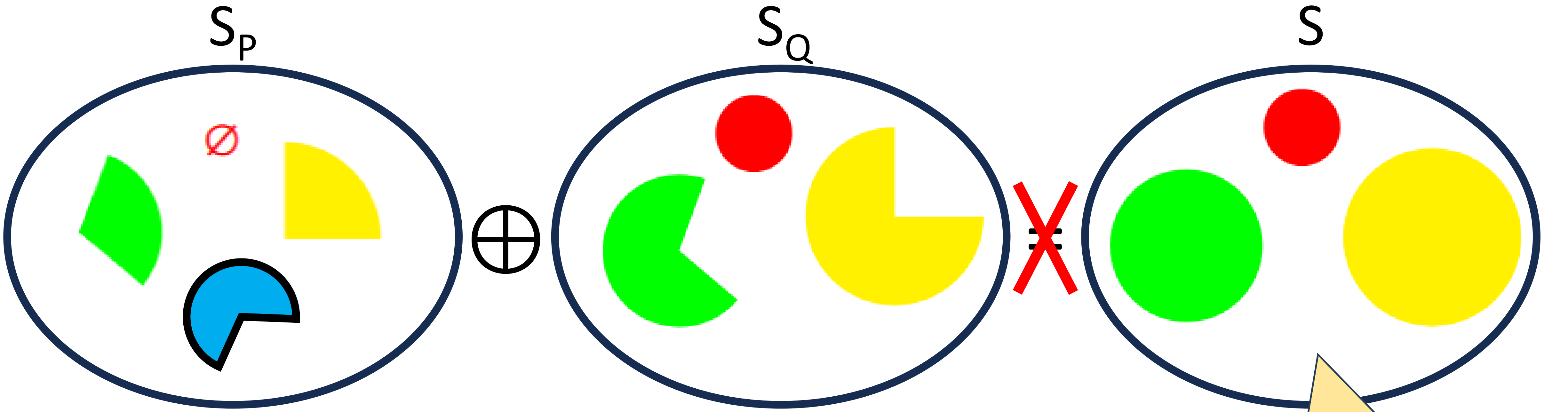
$S \subseteq S_P * S_Q$

Defining $\oplus$



$S \subseteq S_P * S_Q$

Defining ⊕

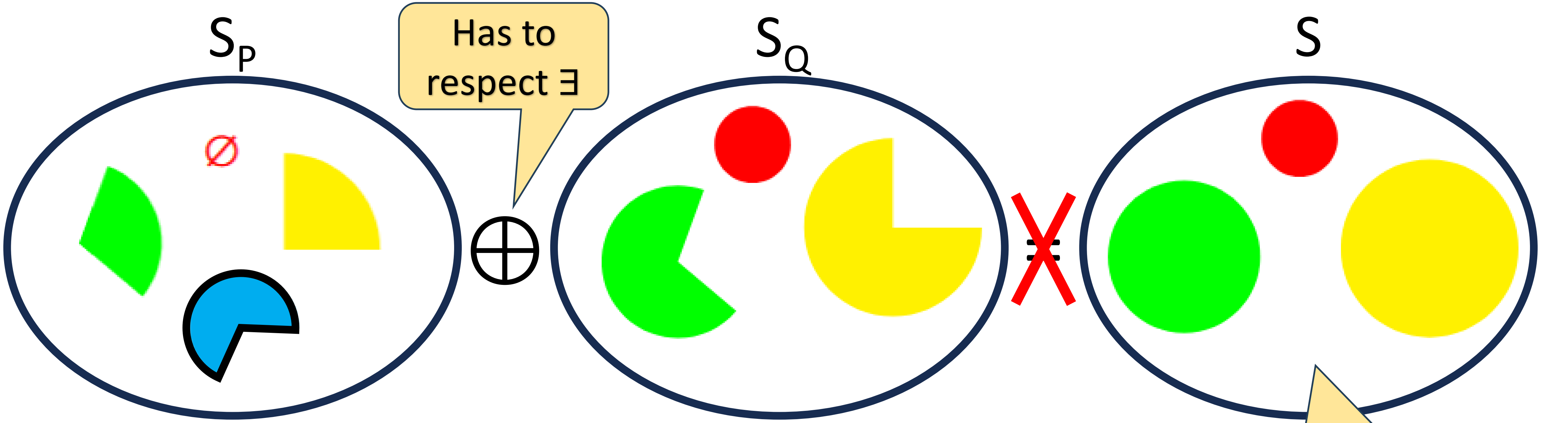


$\exists\langle\sigma\rangle.\ \sigma(x=5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

Defining $\oplus$

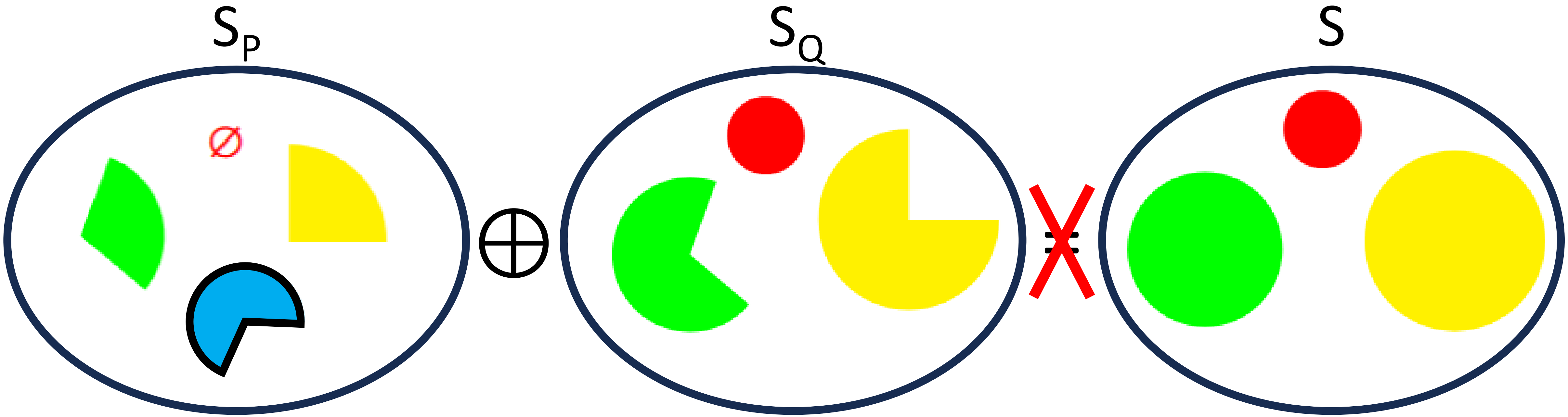


$\exists\langle\sigma\rangle.\ \sigma(x=5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q$

Defining $\oplus$

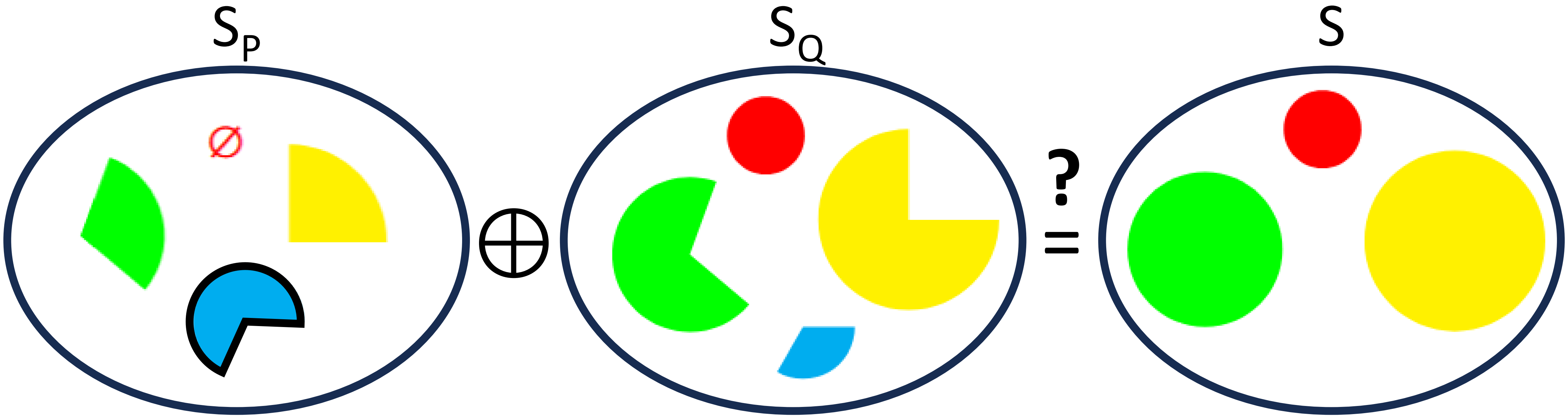


$\exists\langle\sigma\rangle.\ \sigma(x=5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q \wedge \mathbf{plw}(S, S_P, S_Q)$

Defining ⊕

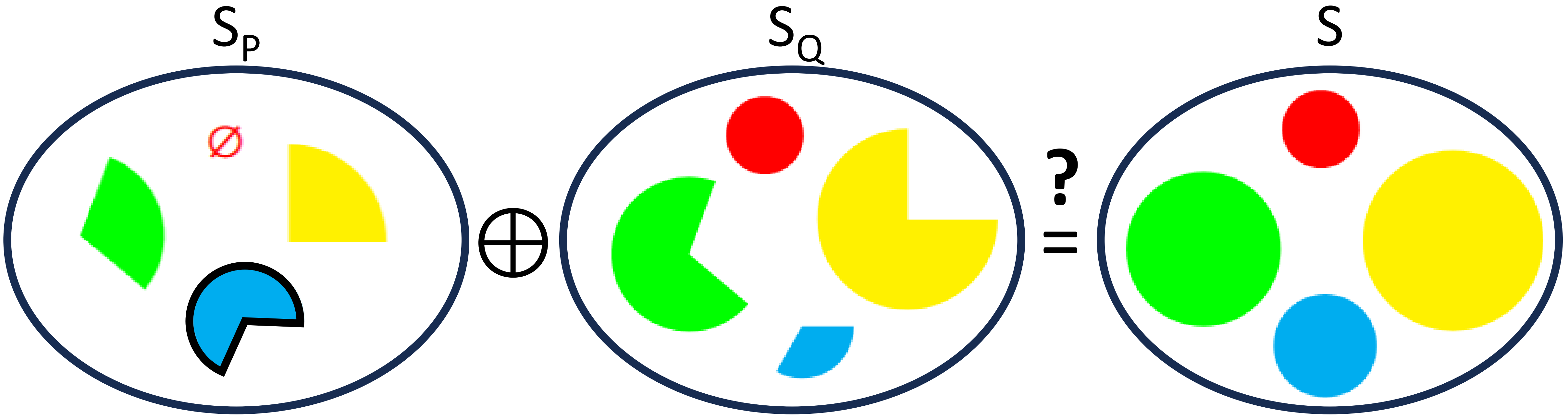


$\exists\langle\sigma\rangle.\ \sigma(x{=}5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q \land$ **plw(S, S_P, S_Q)**

Defining $\oplus$

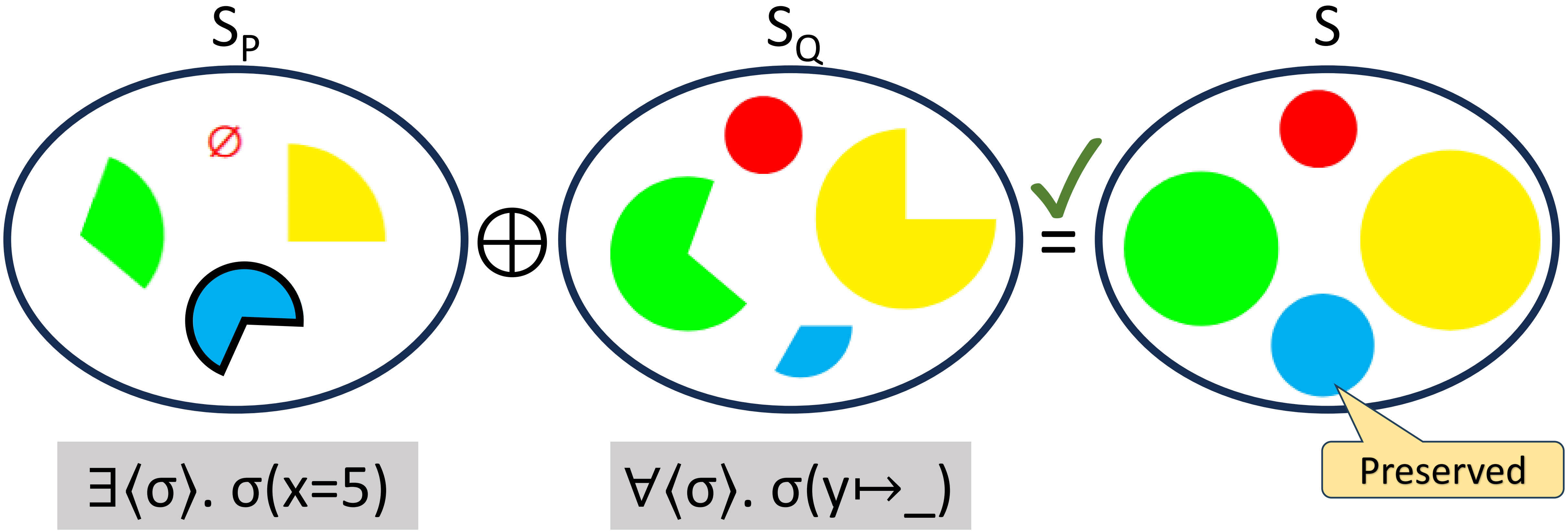


$\exists\langle\sigma\rangle.\ \sigma(x=5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

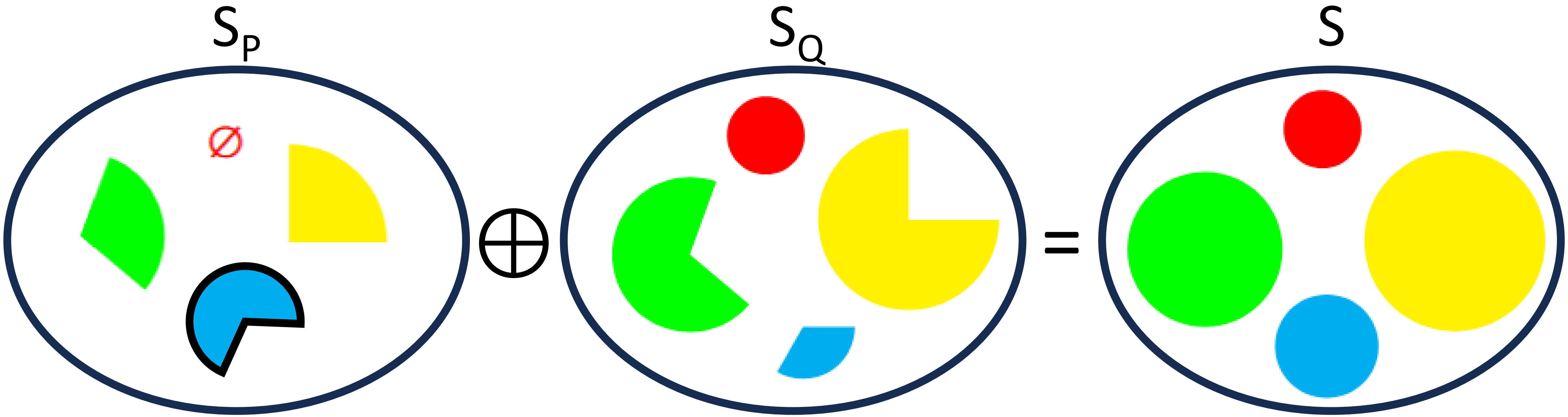
$S \subseteq S_P * S_Q \wedge$ **plw**(S, S_P, S_Q)

Defining $\oplus$



$S \sqsubseteq S_P * S_Q \wedge \mathbf{plw}(\mathbf{S}, \mathbf{S_P}, \mathbf{S_Q})$

Defining ⊕



$\exists\langle\sigma\rangle.\ \sigma(x{=}5)$

$\forall\langle\sigma\rangle.\ \sigma(y\mapsto\_)$

$S \subseteq S_P * S_Q \land \mathrm{plw}(S, S_P, S_Q) \land \mathbf{prw(S, S_P, S_Q)}$

More in the Paper

We identify and solve two non-locality problems

More in the Paper

We identify and solve two non-locality problems

Allocation

More in the Paper

We identify and solve two non-locality problems

Allocation

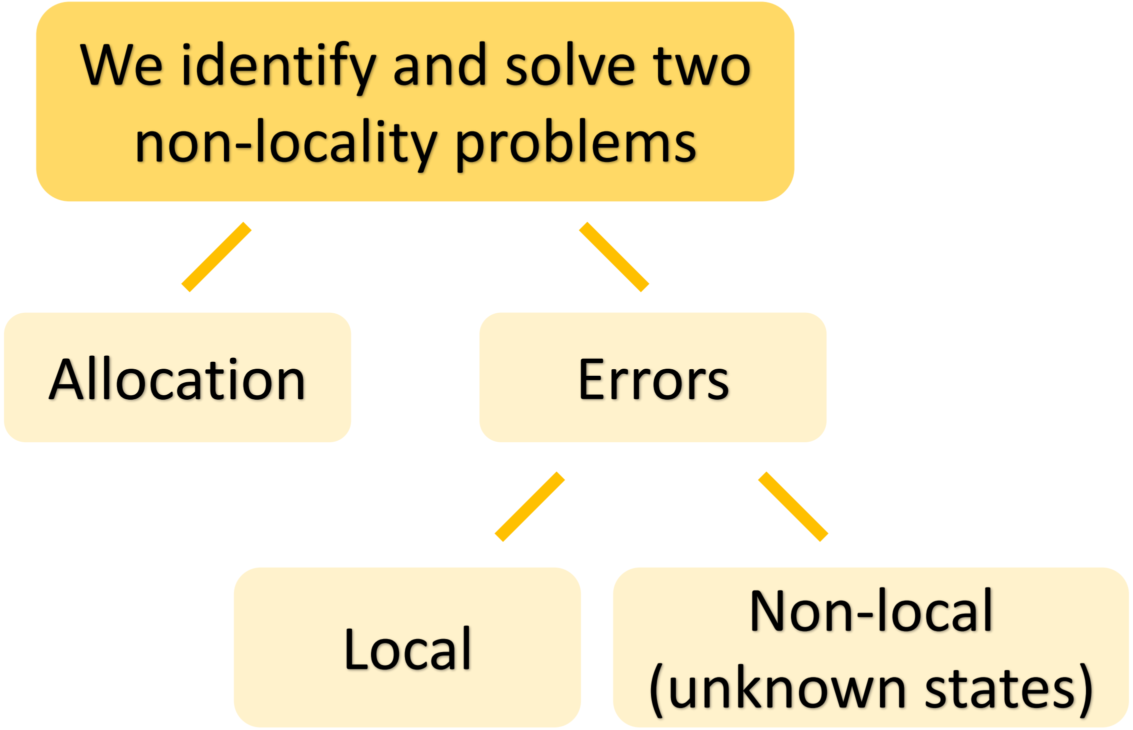
Errors

More in the Paper

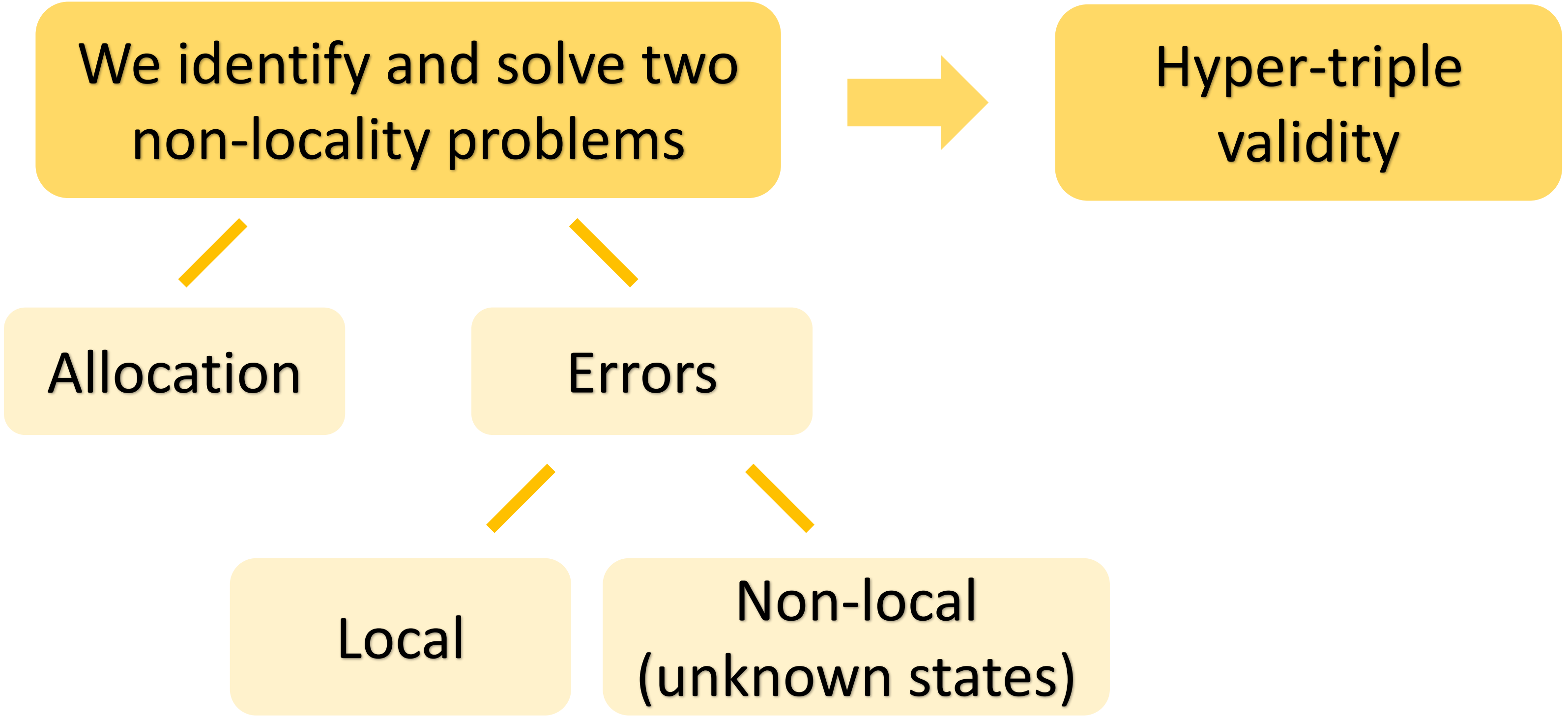
We identify and solve two non-locality problems

- Allocation
- Errors
  - Local

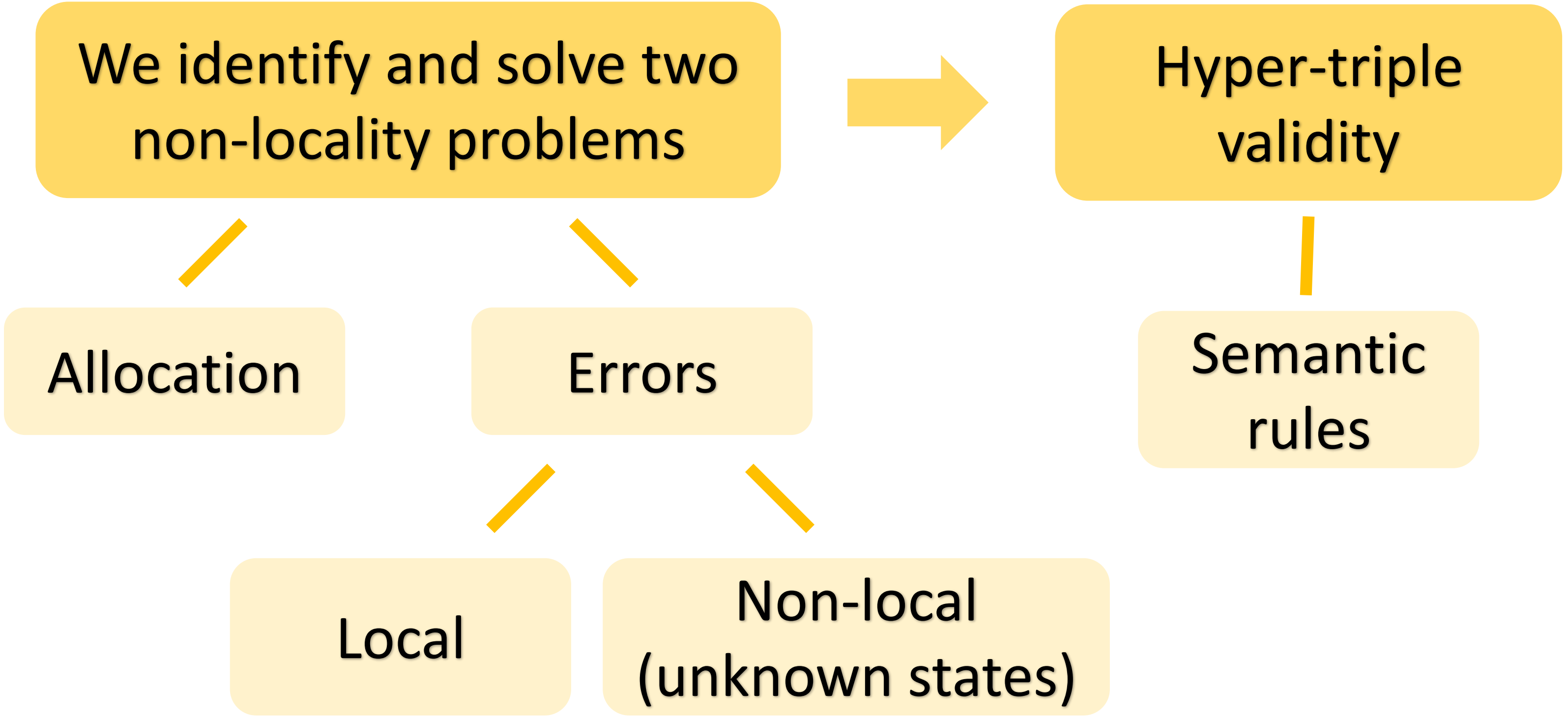
More in the Paper



More in the Paper



More in the Paper



More in the Paper

We identify and solve two non-locality problems

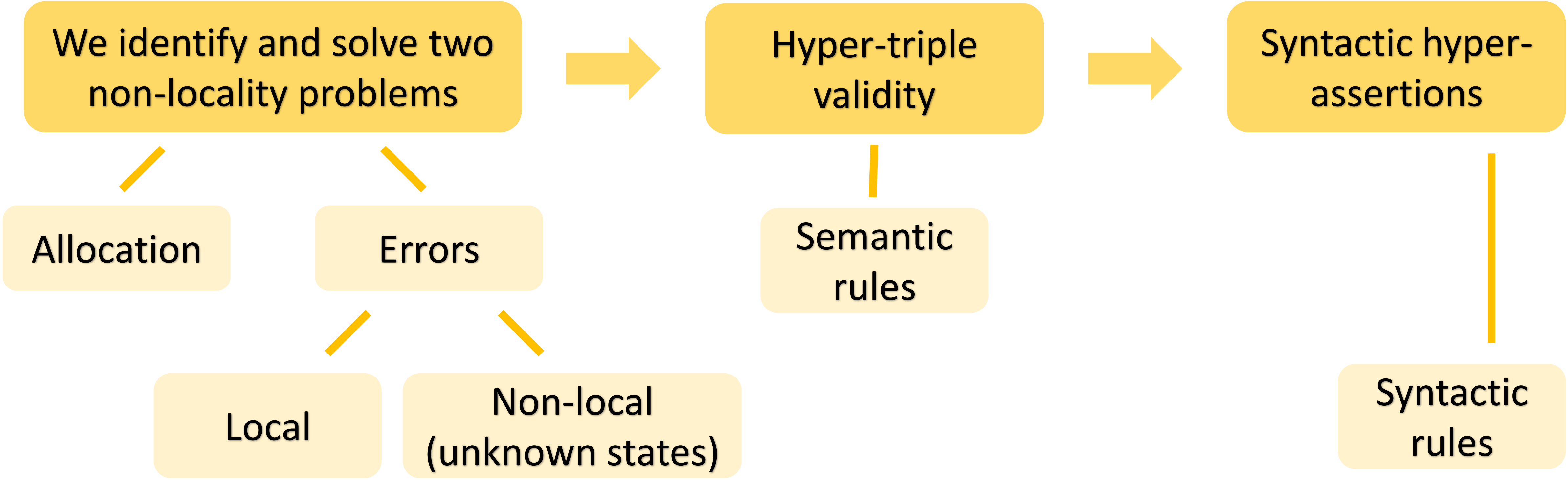
- Allocation
- Errors
  - Local
  - Non-local (unknown states)

Hyper-triple validity

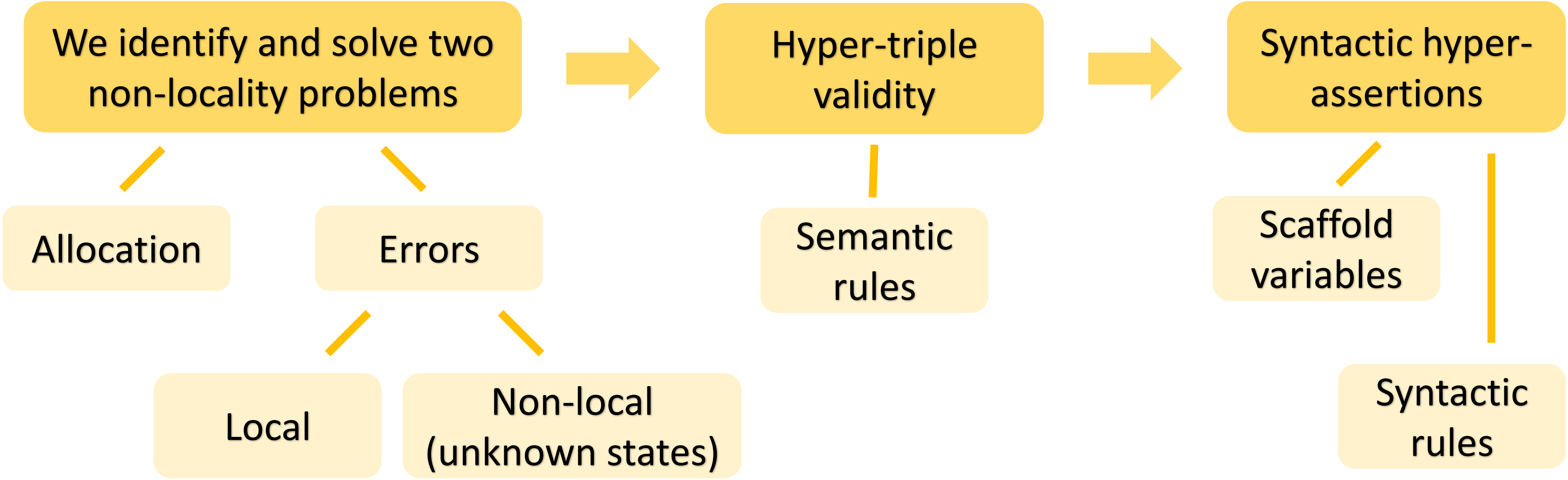
- Semantic rules

Syntactic hyper-assertions

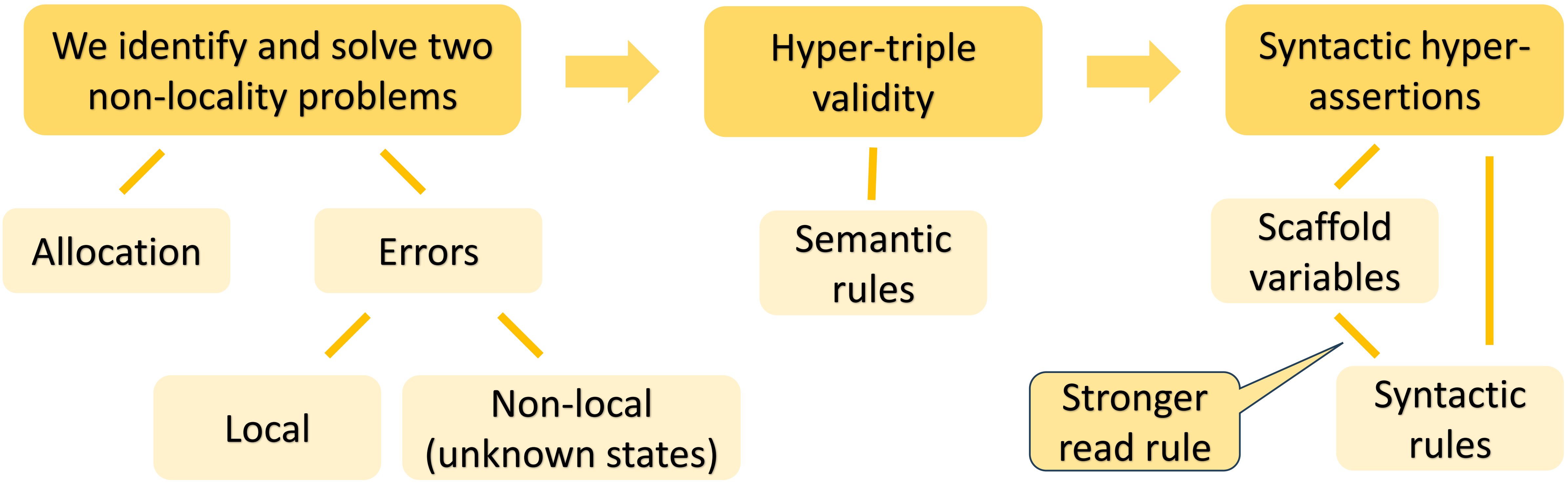
More in the Paper



More in the Paper



More in the Paper



More in the Paper

- We identify and solve two non-locality problems
  - Allocation
  - Errors
    - Local
    - Non-local (unknown states)
- Hyper-triple validity
  - Semantic rules
- Syntactic hyper-assertions
  - Scaffold variables
    - Syntactic rules
  - Syntactic rules

Key Takeaways & Summary

Hyper Separation Logic

- Can reason about arbitrary complex hyperproperties
- Generalized Frame rule
- Machine-checked soundness

Key Challenges

- Designing a ★ that respects ∀ and ∃
- Recovering framing for non-local behavior
  - Non-local errors (unknown states)
  - Allocation (via embedding the frame)