CMU 15-414

Bug Catching: Automated Program Verification and Testing

Abstract Interpretation

9 Nov 2011

Soonho Kong

soonhok@cs.cmu.edu

Problem, Motivation, and Big Picture

“Software is Hard.”

"Software is Hard."

- Donald E. Knuth

Software Verification is Harder:

Software Verification is Harder:

"Any **non-trivial** property about the language recognized by a Turing machine is **undecidable**."

- Rice's Theorem

- Large / unbounded base types: int, float, string
- User-defined types / classes
- Pointers/aliasing + unbounded #'s of heap allocated cells
- Procedure calls / recursion / calls through pointers / dynamic method lookup / overloading
- Concurrency + unbounded #'s of threads
- Templates / generics / include files
- Interrupts / exceptions / callbacks
- Use of secondary storage: files, databases
- Absent source code for: libraries, system calls, mobile code
- Esoteric features: continuations, self-modifying code
- Size (e.g., MS Word = 1.4 MLOC)

"In the development of the understanding of complex phenomena, the most powerful tool available to the human intellect is   ."

- C. A. R. Hoare

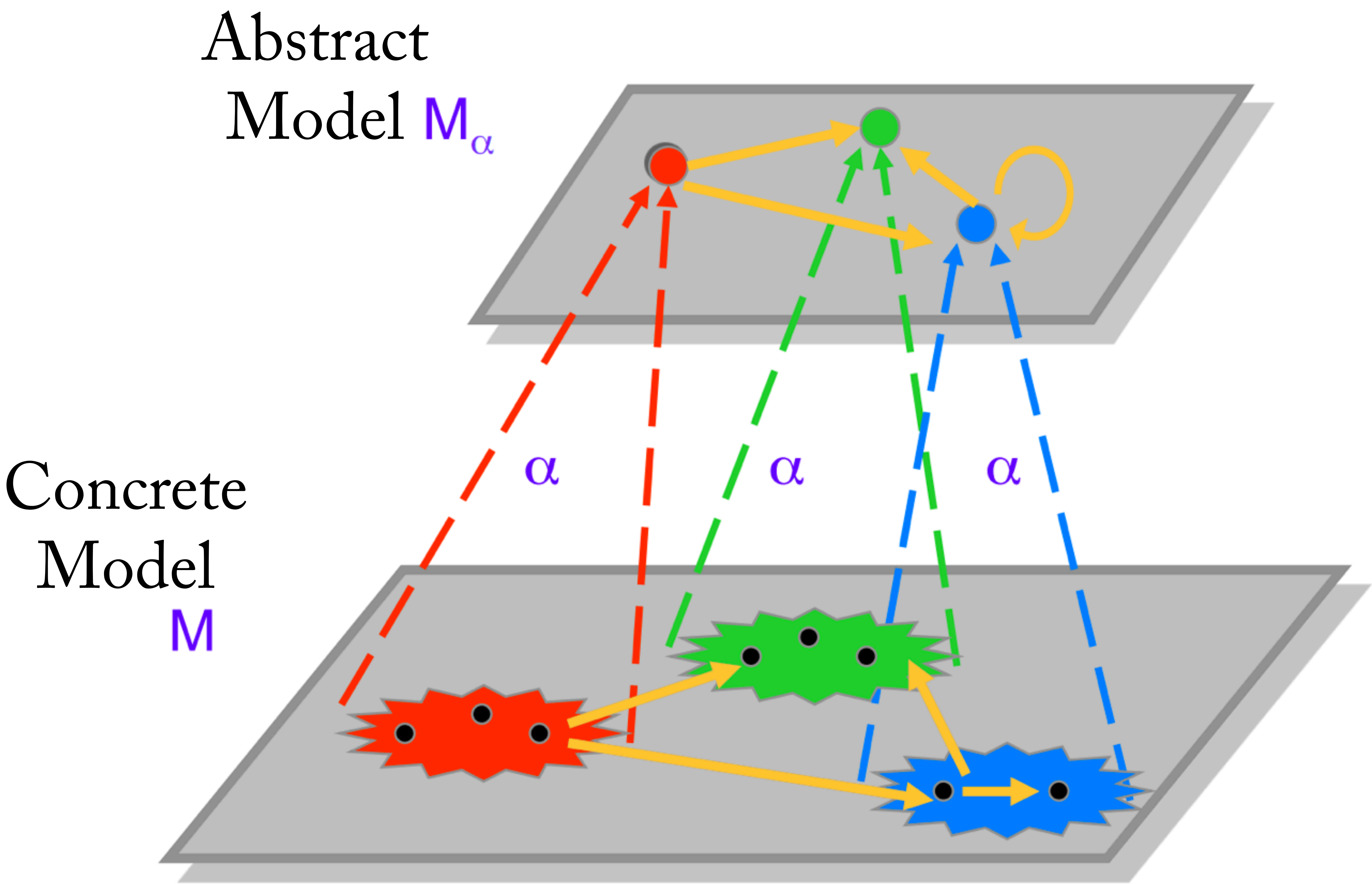
"In the development of the understanding of complex phenomena, the most powerful tool available to the human intellect is **abstraction**."

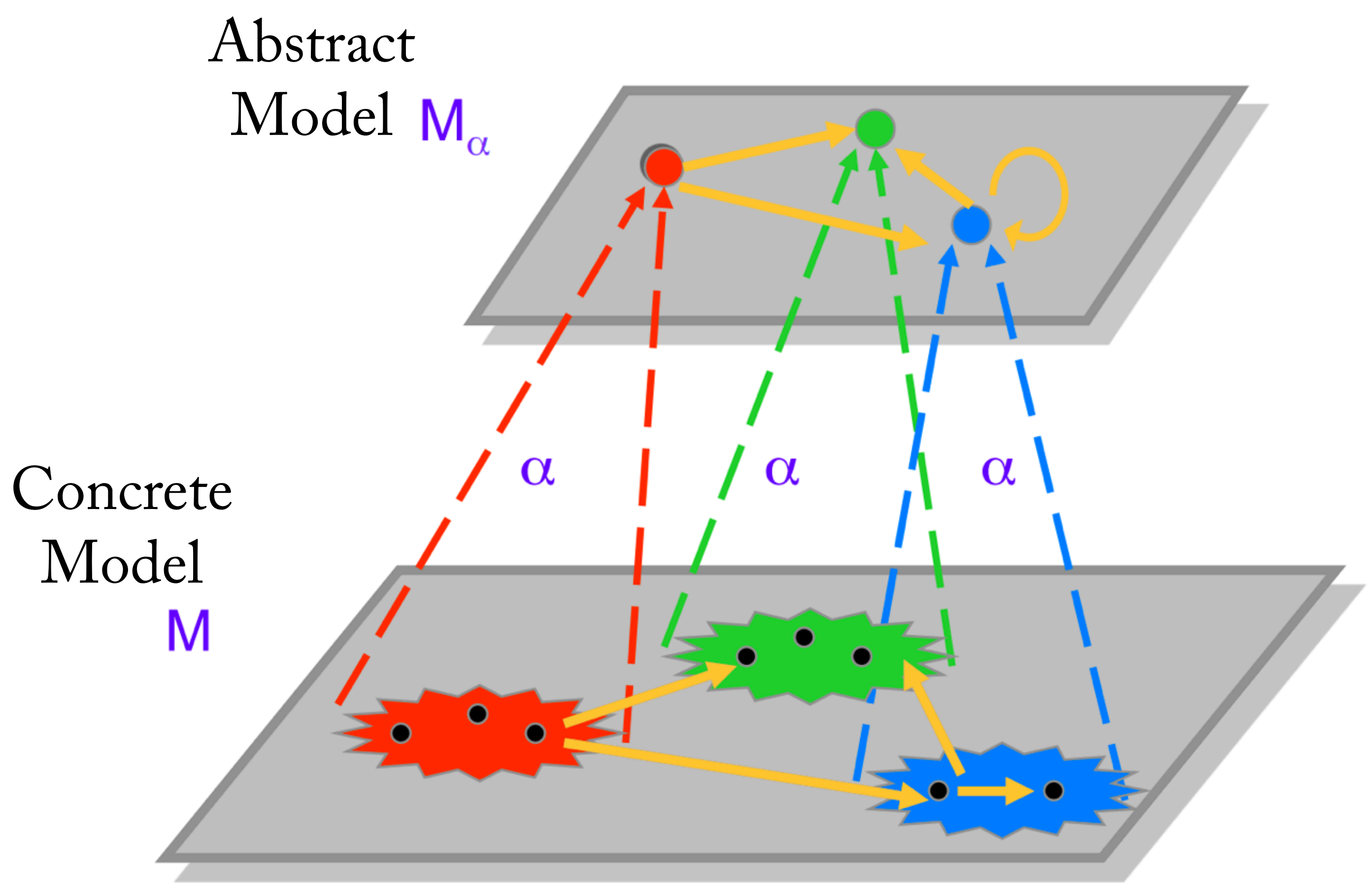
- C. A. R. Hoare

"The purpose of **abstraction** is not to be vague, but to create a new semantic level in which one can be absolutely precise."

- Edsger W. Dijkstra

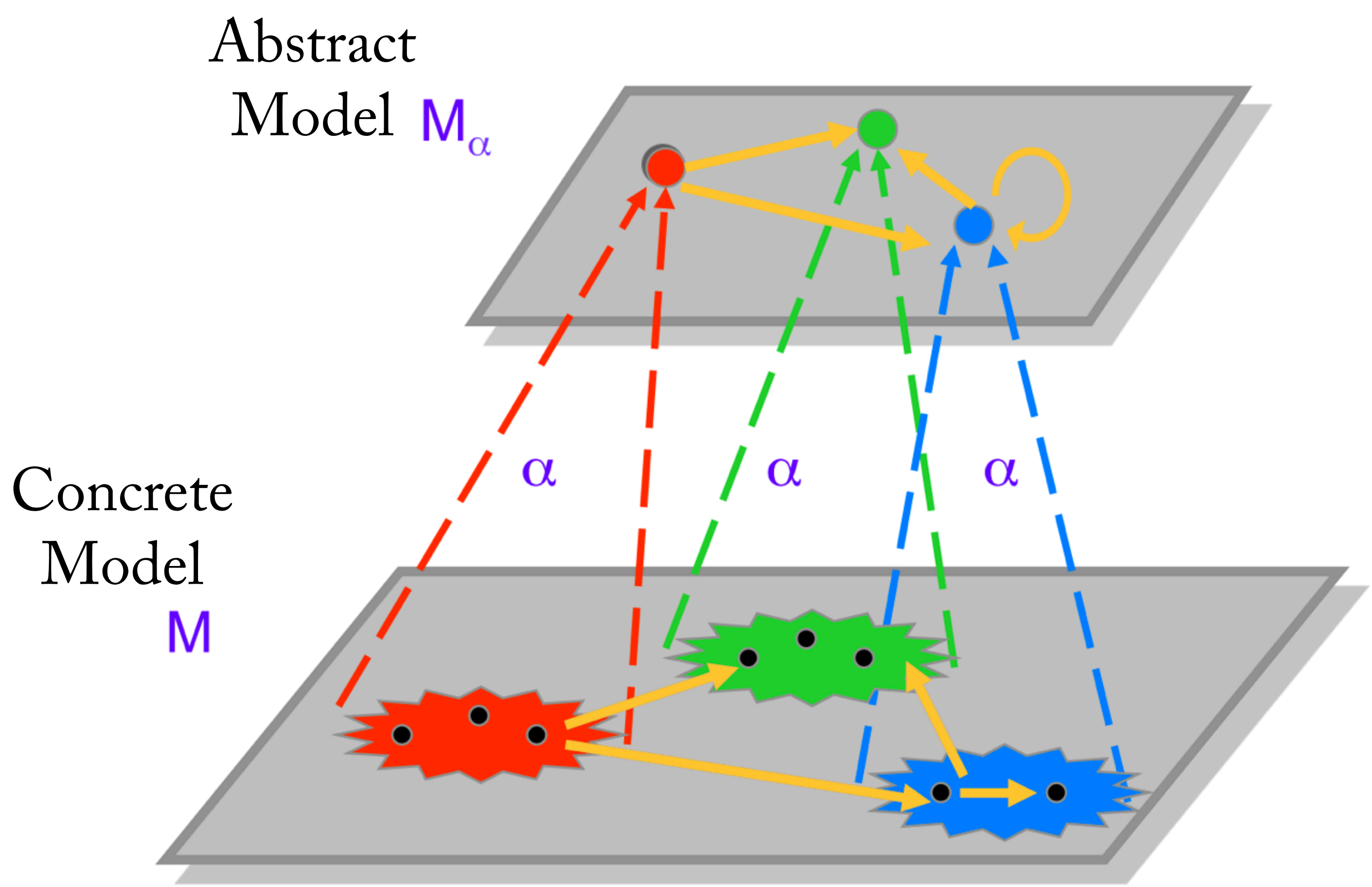
What does Abstraction mean to Model Check Software?





What do we expect from Abstraction?

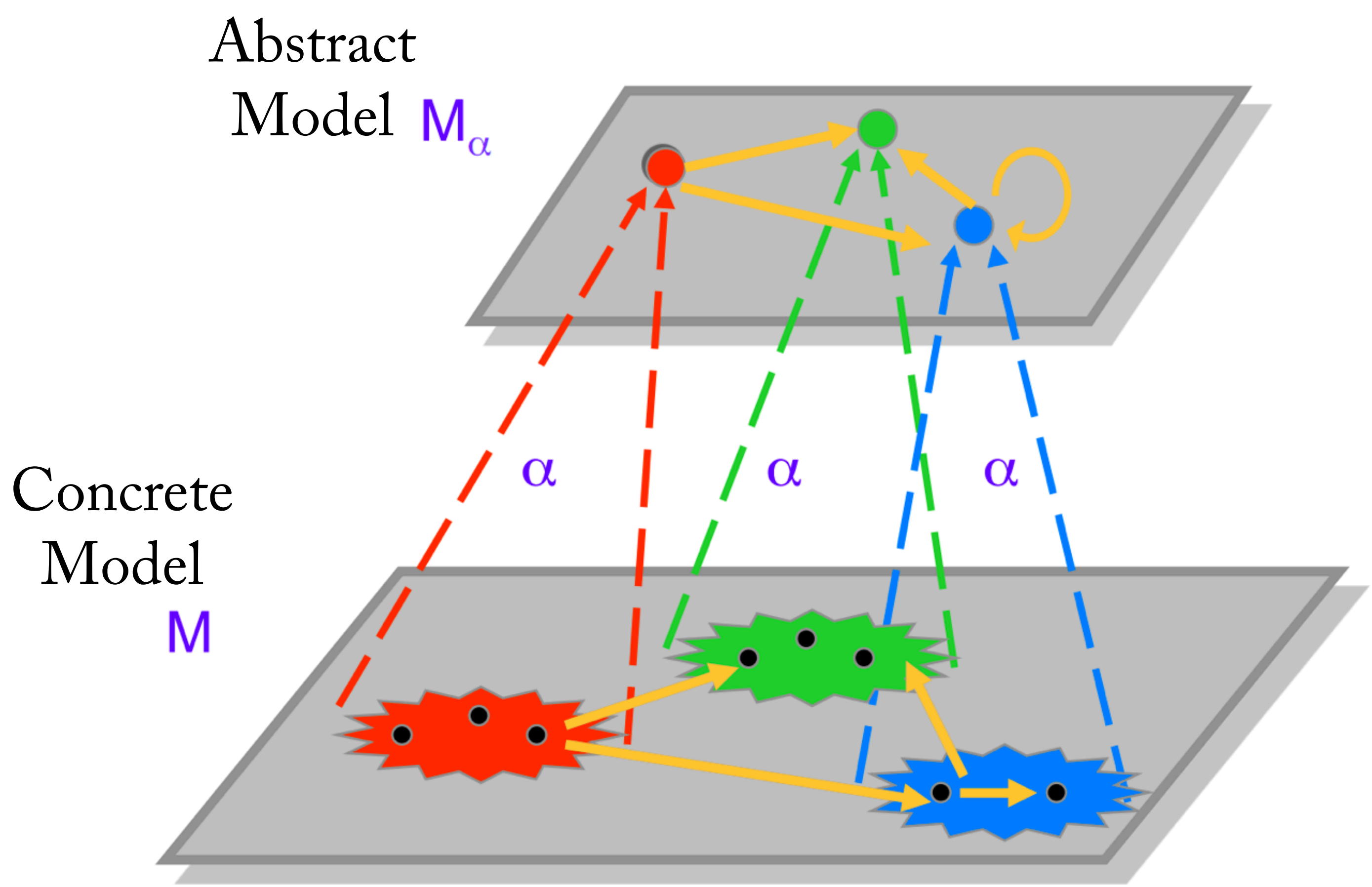
$$\hat{\mathcal{M}} \models \hat{\phi} \Longleftrightarrow \mathcal{M} \models \phi$$



What do we expect from Abstraction?

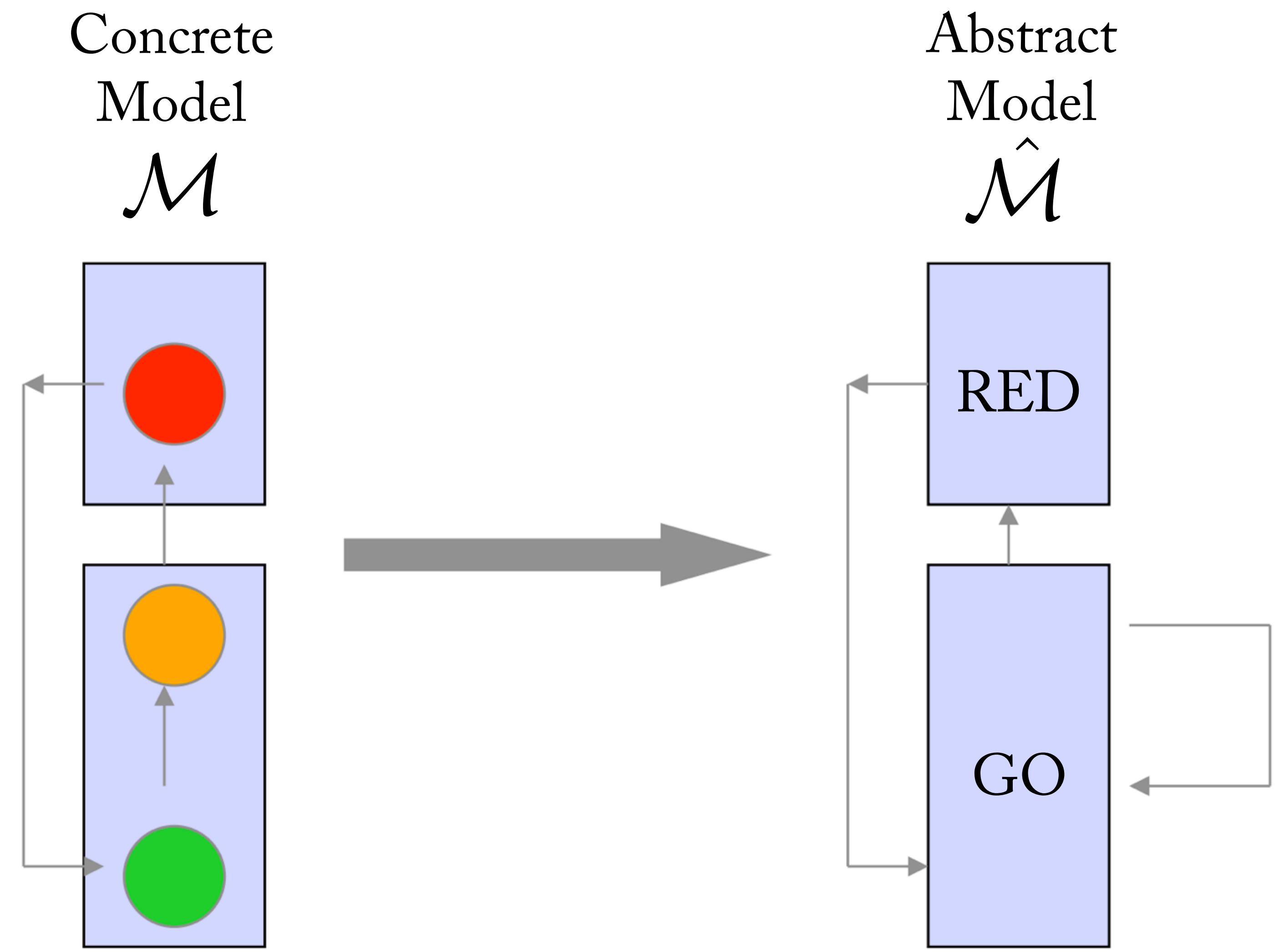
$$\hat{\mathcal{M}} \models \hat{\phi} \implies \mathcal{M} \models \phi$$ ✔

Preservation Theorem (Clarke, Grumberg, Long)
If property(ACTL*) holds on abstract model, it holds on concrete model



What do we expect from Abstraction?

$$\hat{\mathcal{M}} \not\models \hat{\phi} \implies \mathcal{M} \not\models \phi$$ ✗

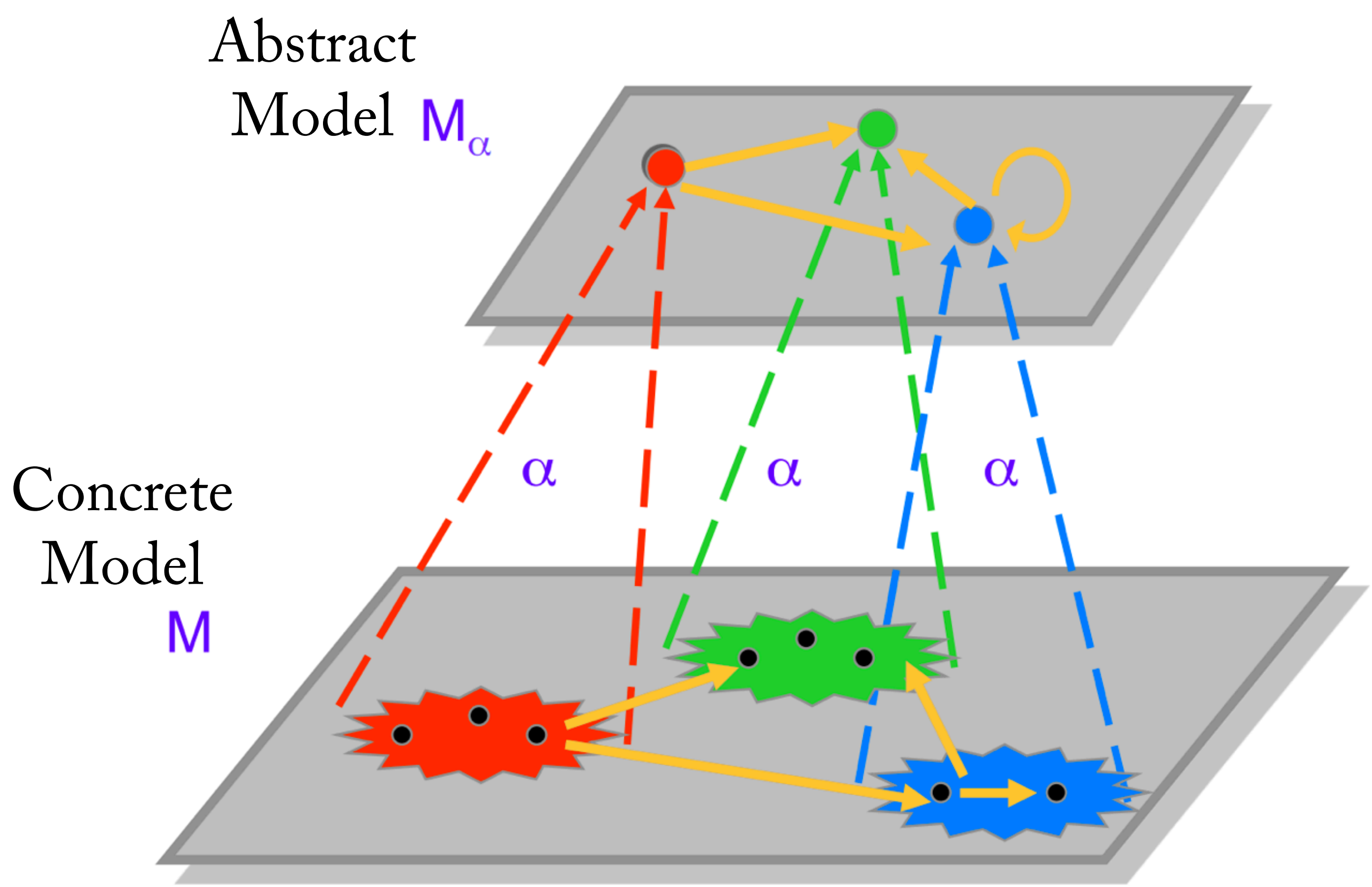


AG AF ● ✓ AG AF RED ✗

What do we expect from Abstraction?

$$\hat{\mathcal{M}} \not\models \hat{\phi} \implies \mathcal{M} \not\models \phi$$ ✗

Spurious Counterexample: <RED> <GO> <GO> <GO> ...
Artifact of the abstraction!



What we should expect from Abstraction:

$$\hat{\mathcal{M}} \models \hat{\phi} \implies \mathcal{M} \models \phi$$ ✔

$$\hat{\mathcal{M}} \not\models \hat{\phi} \implies \mathcal{M} \not\models \phi$$ ✘

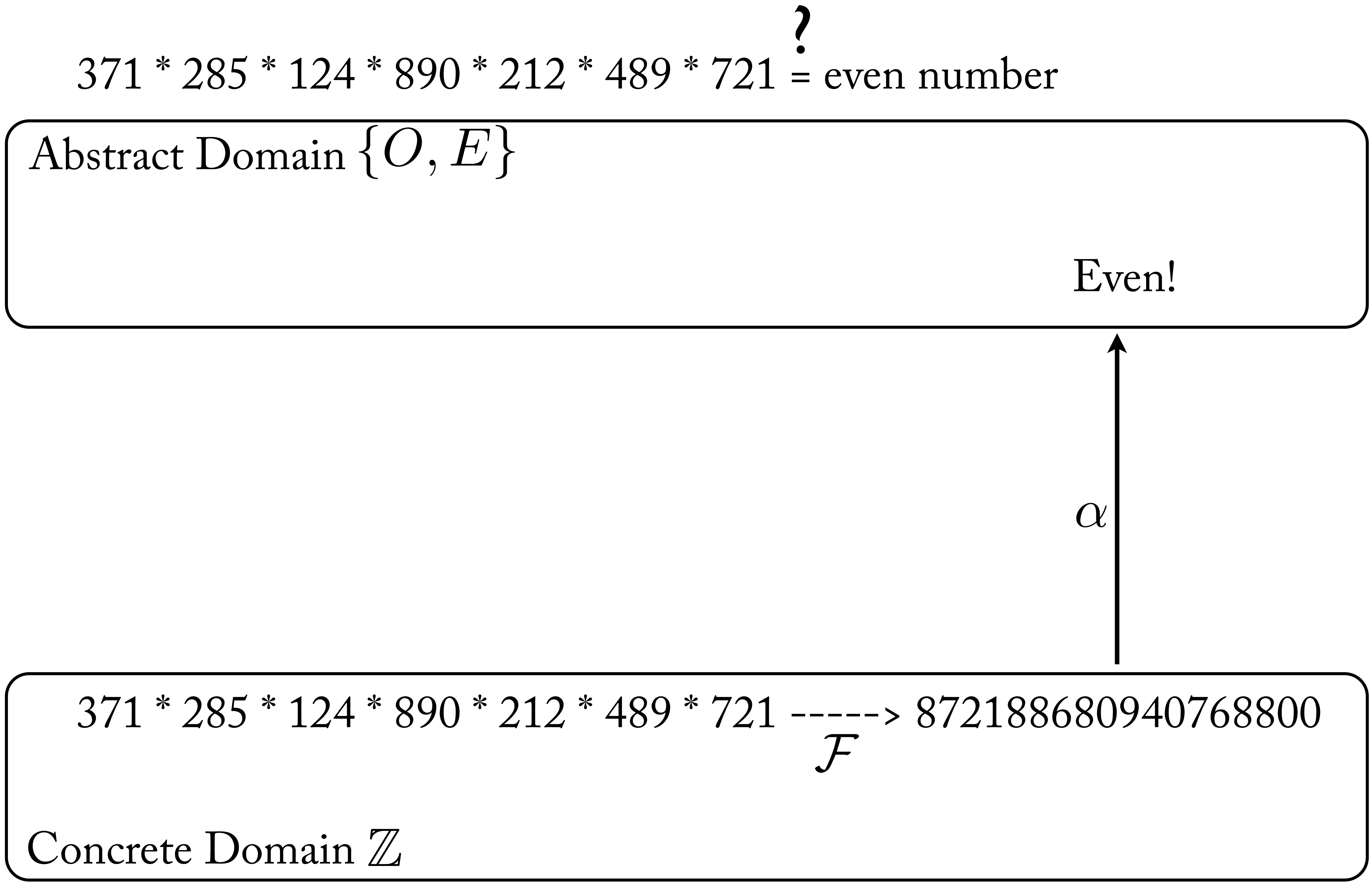
Informal Introduction to
Abstract Interpretation with Examples

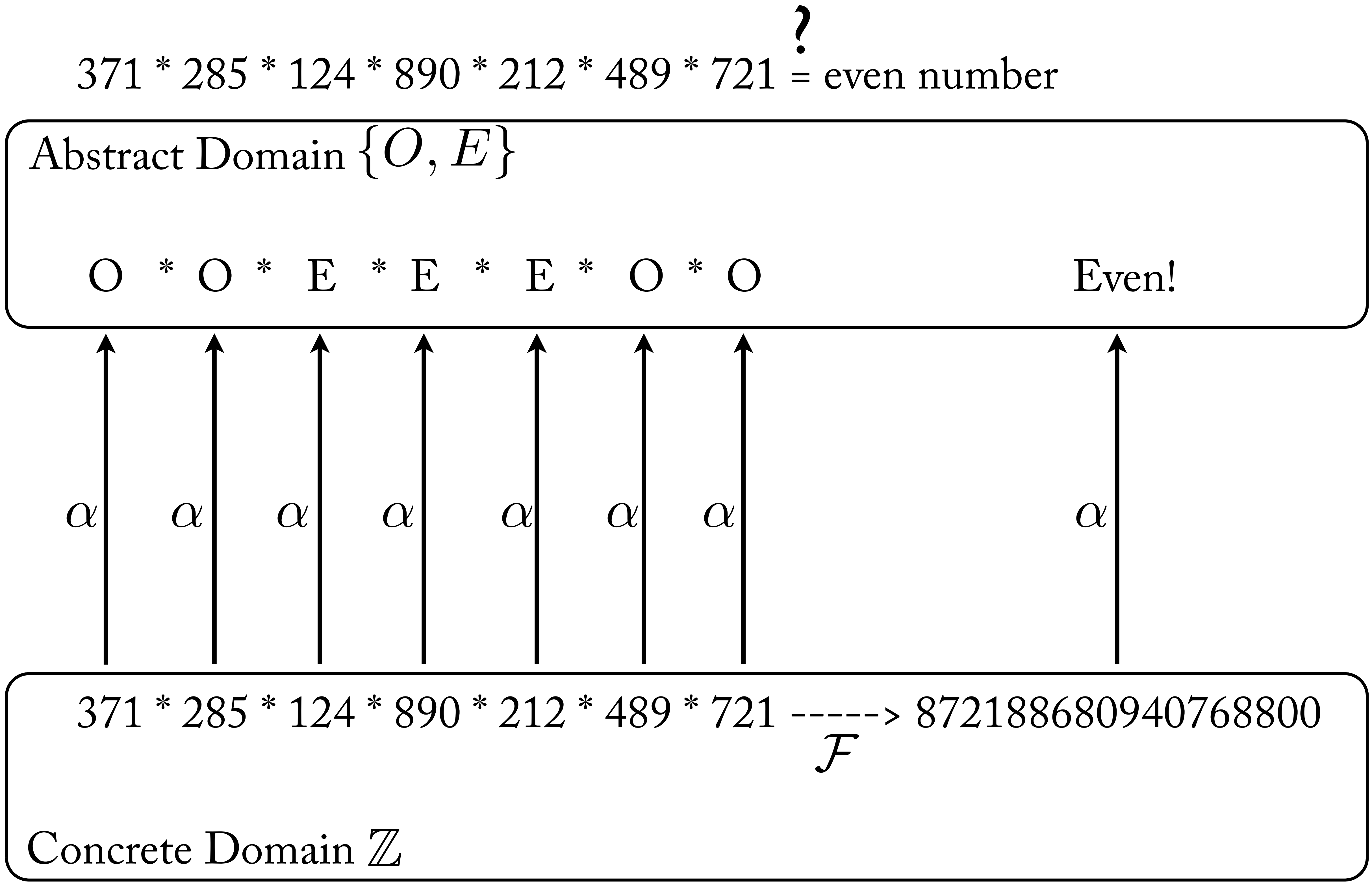
$$371 * 285 * 124 * 890 * 212 * 489 * 721 \stackrel{?}{=} \text{even number}$$

371 * 285 * 124 * 890 * 212 * 489 * 721 $\stackrel{?}{=}$ even number

371 * 285 * 124 * 890 * 212 * 489 * 721 $\xrightarrow[\mathcal{F}]{}$ 87218868094076800

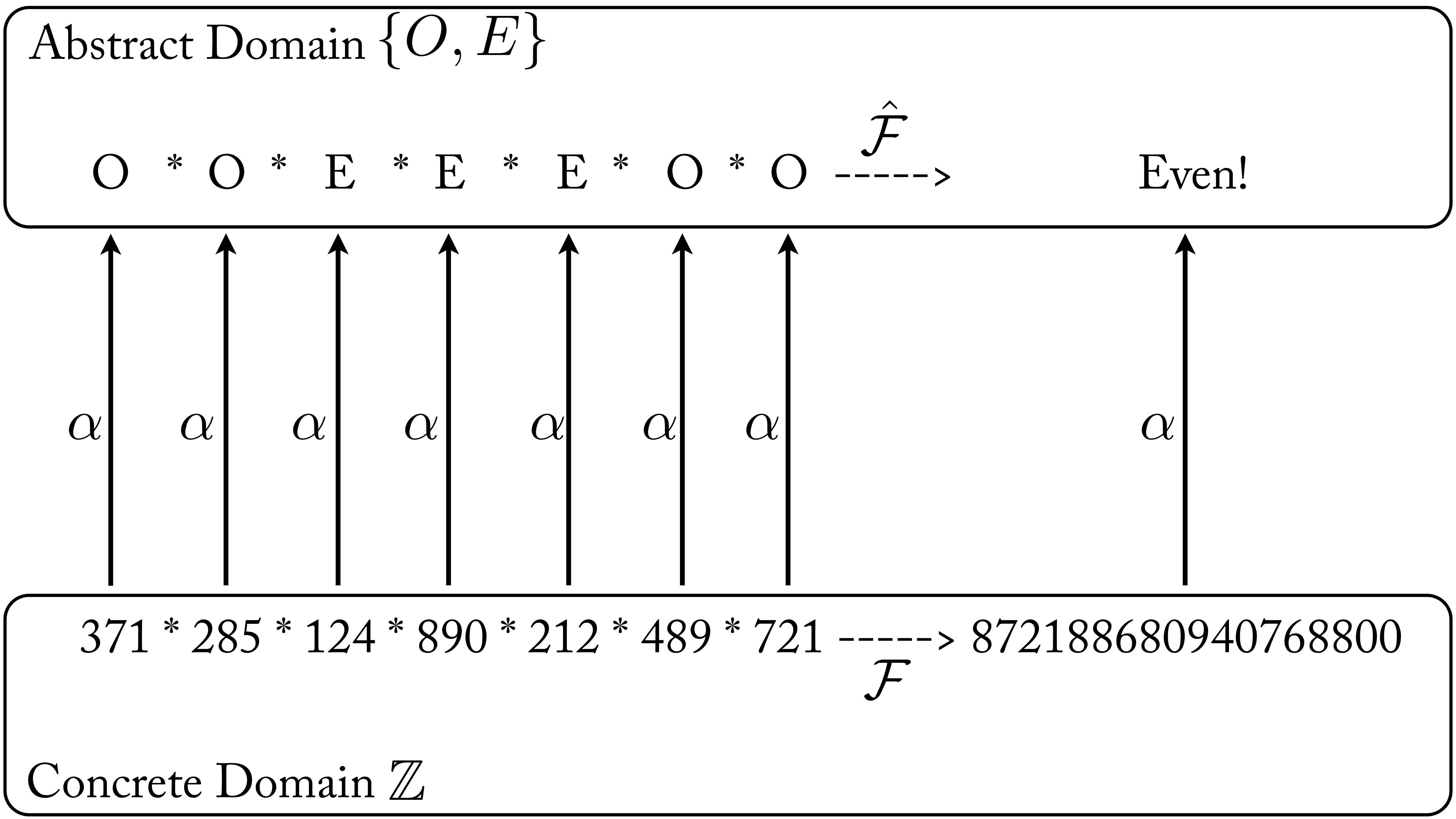
Concrete Domain $\mathbb{Z}$


?
371 * 285 * 124 * 890 * 212 * 489 * 721 = even number
Abstract Domain $\{O, E\}$
Even!
α
371 * 285 * 124 * 890 * 212 * 489 * 721 -----> 8721886809407688000
$\mathcal{F}$
Concrete Domain $\mathbb{Z}$

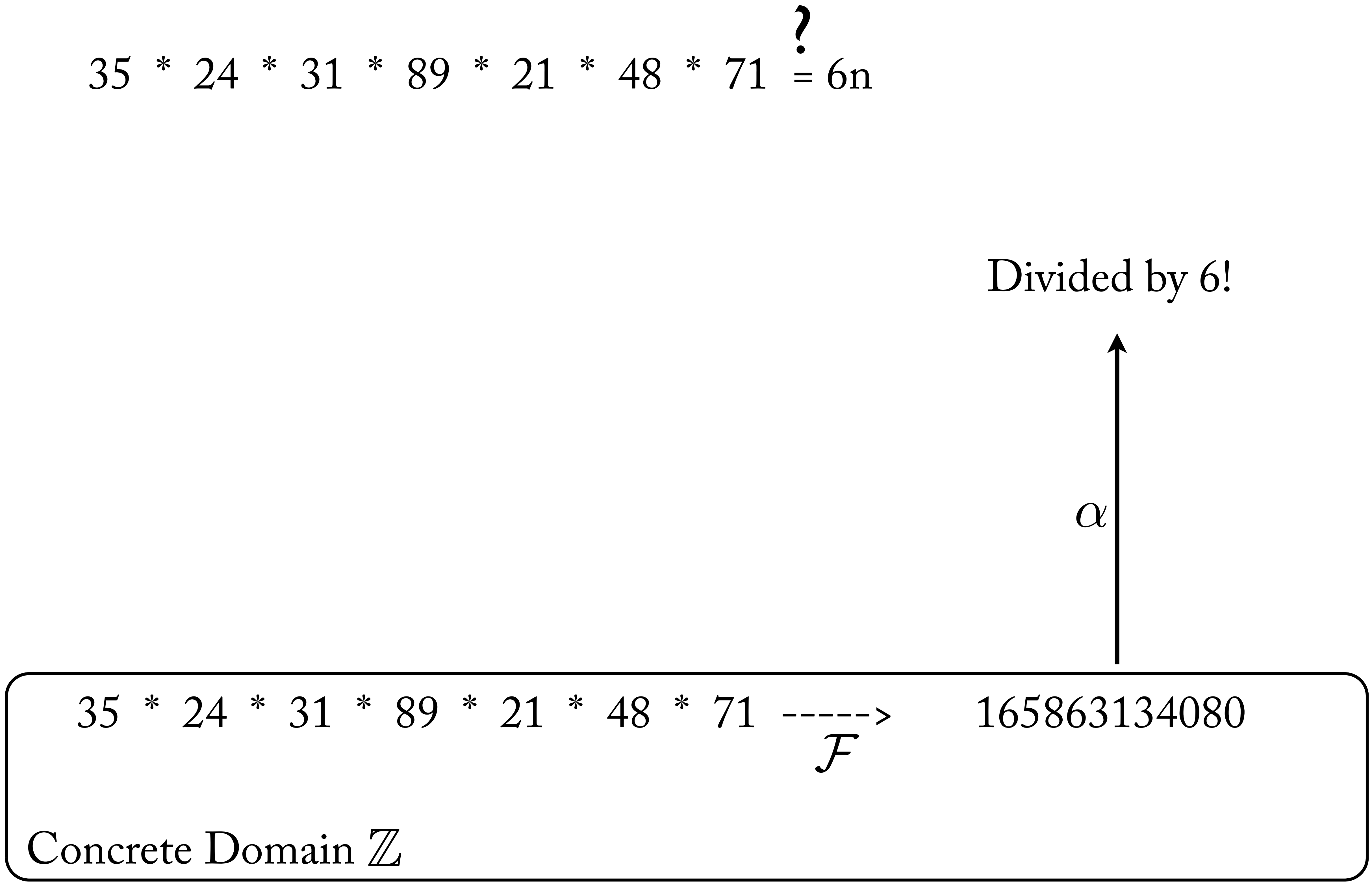


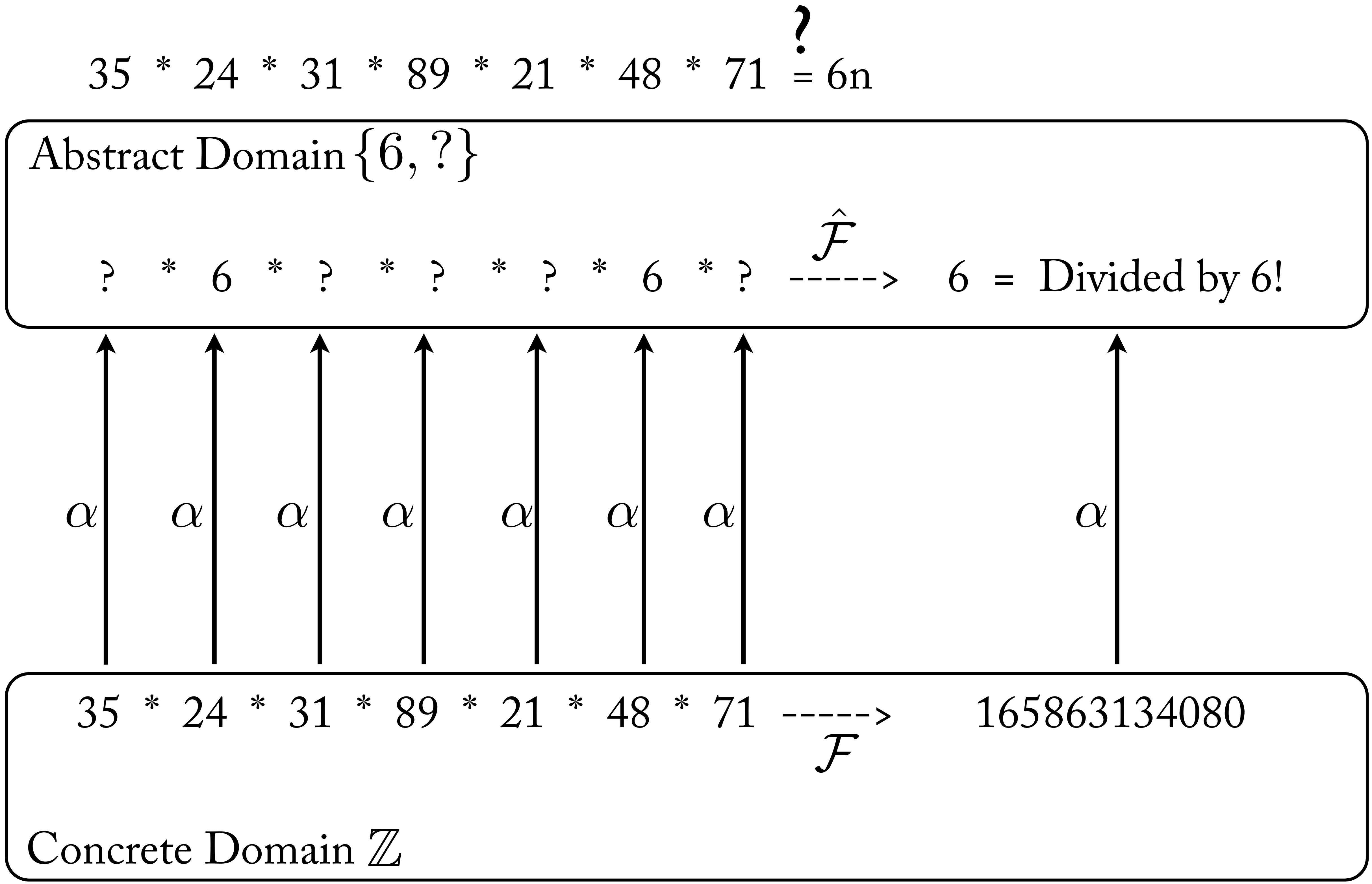
371 * 285 * 124 * 890 * 212 * 489 * 721 ?= even number
Abstract Domain $\{O, E\}$
O * O * E * E * E * O * O
Even!
α
371 * 285 * 124 * 890 * 212 * 489 * 721 -----> 87218868094076800
$\mathcal{F}$
Concrete Domain $\mathbb{Z}$

$$371 * 285 * 124 * 890 * 212 * 489 * 721 \stackrel{?}{=} \text{even number}$$

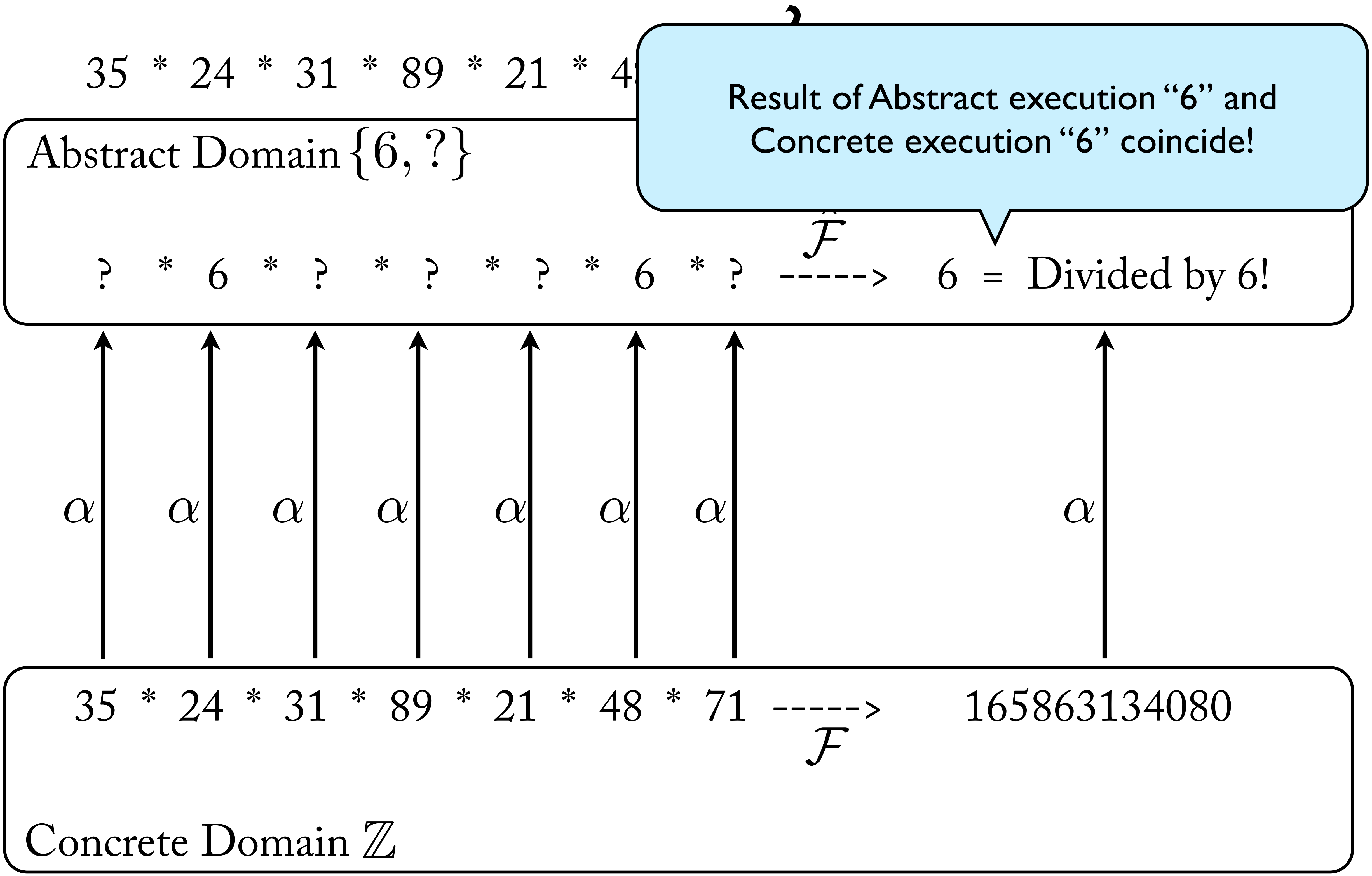


$$35 * 24 * 31 * 89 * 21 * 48 * 71 \overset{?}{=} 6n$$


35 * 24 * 31 * 89 * 21 * 48 * 71 ?= 6n
Divided by 6!
α
35 * 24 * 31 * 89 * 21 * 48 * 71 -----> 165863134080
F
Concrete Domain ℤ



35 * 24 * 31 * 89 * 21 * 48 * 71 ?= 6n
Abstract Domain {6, ?}
? * 6 * ? * ? * ? * 6 * ? ------> 6 = Divided by 6!
$\hat{\mathcal{F}}$
α
35 * 24 * 31 * 89 * 21 * 48 * 71 ------> 165863134080
$\mathcal{F}$
Concrete Domain $\mathbb{Z}$


35 * 24 * 31 * 89 * 21 * 4
Result of Abstract execution "6" and Concrete execution "6" coincide!
Abstract Domain {6, ?}
? * 6 * ? * ? * ? * 6 * ? -----> 6 = Divided by 6!
α
35 * 24 * 31 * 89 * 21 * 48 * 71 -----> 165863134080
Concrete Domain ℤ

$$371 * 285 * 124 * 890 * 212 * 489 * 721 \stackrel{?}{=} 6n$$

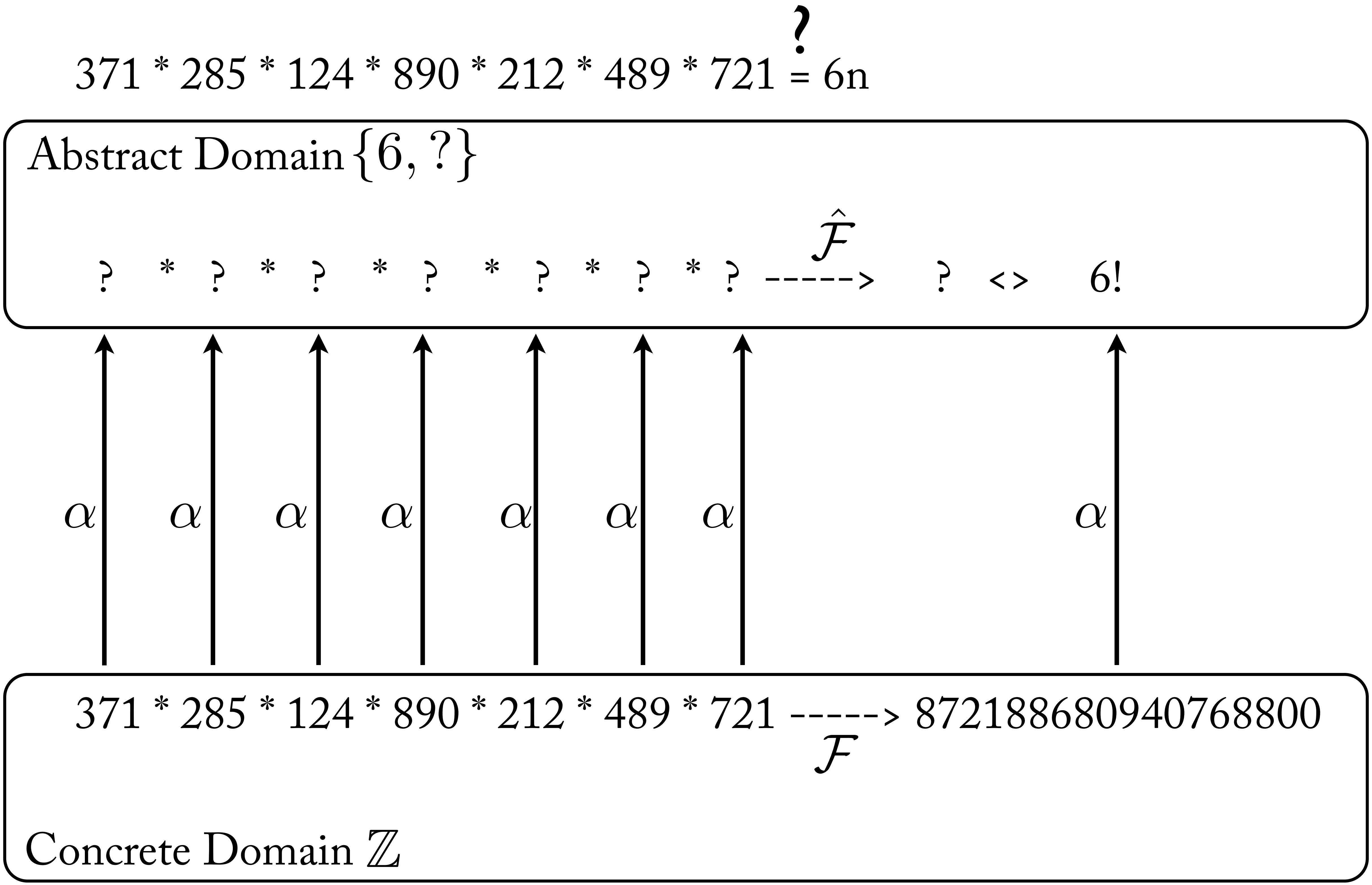
$371 * 285 * 124 * 890 * 212 * 489 * 721 \stackrel{?}{=} 6n$

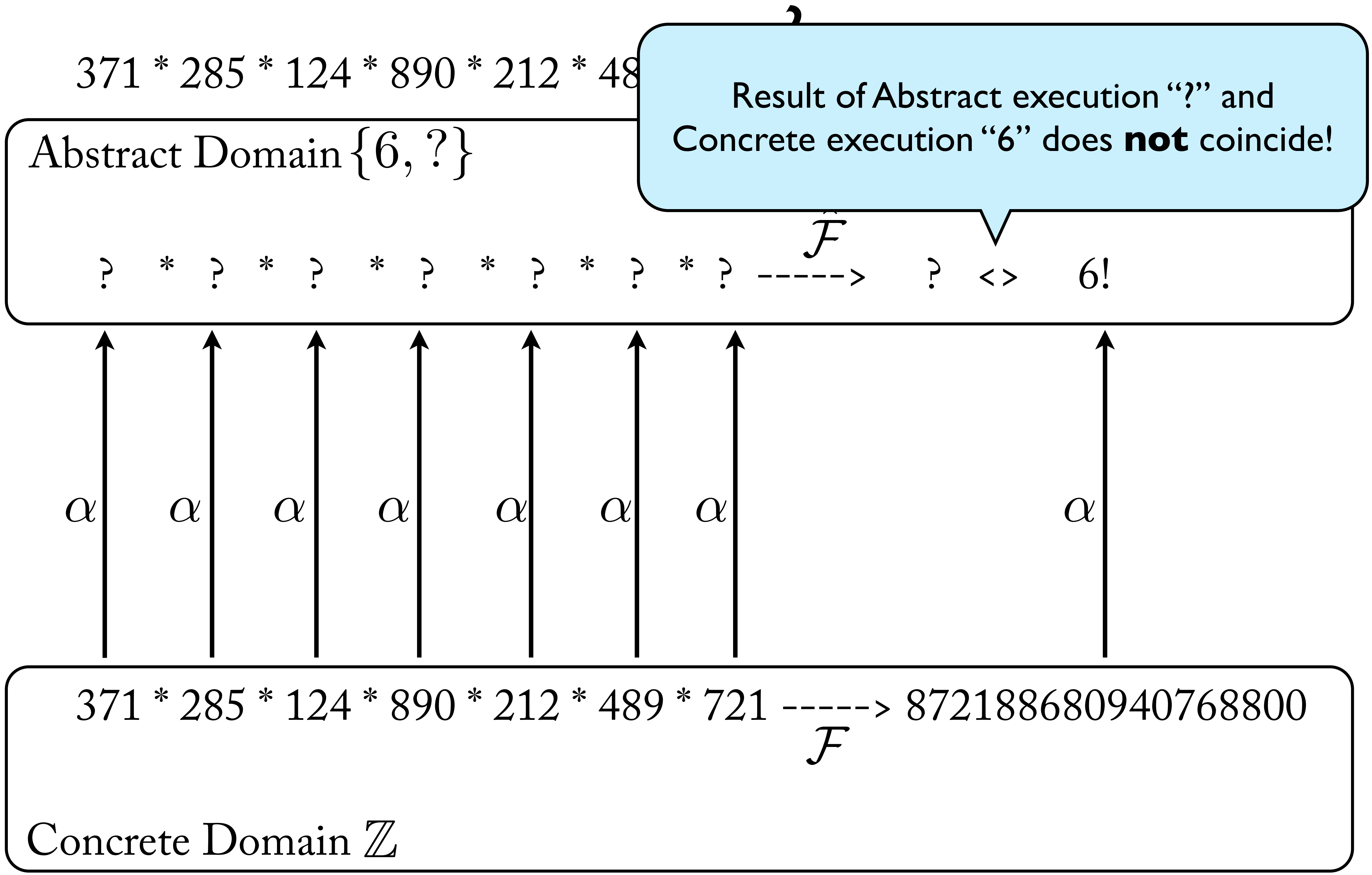
6!

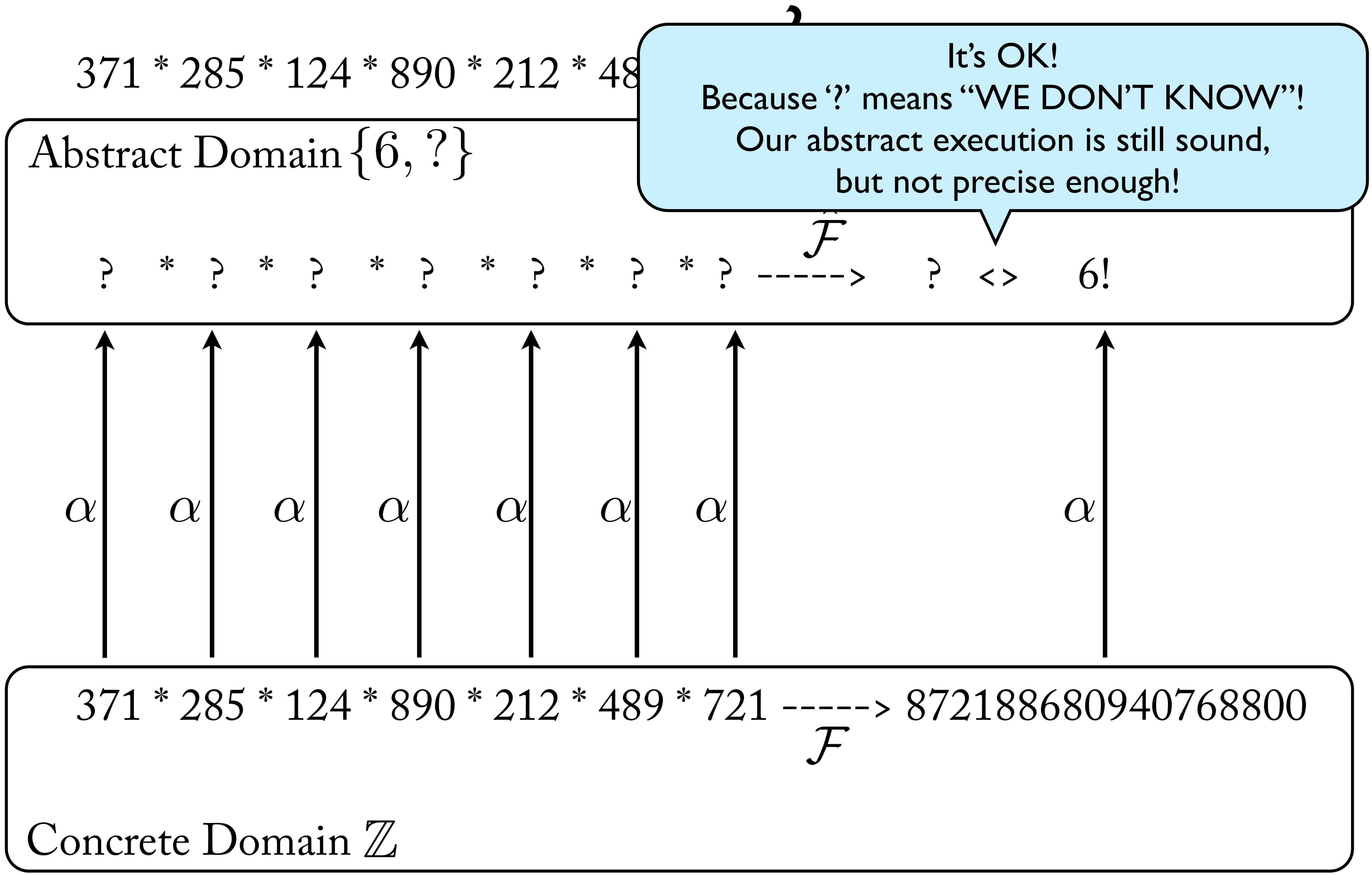
α

$371 * 285 * 124 * 890 * 212 * 489 * 721 \xrightarrow[\mathcal{F}]{} 872188680940768800$

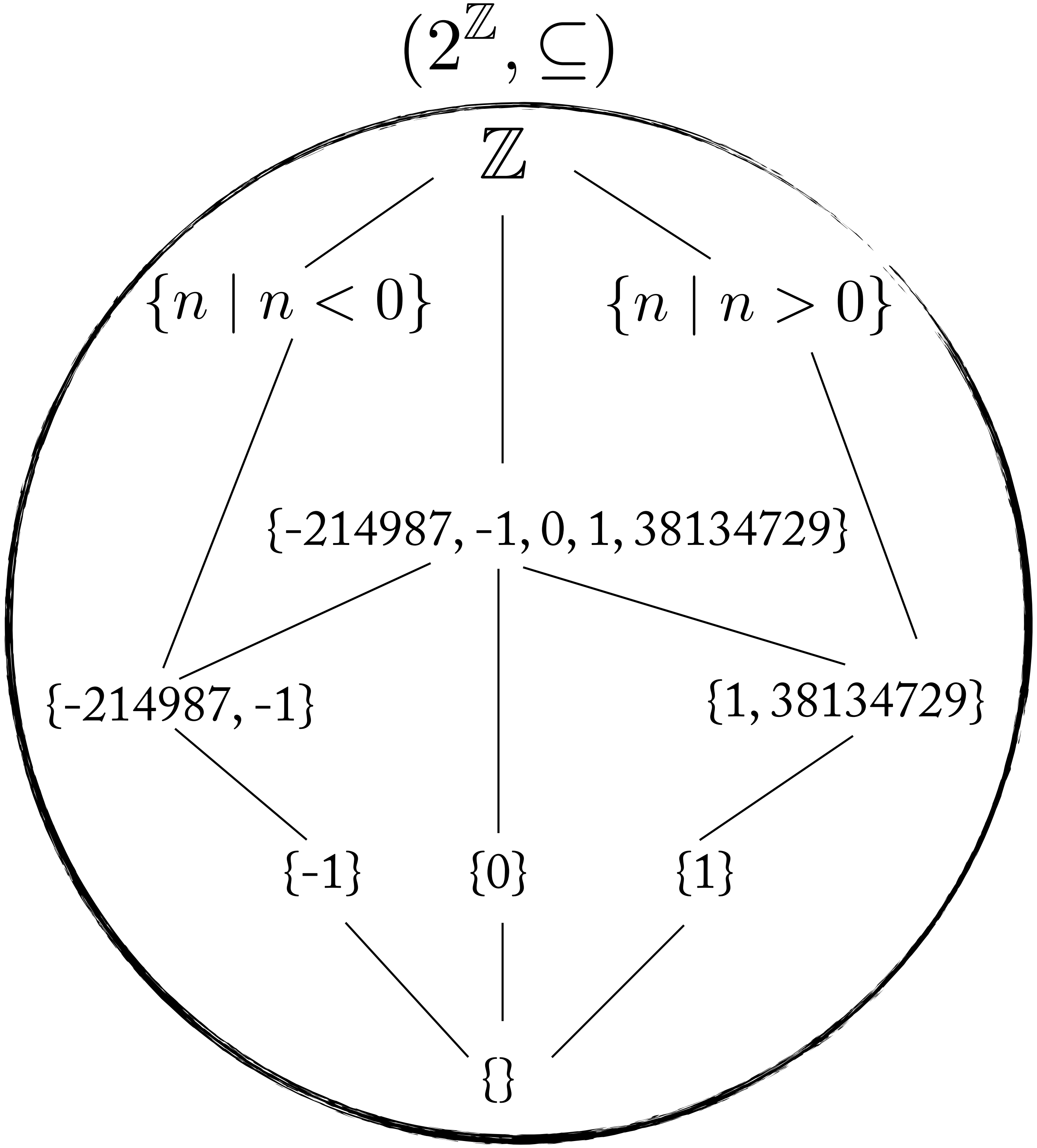
Concrete Domain $\mathbb{Z}$


?
371 * 285 * 124 * 890 * 212 * 489 * 721 = 6n
Abstract Domain {6, ?}
F̂
? * ? * ? * ? * ? * ? * ? ------> ? <> 6!
α
α
α
α
α
α
α
α
371 * 285 * 124 * 890 * 212 * 489 * 721 -----> 87218868094076880
F
Concrete Domain ℤ

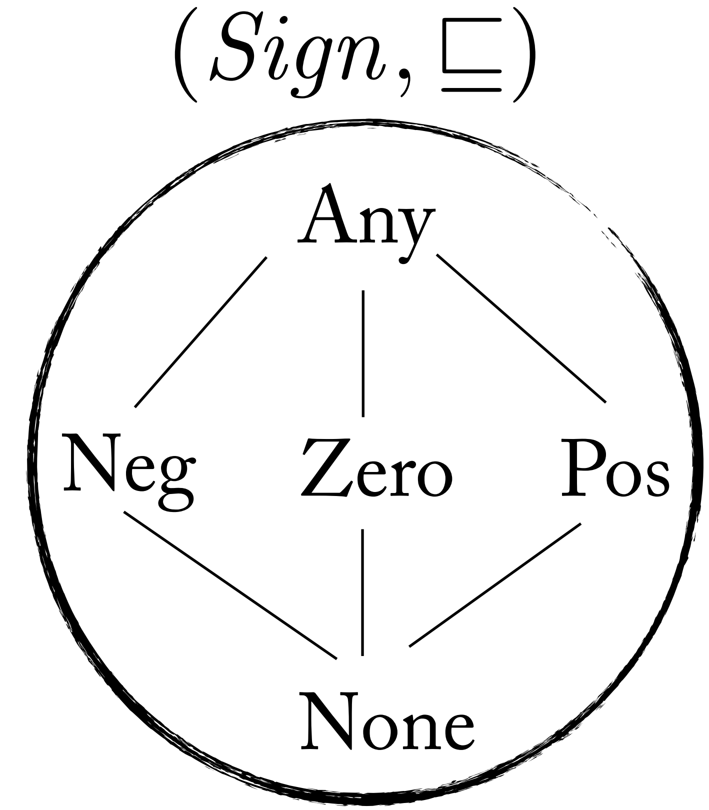

371 * 285 * 124 * 890 * 212 * 48
Result of Abstract execution “?” and
Concrete execution “6” does **not** coincide!
Abstract Domain {6, ?}
$\hat{\mathcal{F}}$
? * ? * ? * ? * ? * ? * ? -----> ? <> 6!
α
371 * 285 * 124 * 890 * 212 * 489 * 721 -----> 872188680940768800
$\mathcal{F}$
Concrete Domain $\mathbb{Z}$


371 * 285 * 124 * 890 * 212 * 48
It's OK!
Because '?' means "WE DON'T KNOW"!
Our abstract execution is still sound,
but not precise enough!
Abstract Domain {6, ?}
$\mathcal{F}$
? * ? * ? * ? * ? * ? * ? -----> ? <> 6!
α
371 * 285 * 124 * 890 * 212 * 489 * 721 -----> 872188680940768800
$\mathcal{F}$
Concrete Domain $\mathbb{Z}$

Formal Introduction to
Abstract Interpretation with Examples

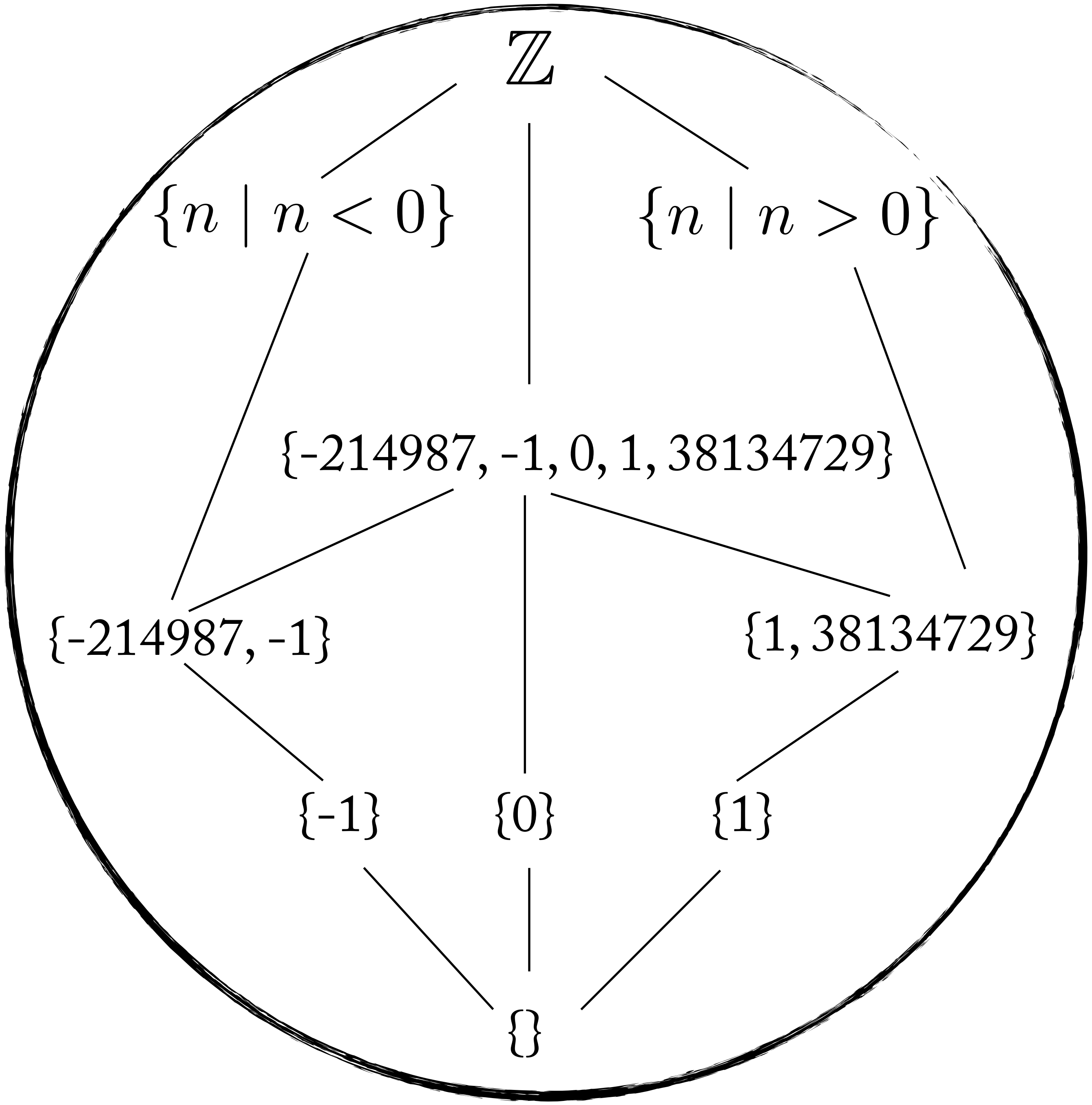


Concrete Domain : D



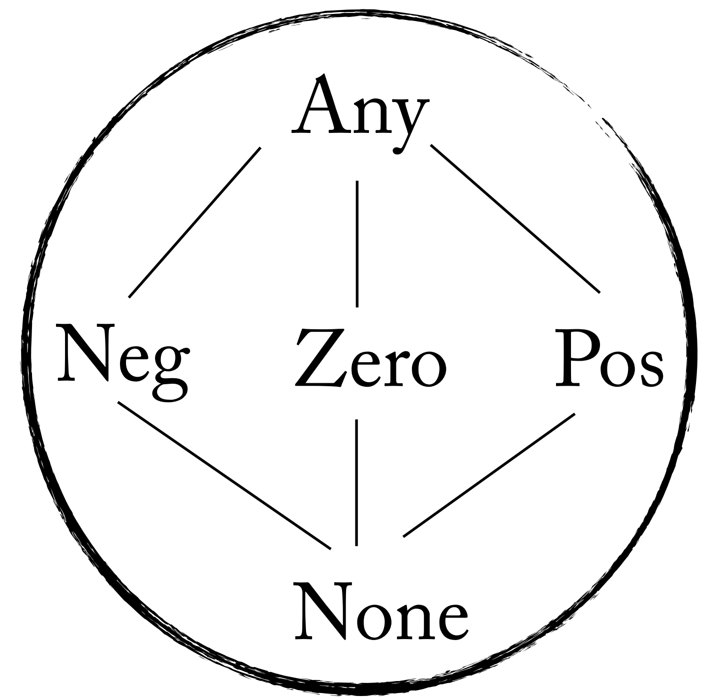
Abstract Domain : $\hat{D}$

$(2^{\mathbb{Z}}, \subseteq)$



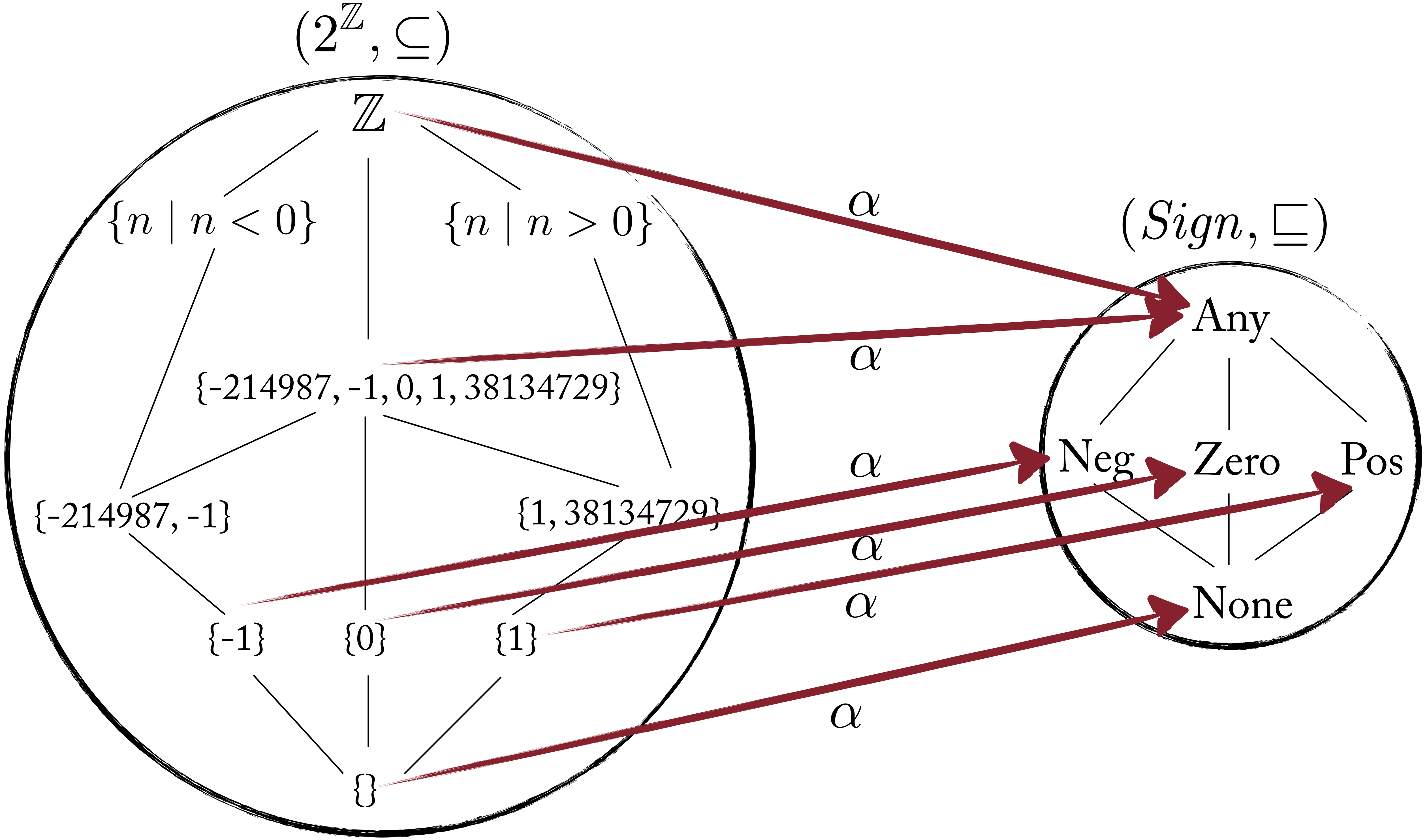
Concrete Domain : D

$(Sign, \sqsubseteq)$



Abstract Domain : $\hat{D}$

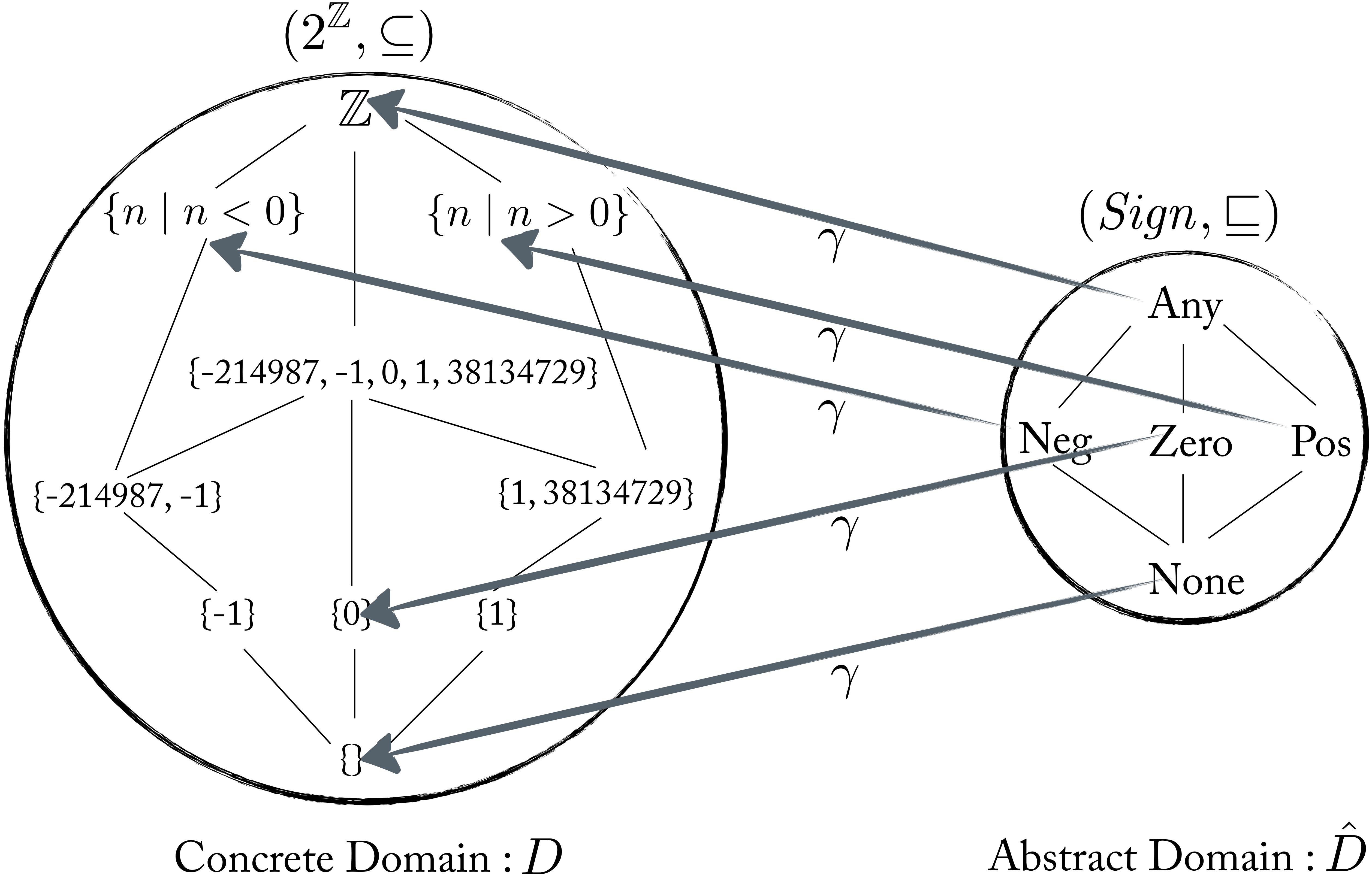
Need to have abstraction function(α) and concretization function (γ) to give a meaning to an Abstract Domain $\hat{D}$



Concrete Domain : D

Abstract Domain : $\hat{D}$

Need to have abstraction function(α) and concretization function (γ) to give a meaning to an Abstract Domain $\hat{D}$

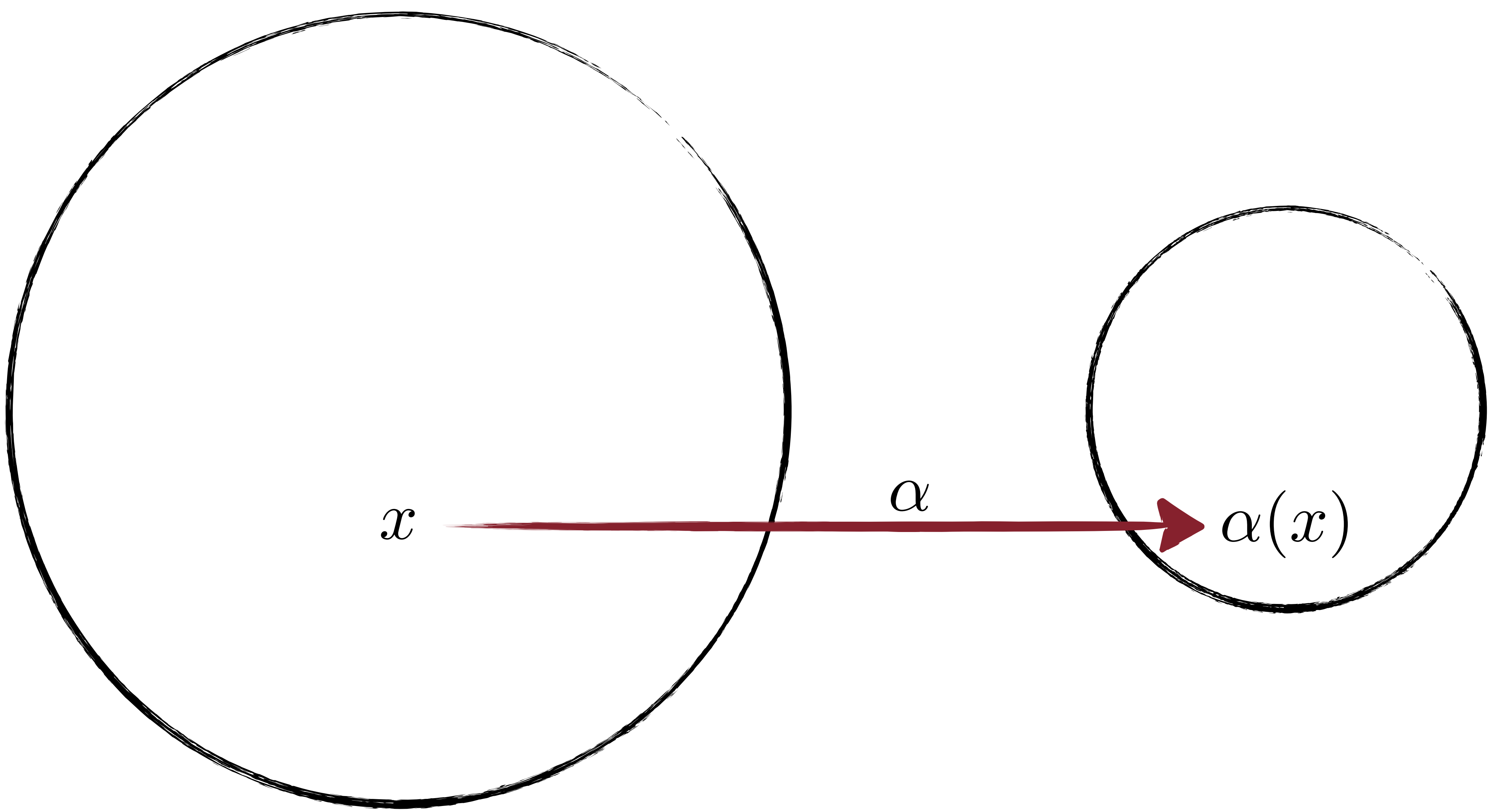


Need to have abstraction function(α) and concretization function (γ) to give a meaning to an Abstract Domain $\hat{D}$

What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



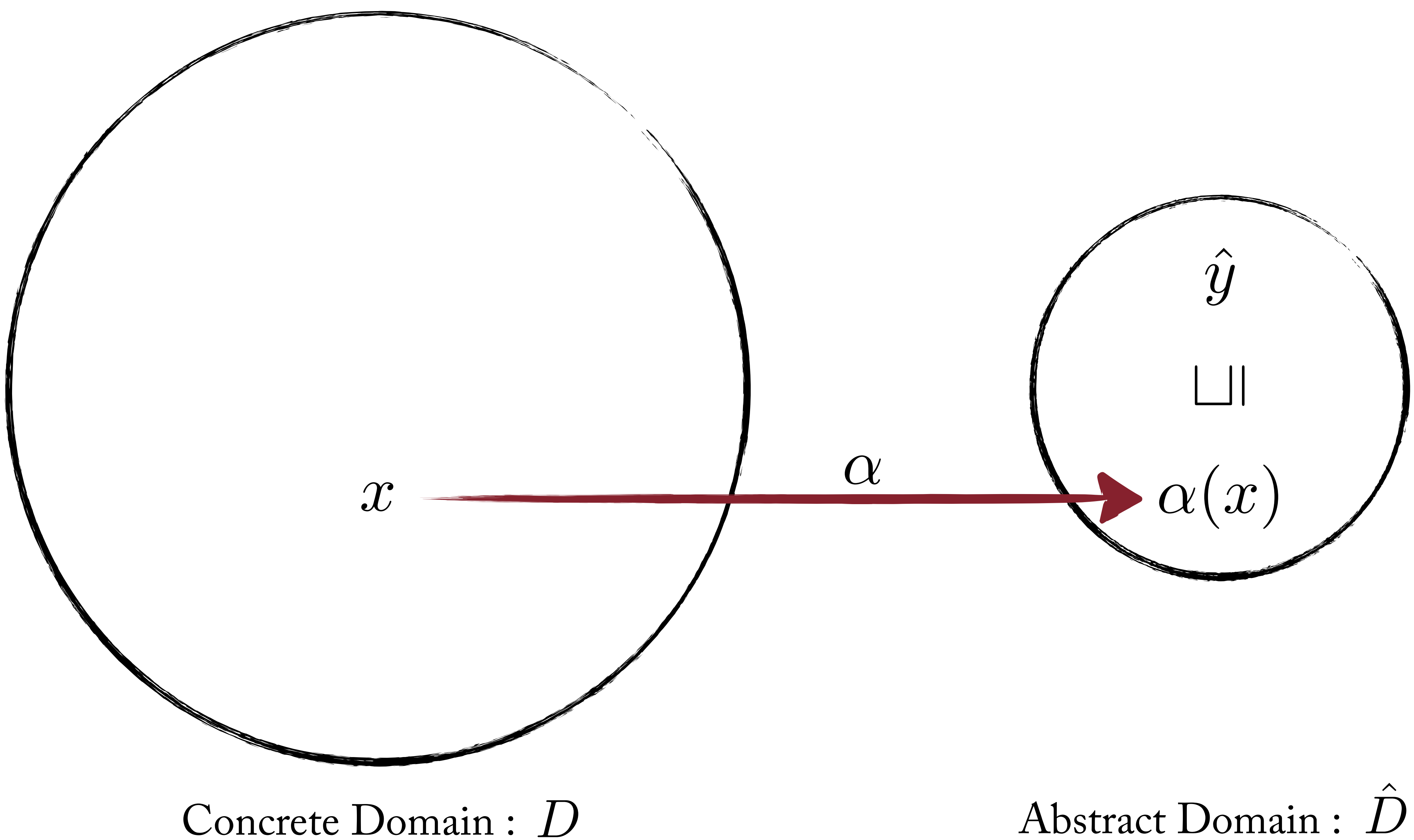
Concrete Domain : D

Abstract Domain : $\hat{D}$

What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

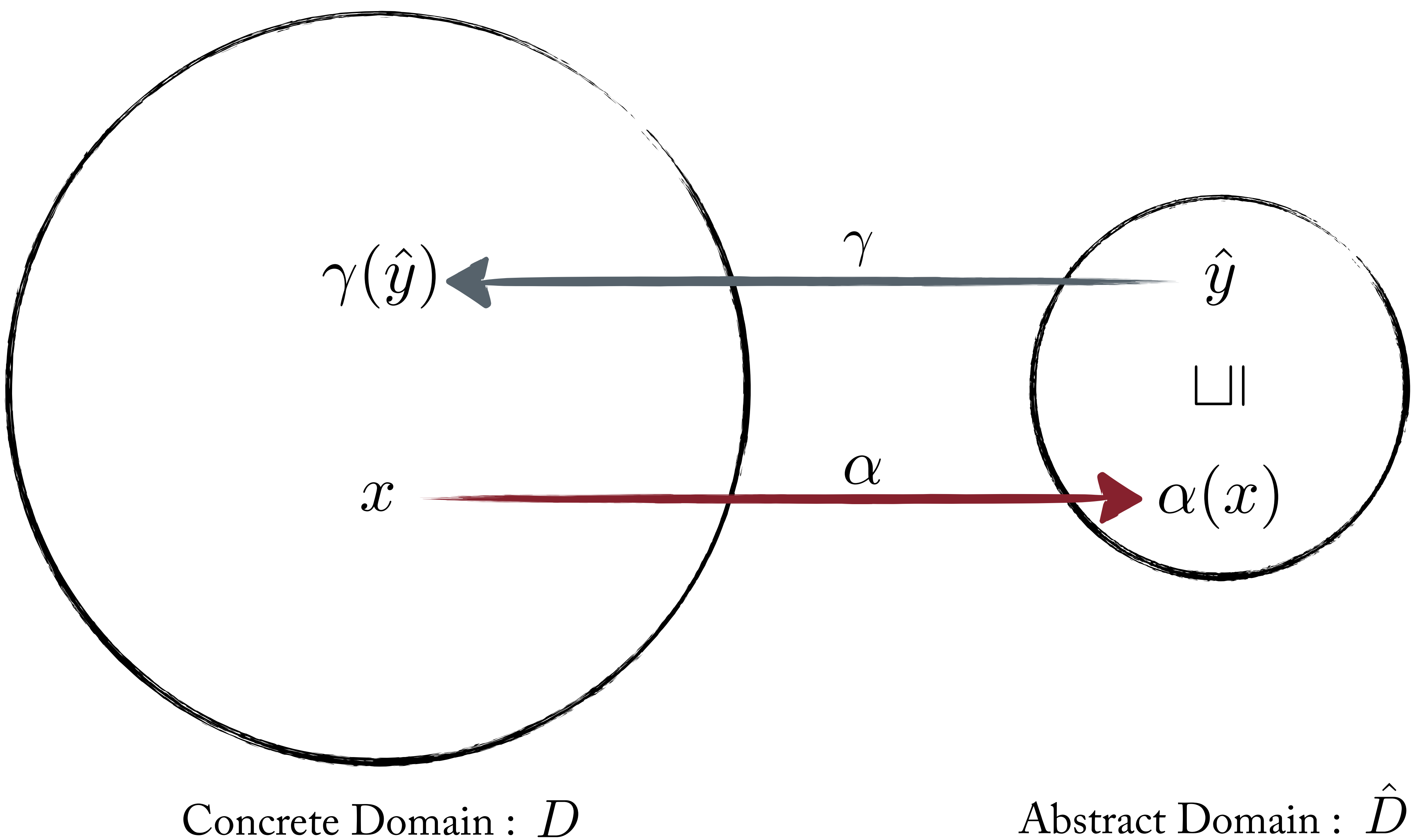
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

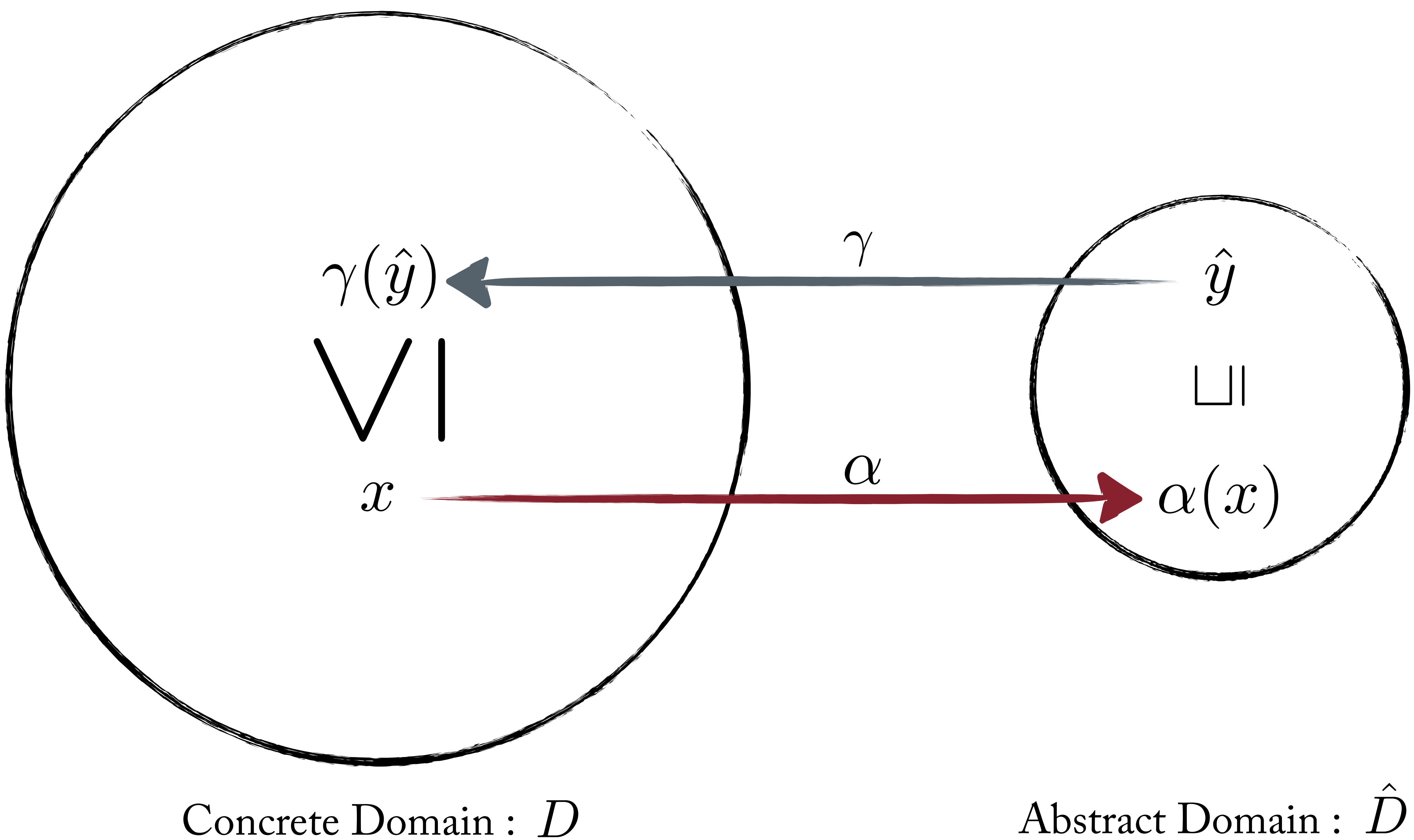
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

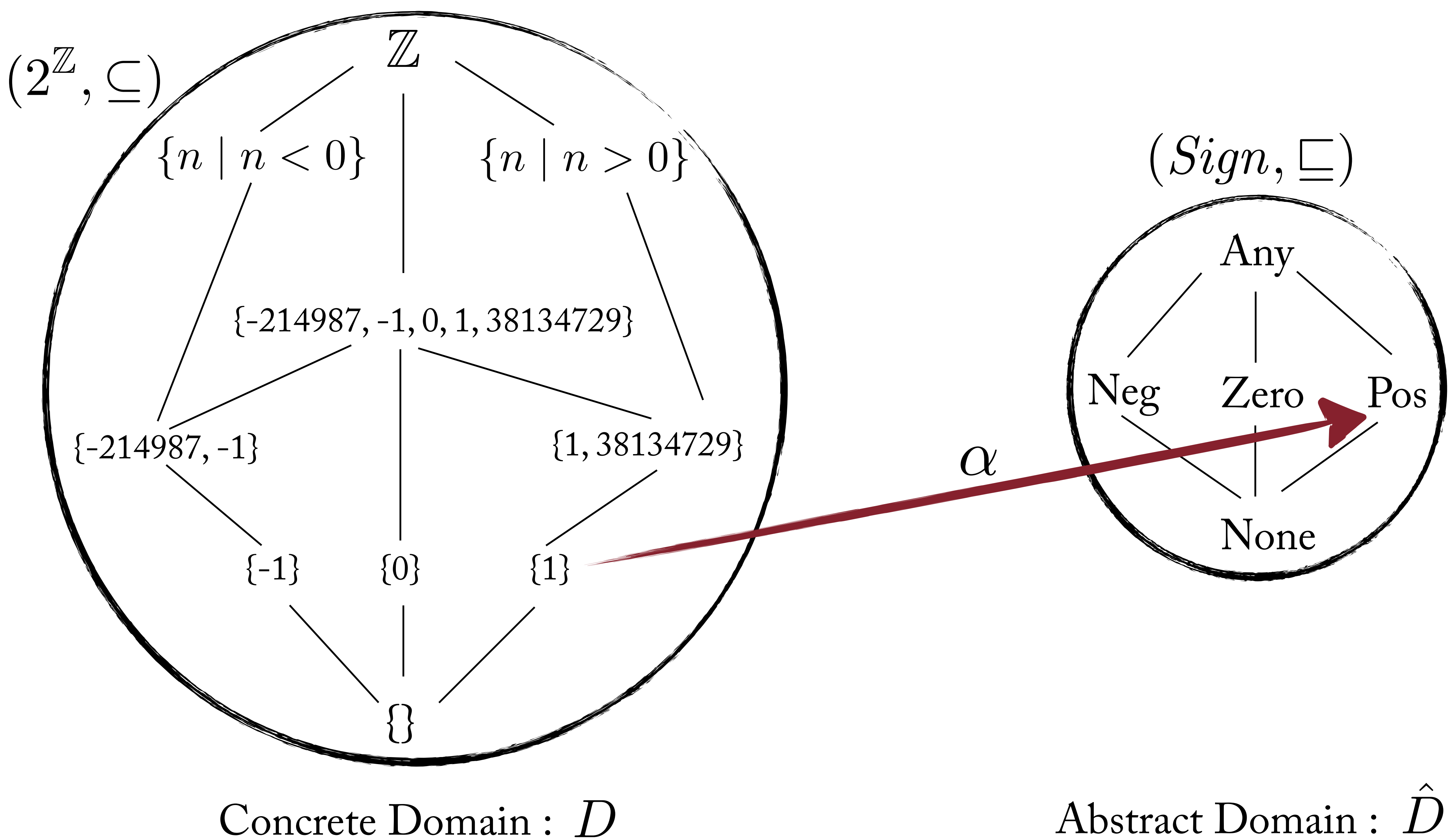
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

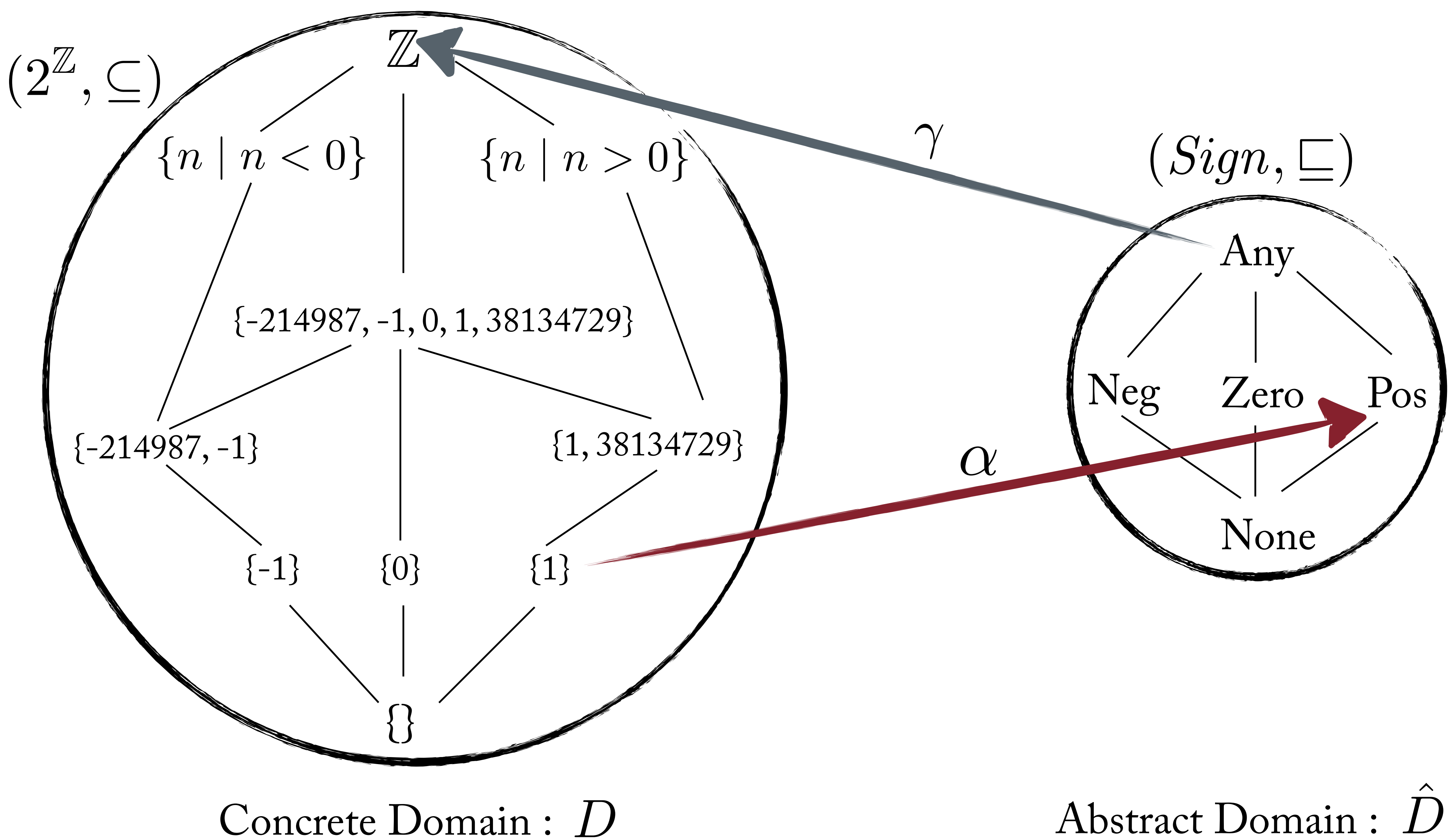
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$

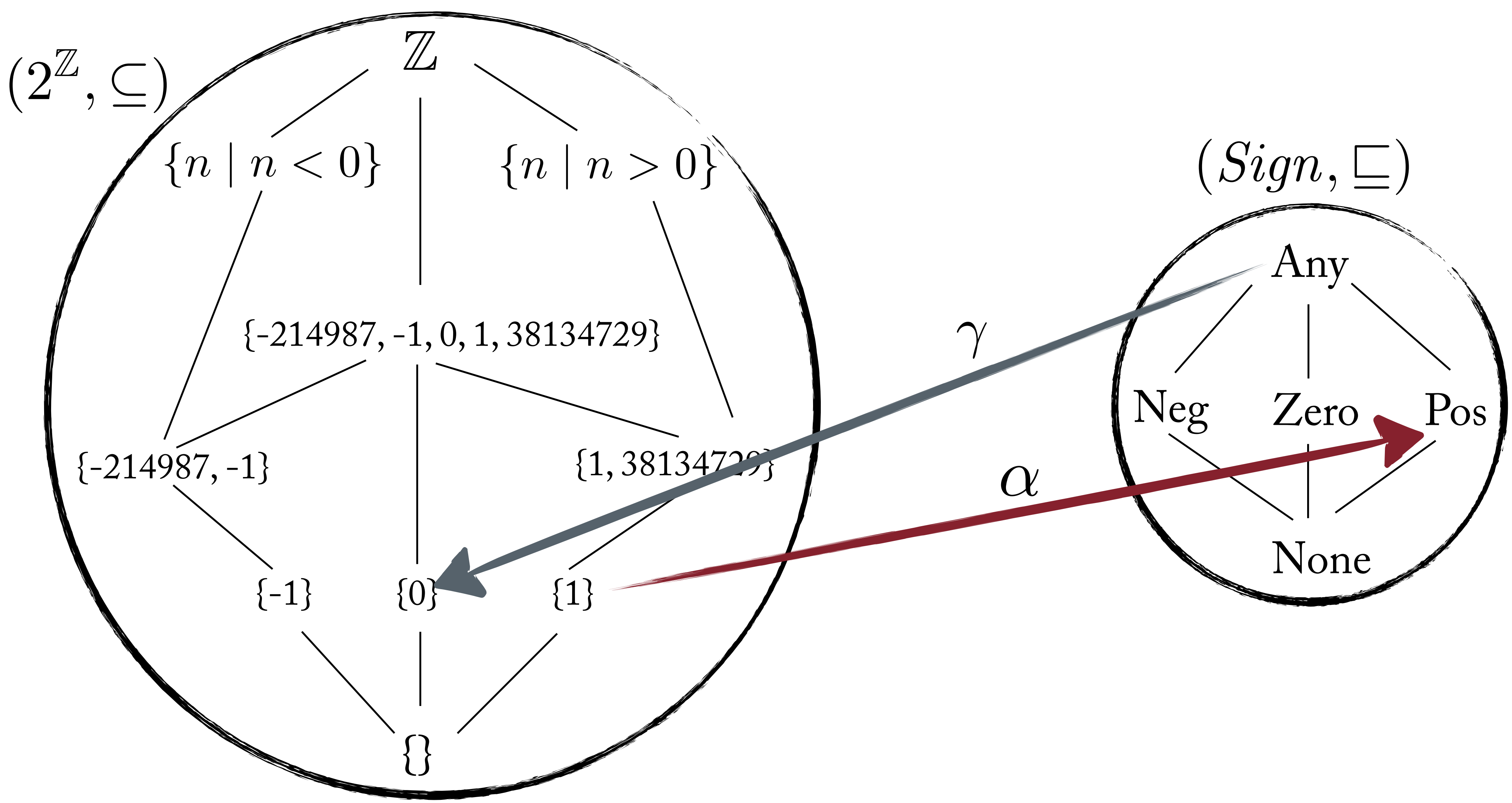


What is the general condition that α and γ should satisfy?

~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



Concrete Domain : D

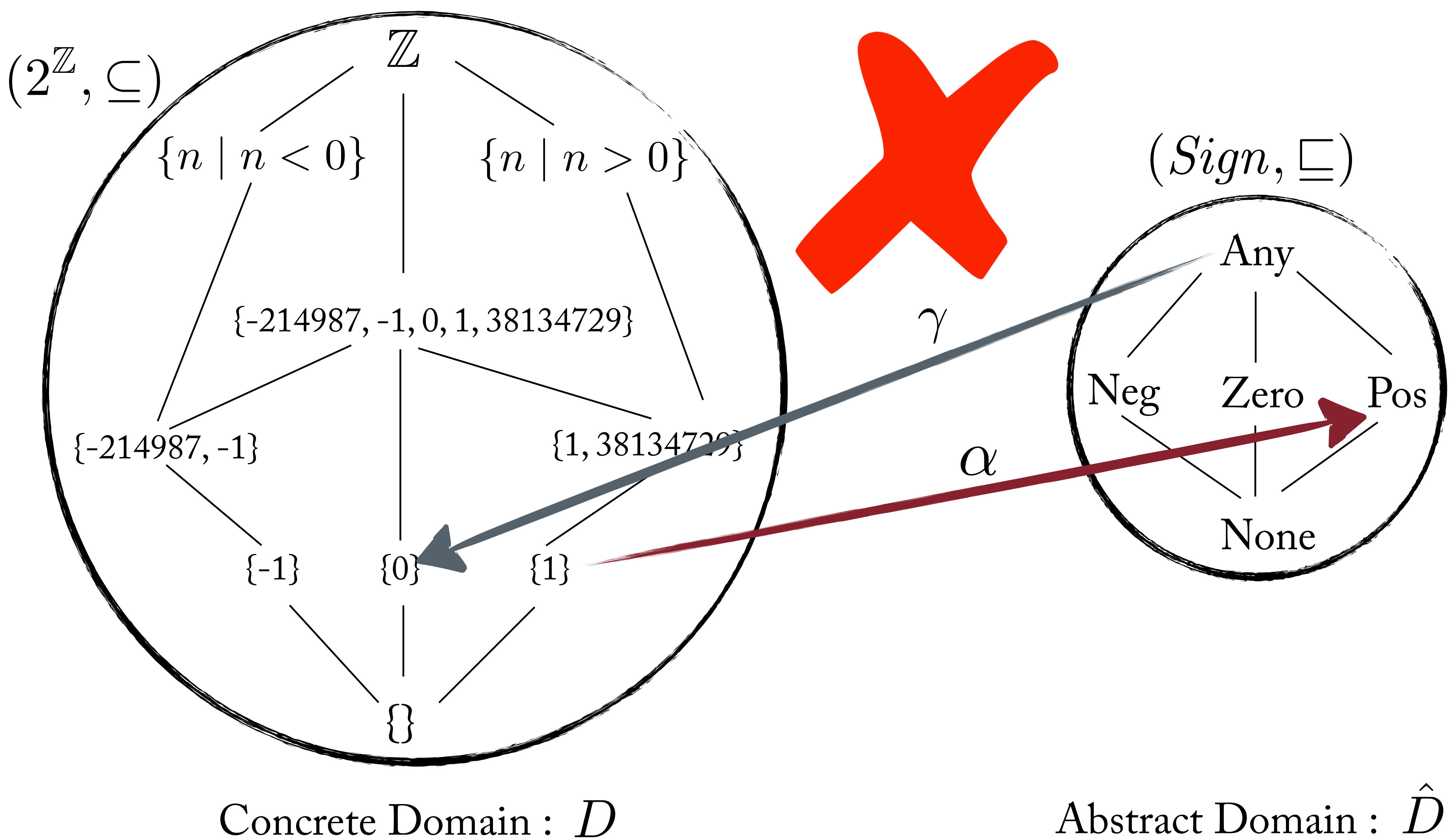
Abstract Domain : $\hat{D}$

What is the general condition that α and γ should satisfy?

~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

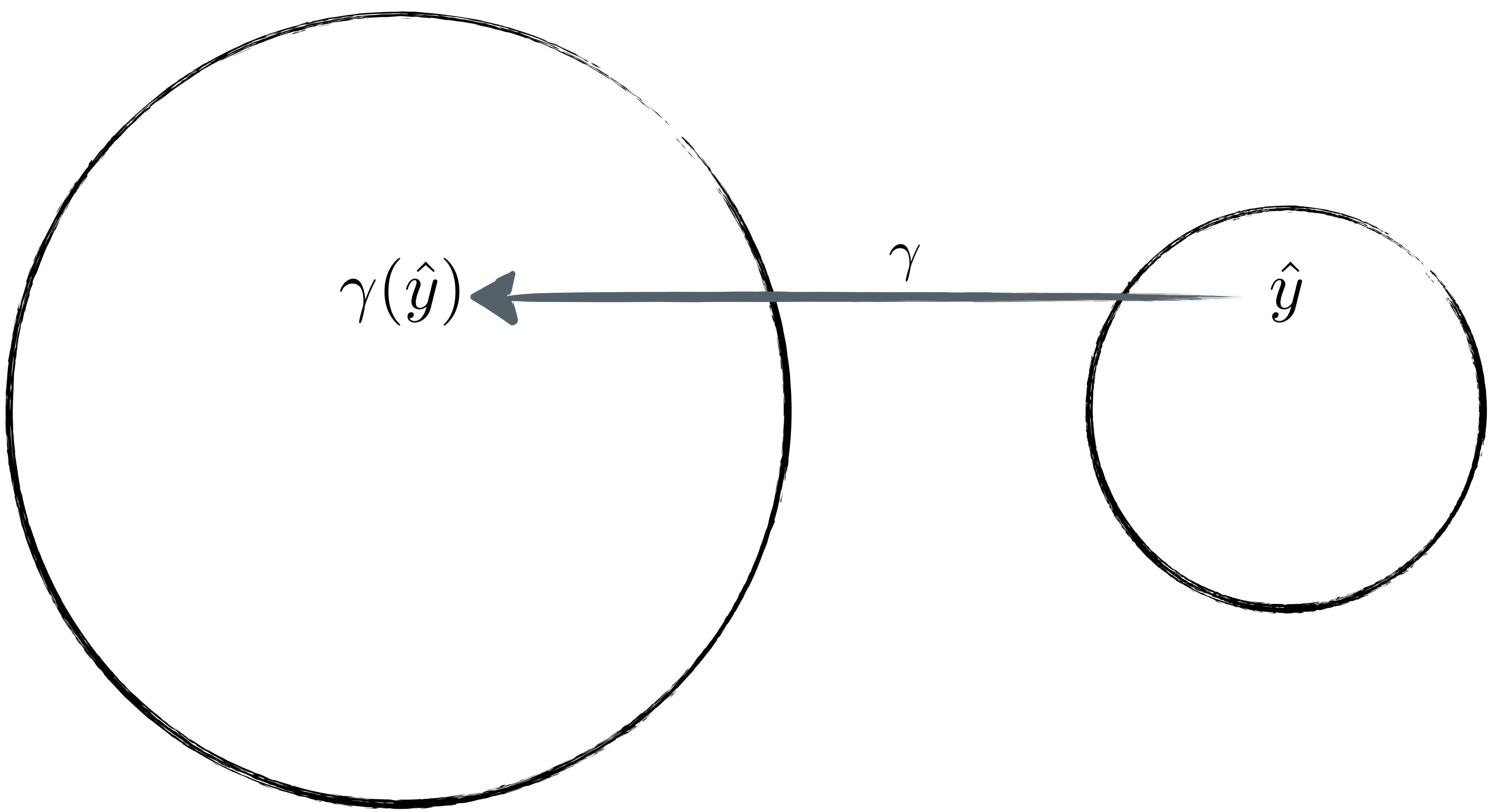
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$

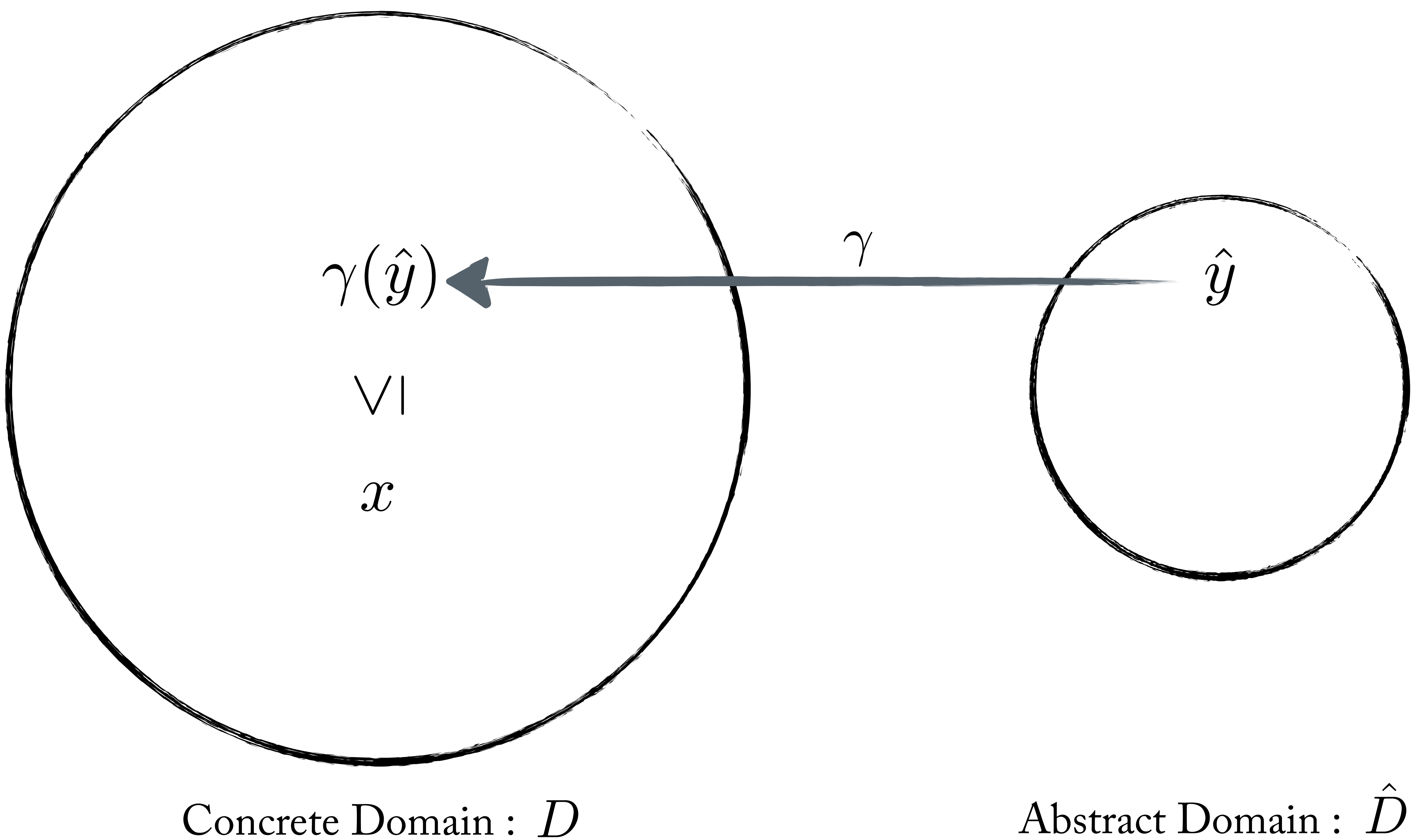


Concrete Domain : D Abstract Domain : $\hat{D}$

What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

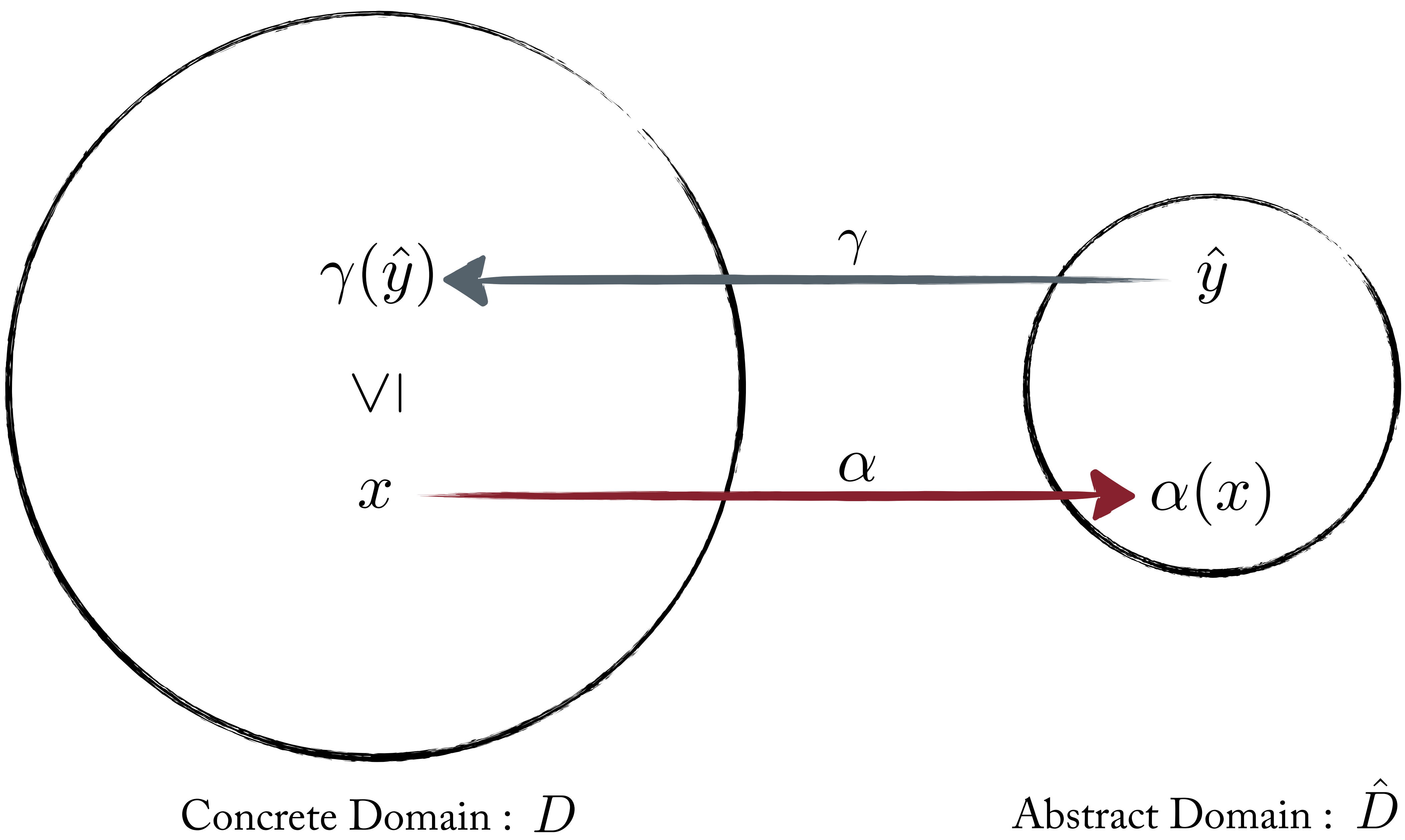
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

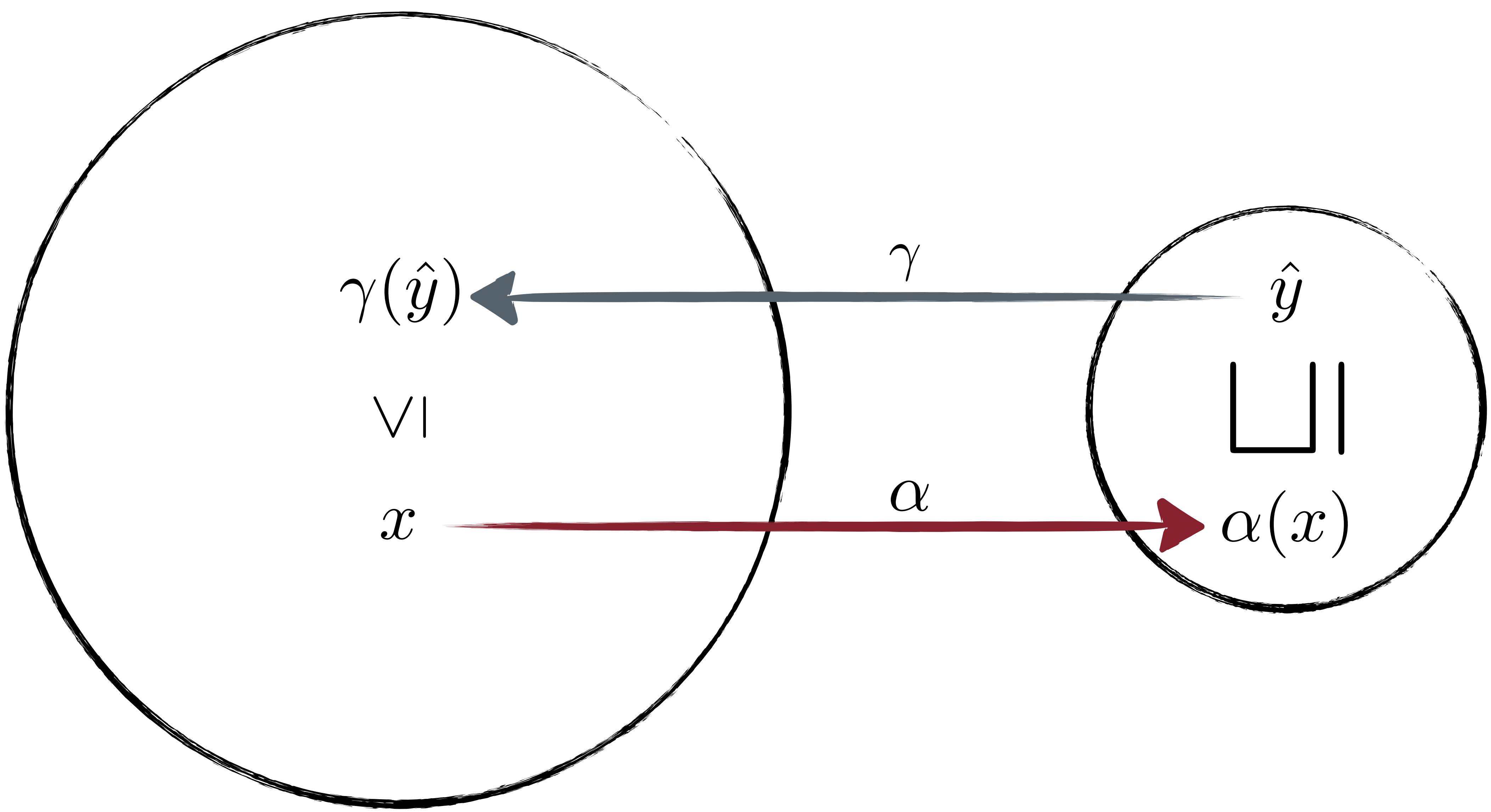
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\rightleftarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



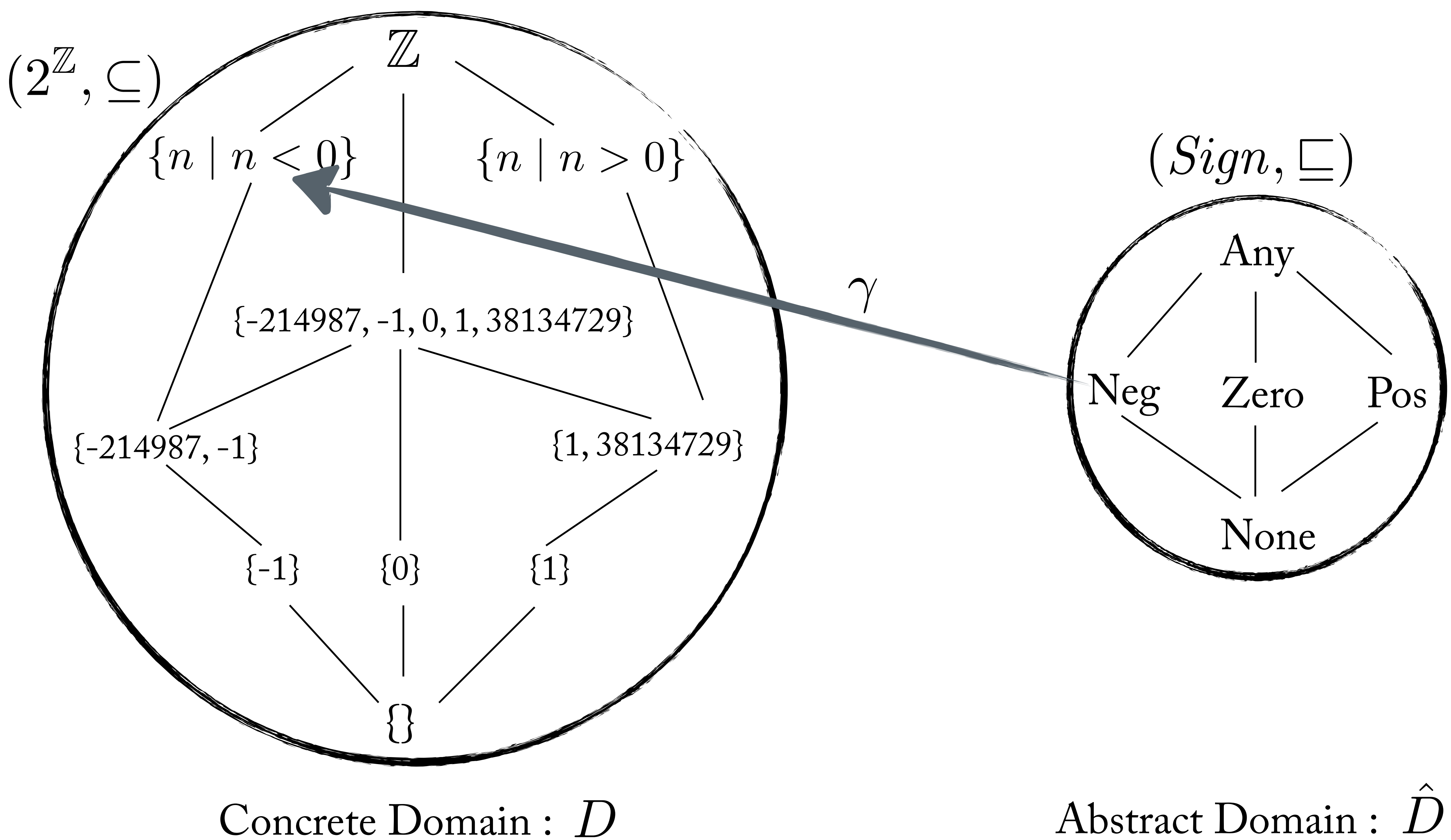
Concrete Domain : D

Abstract Domain : $\hat{D}$

What is the general condition that α and γ should satisfy?
~ Galois Connection.

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\rightleftarrows}} (\hat{D}, \sqsubseteq)$$

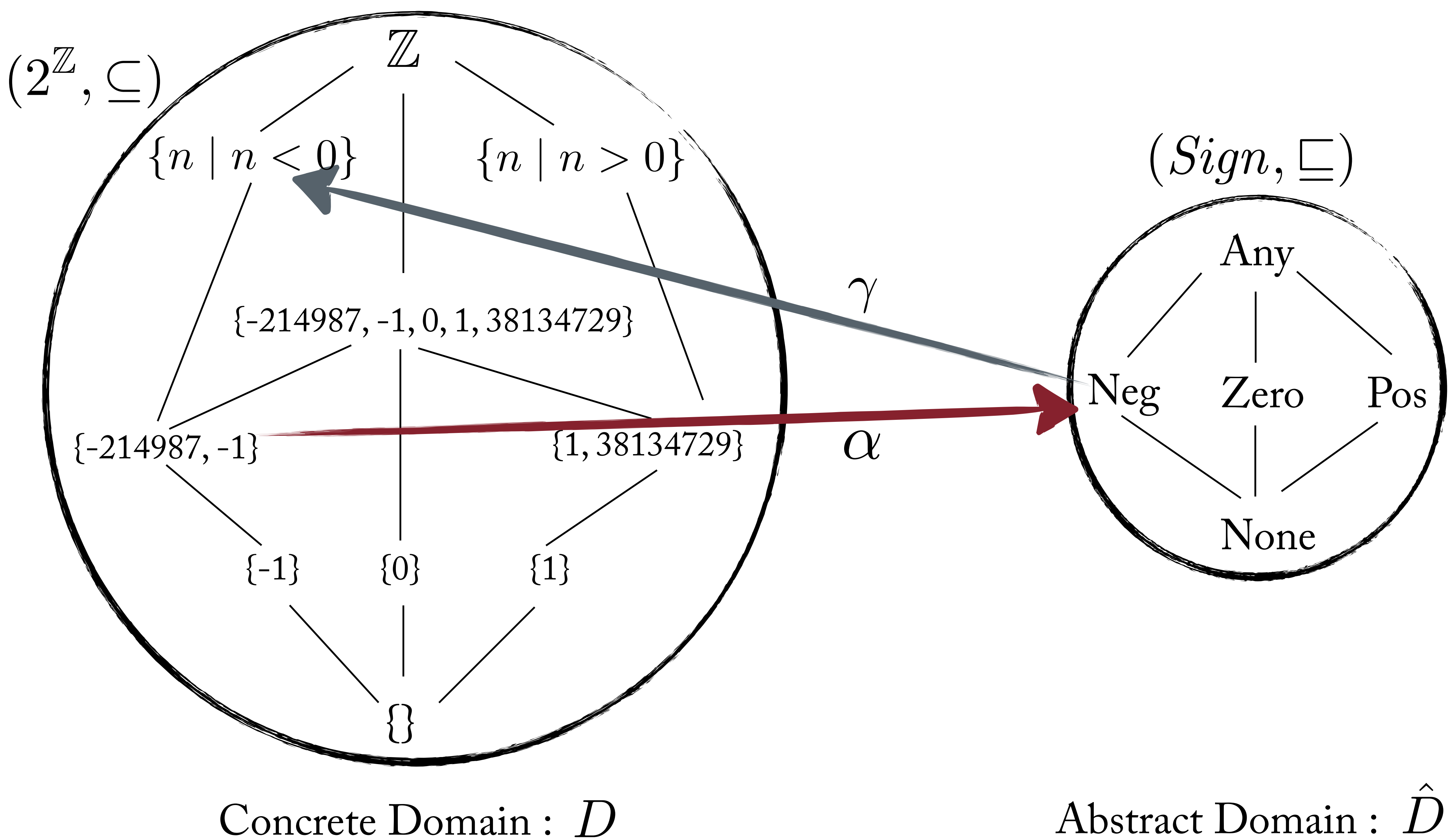
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$



What is the general condition that α and γ should satisfy?
~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$

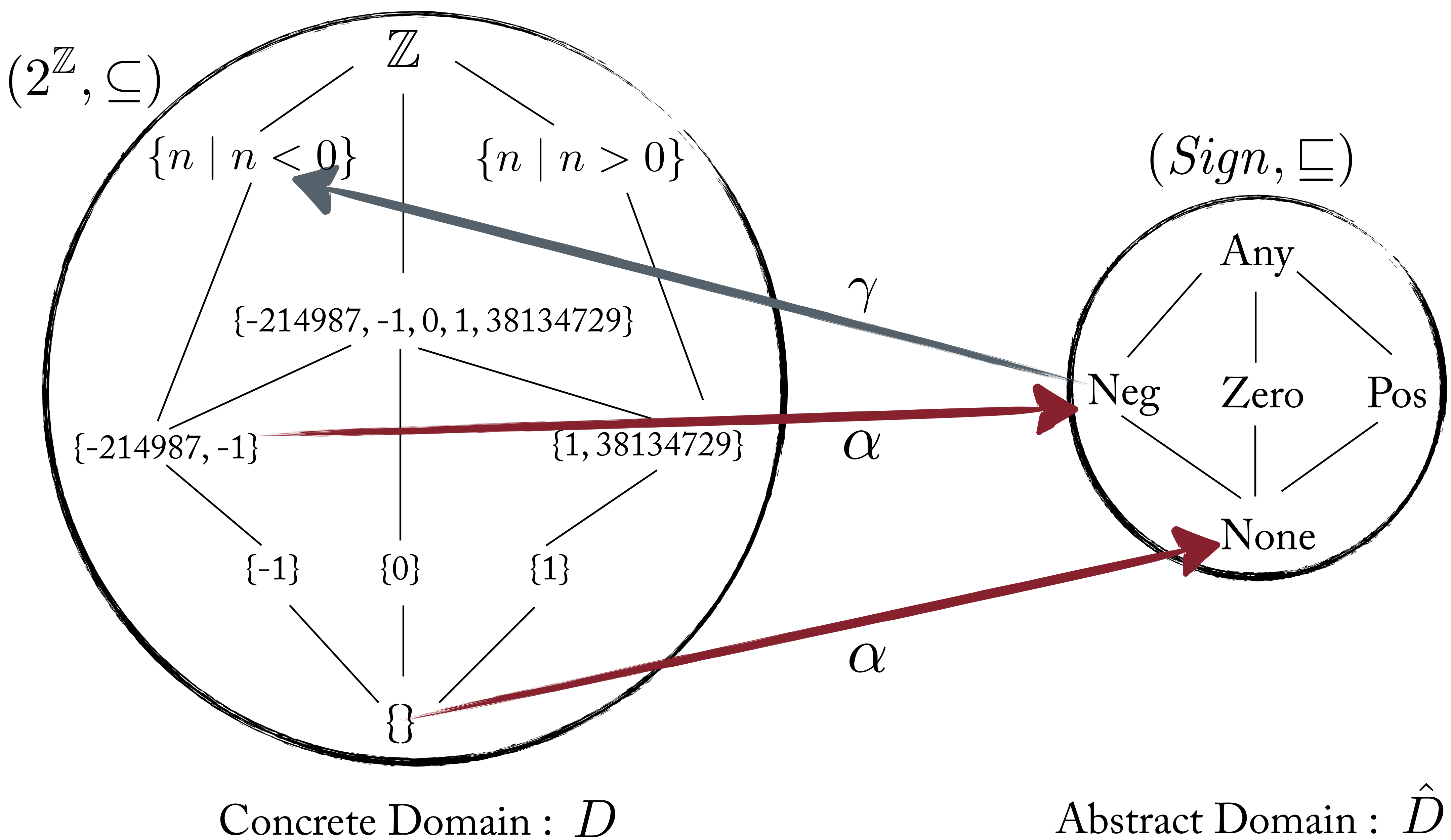


What is the general condition that α and γ should satisfy?

~ **Galois Connection.**

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\rightleftarrows}} (\hat{D}, \sqsubseteq)$$

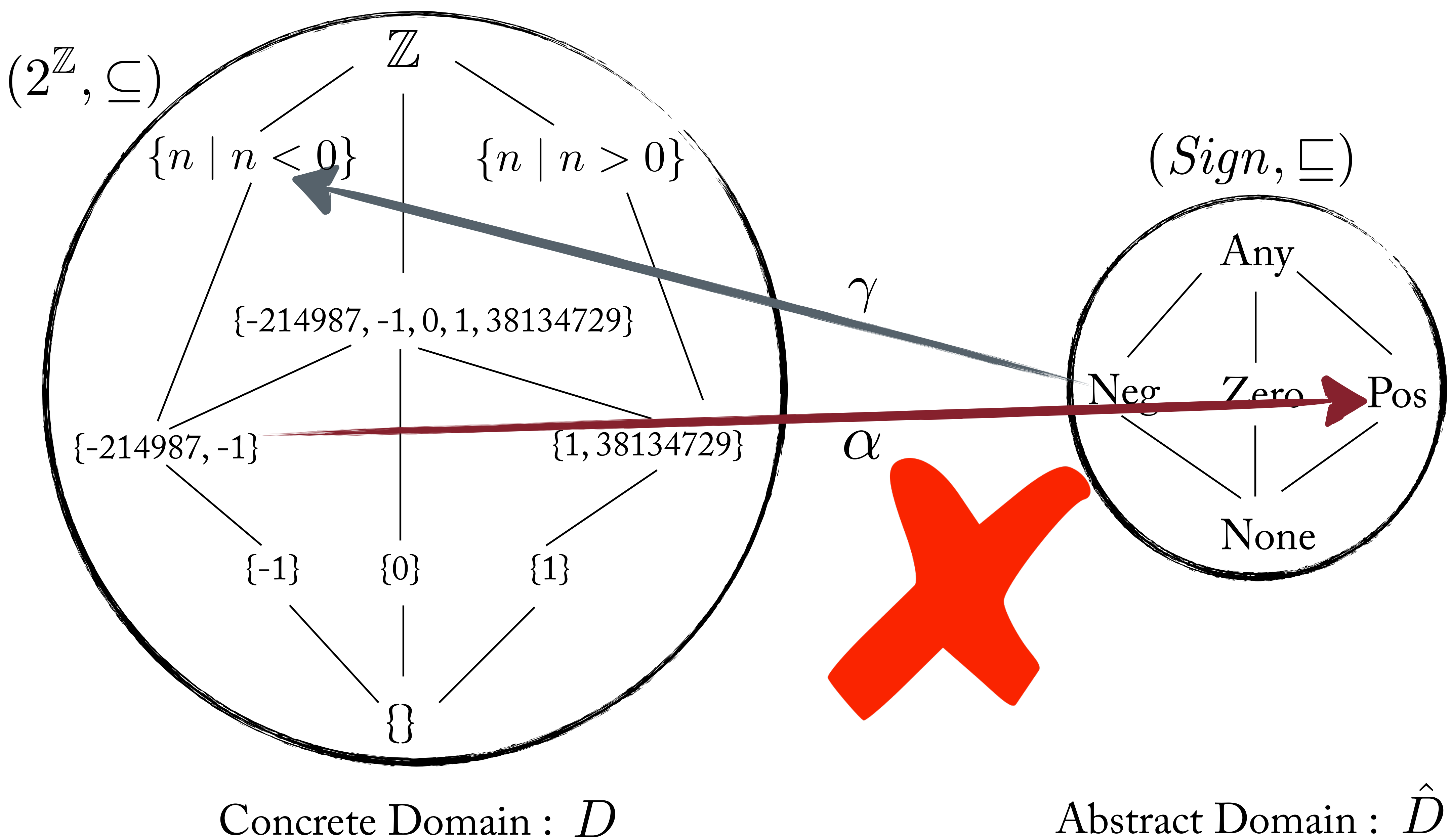
$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$

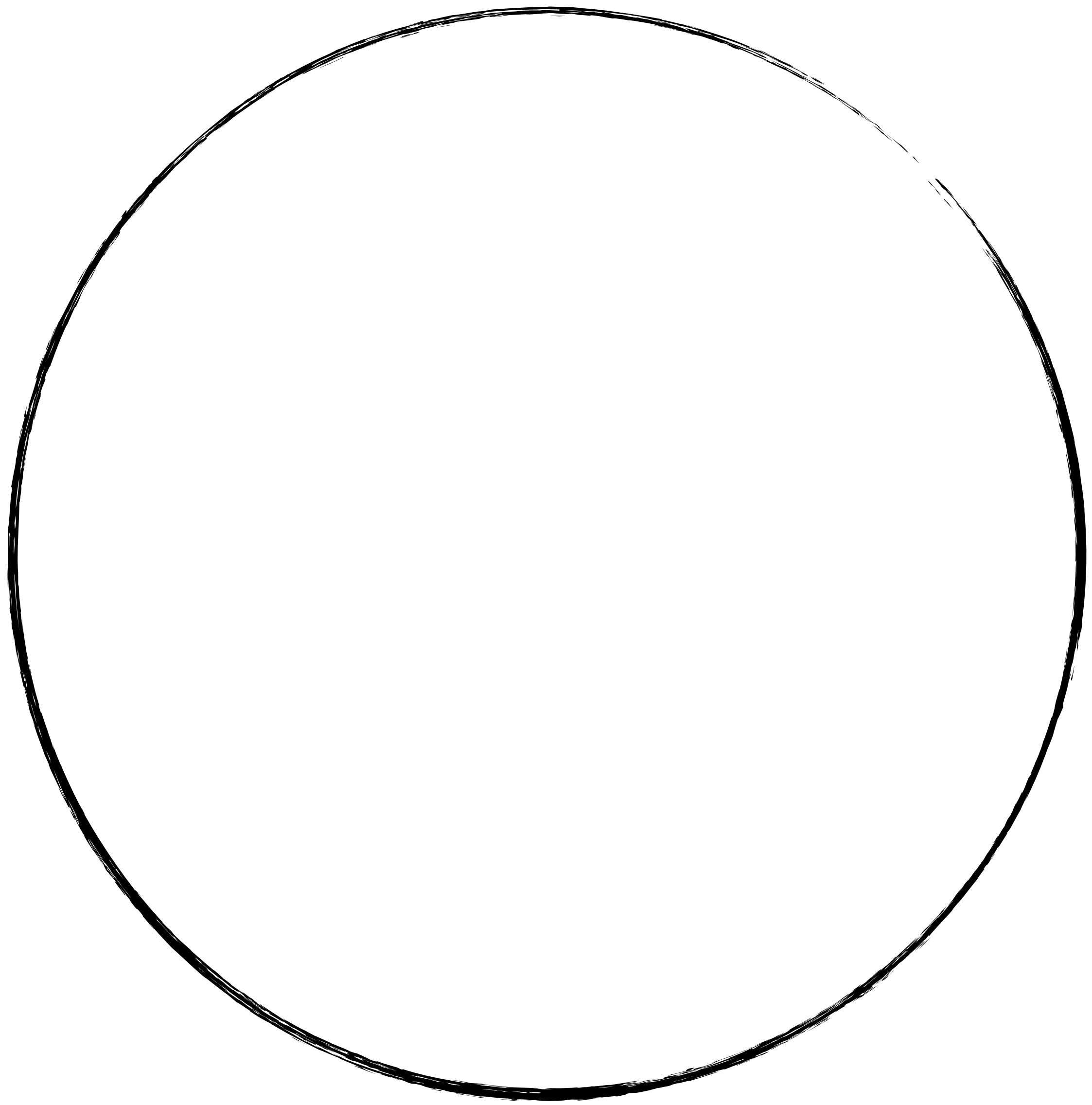


What is the general condition that α and γ should satisfy?
~ Galois Connection.

$$(D, \leq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\hat{D}, \sqsubseteq)$$

$$\forall x \in D, \hat{y} \in \hat{D} : \alpha(x) \sqsubseteq \hat{y} \iff x \leq \gamma(\hat{y})$$





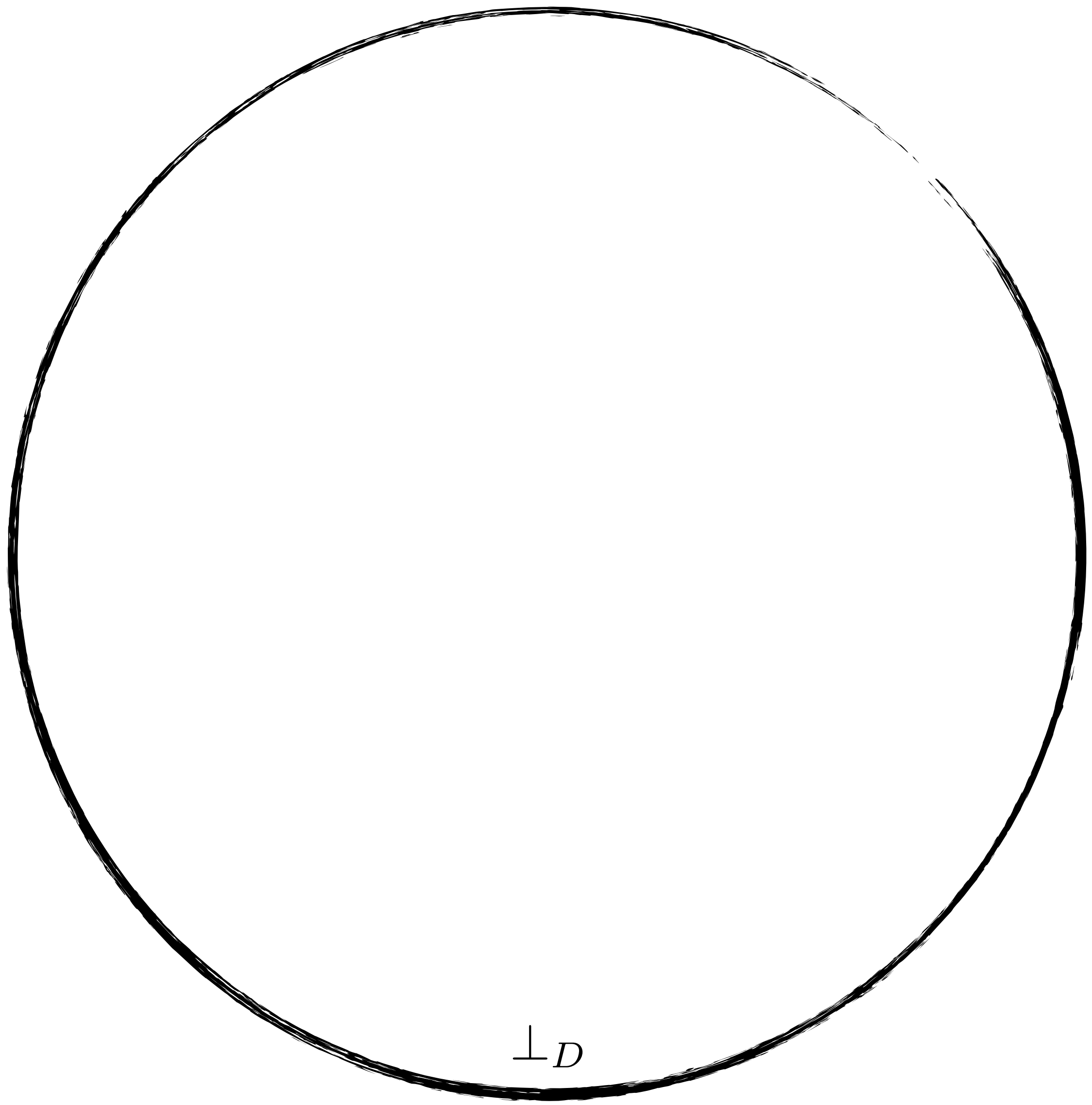
$$(D, \leq)$$

Concrete Domain

$$F : D \to D$$

Concrete Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



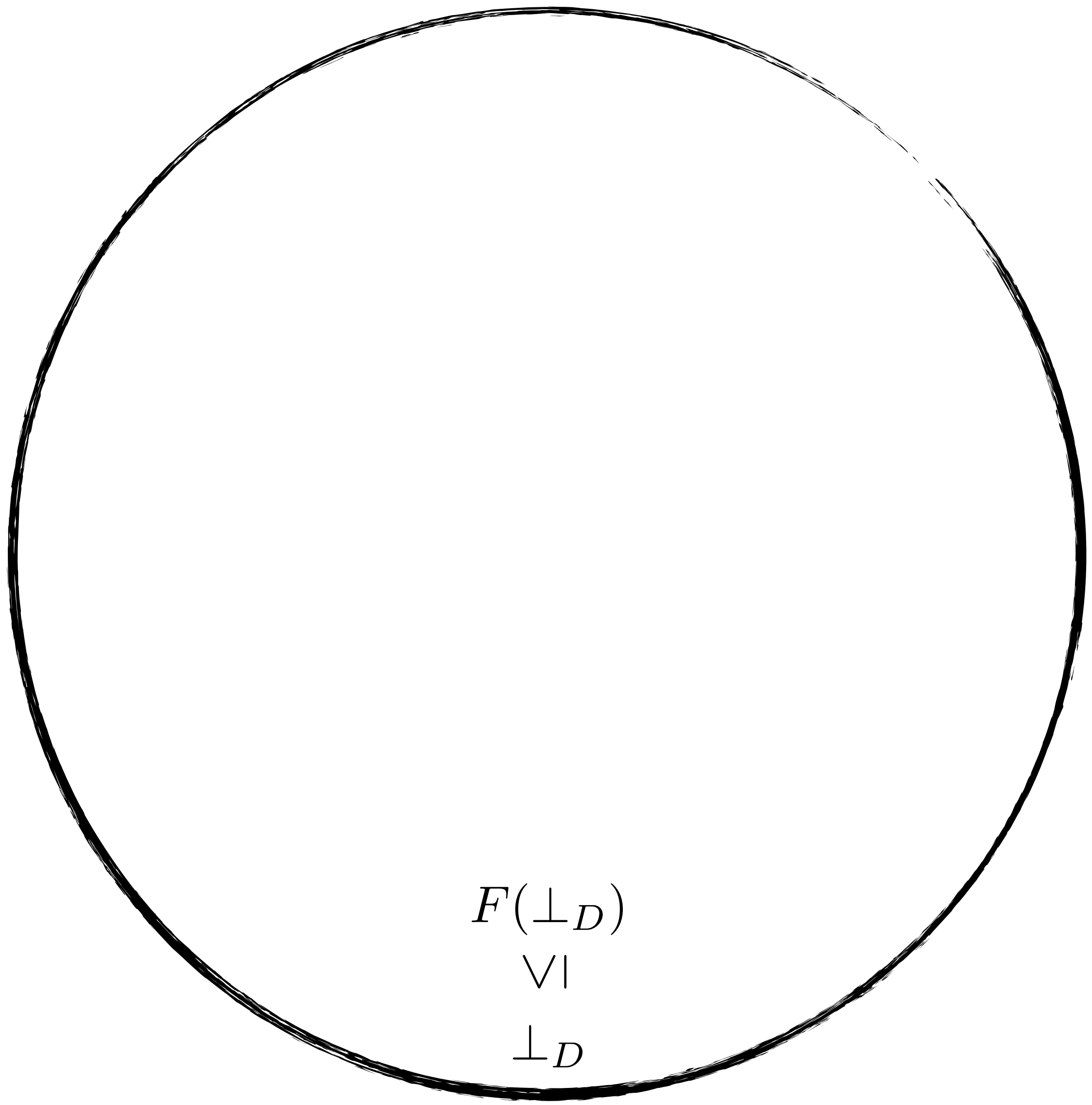
$$(D, \leq)$$

Concrete Domain

$$F : D \to D$$

Concrete Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\perp_D)$$



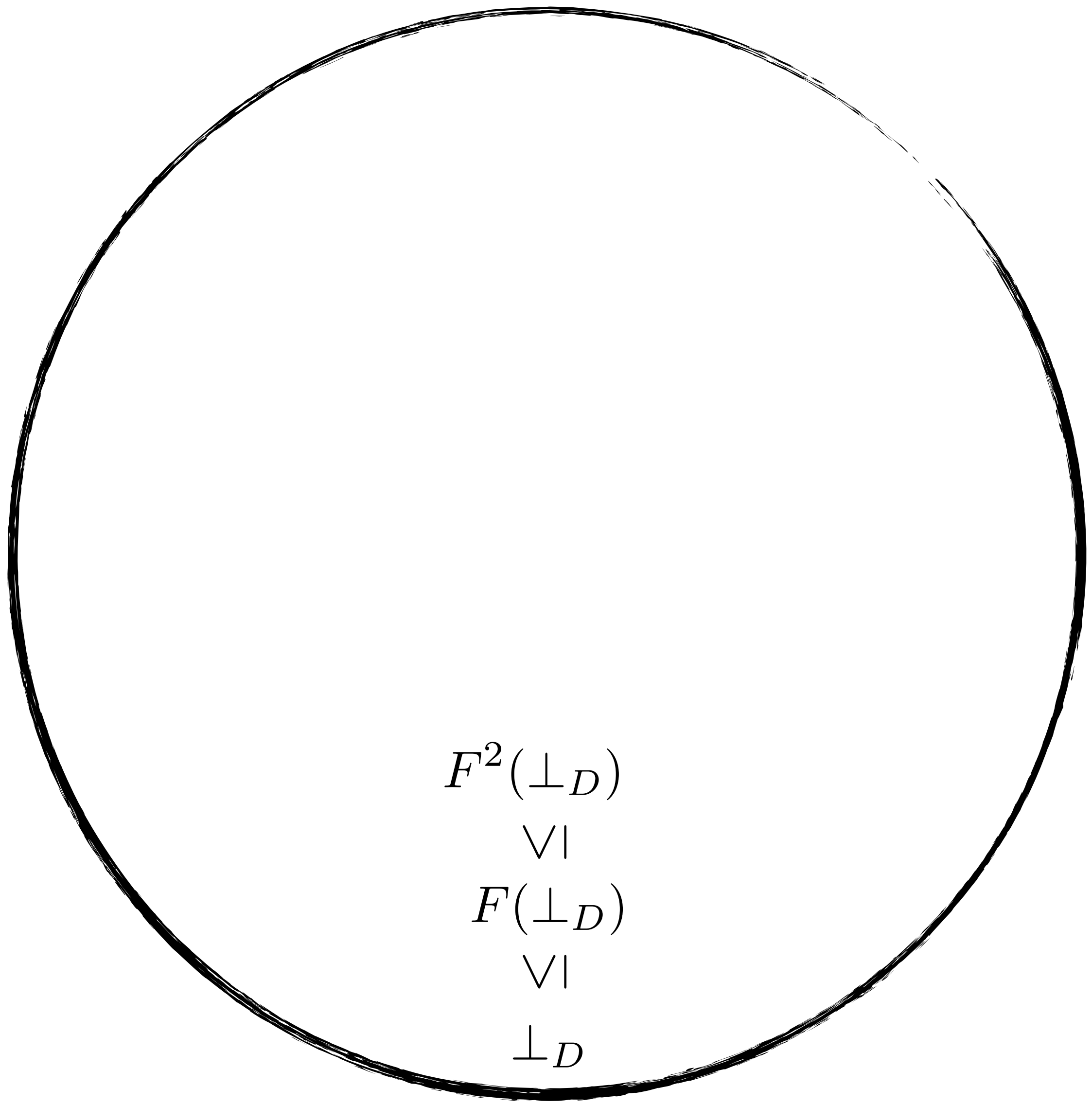
$$(D, \leq)$$

Concrete Domain

$$F : D \to D$$

Concrete Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



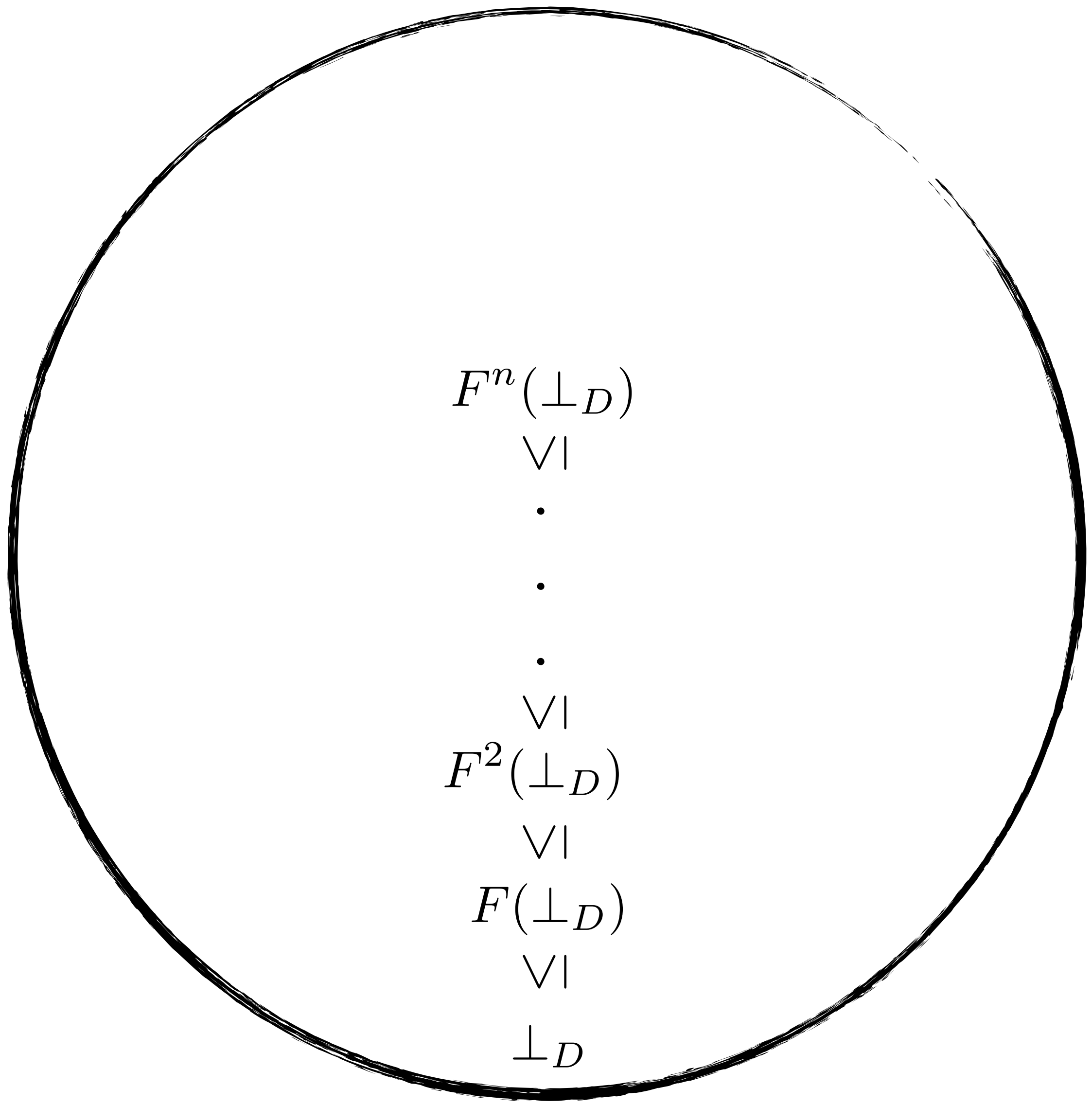
$$(D, \leq)$$

Concrete Domain

$$F : D \to D$$

Concrete Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



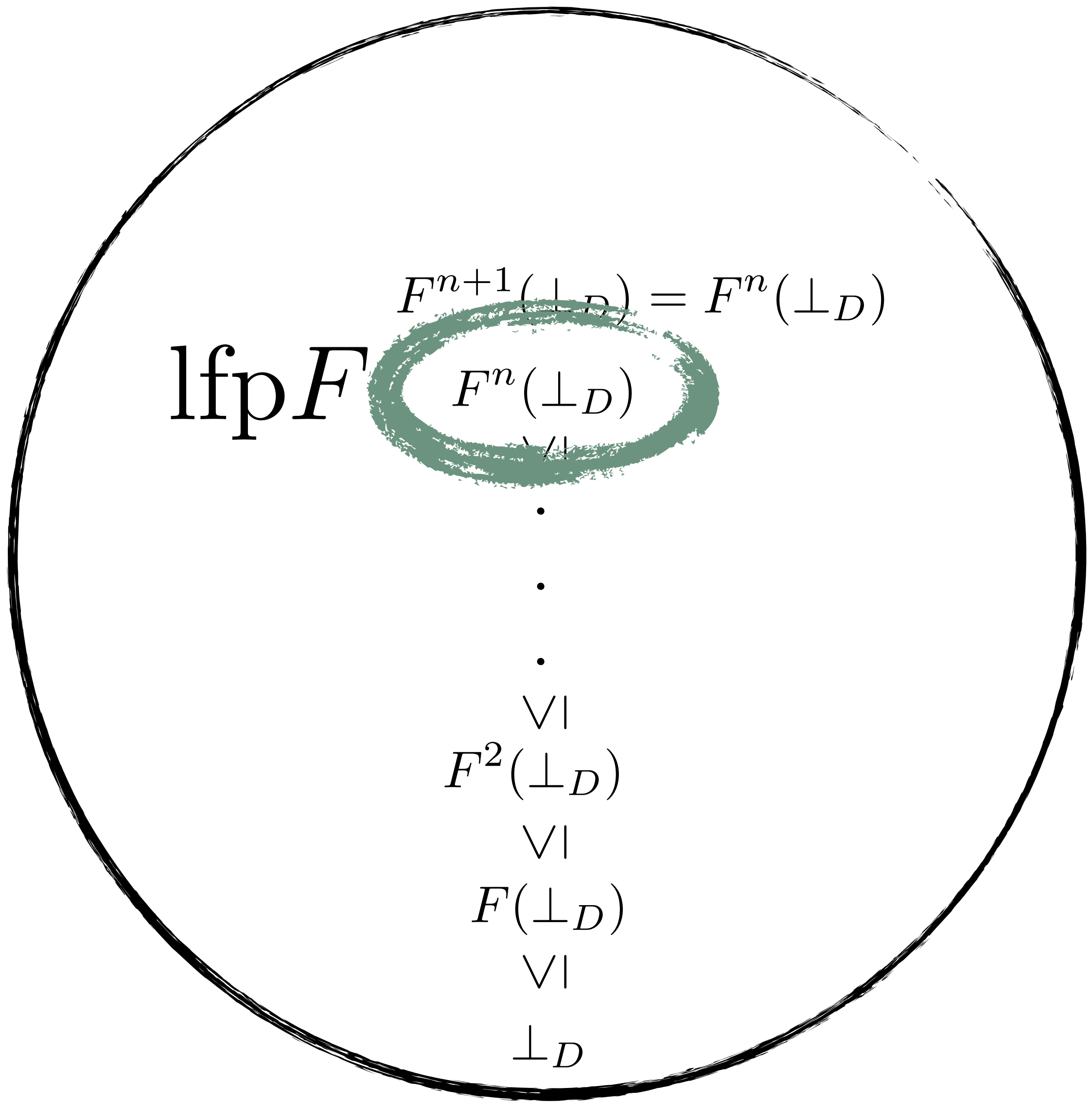
$$(D, \leq)$$

Concrete Domain

$$F : D \to D$$

Concrete Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



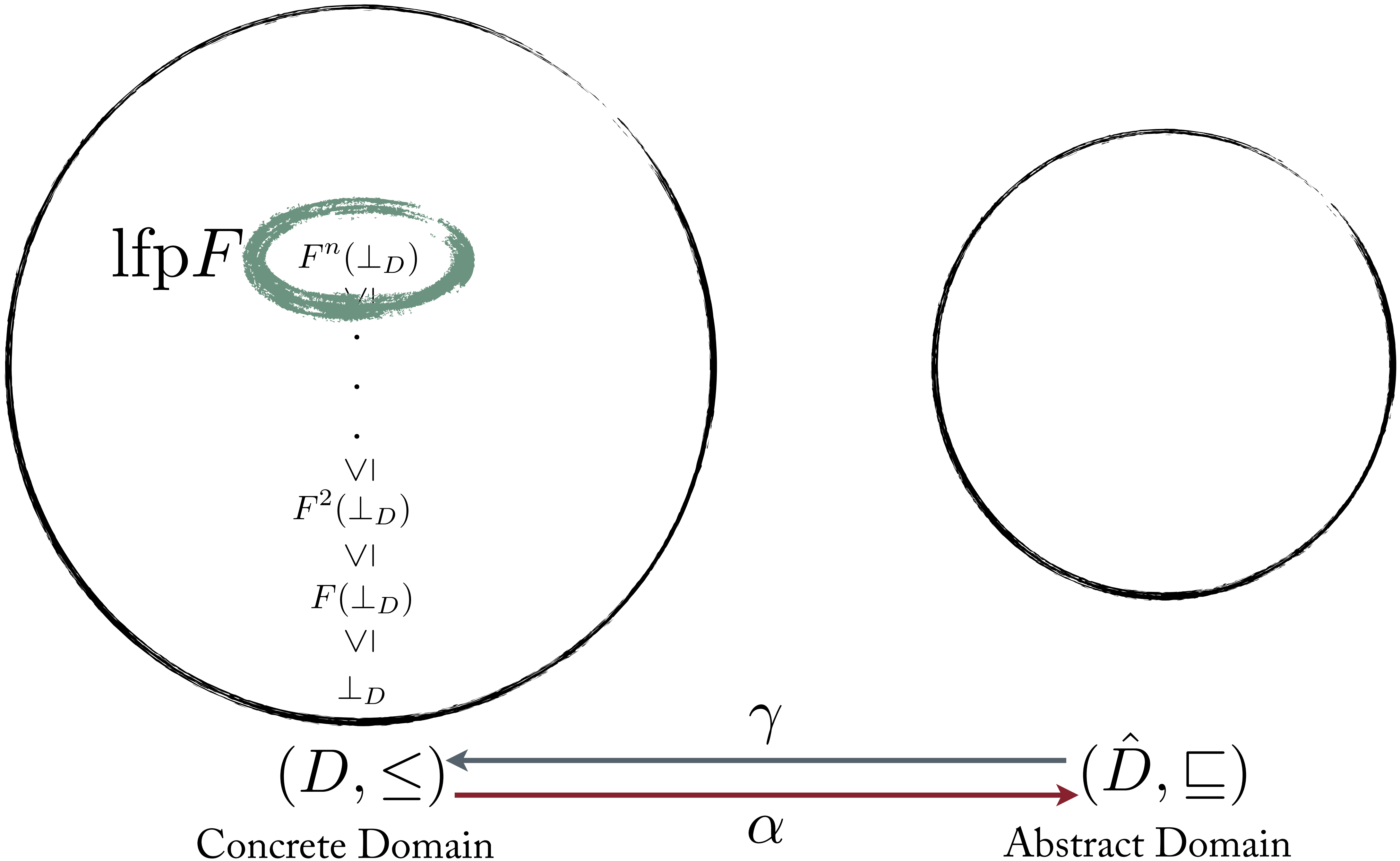
$(D, \leq)$

Concrete Domain

$F : D \to D$

Concrete Semantic Function

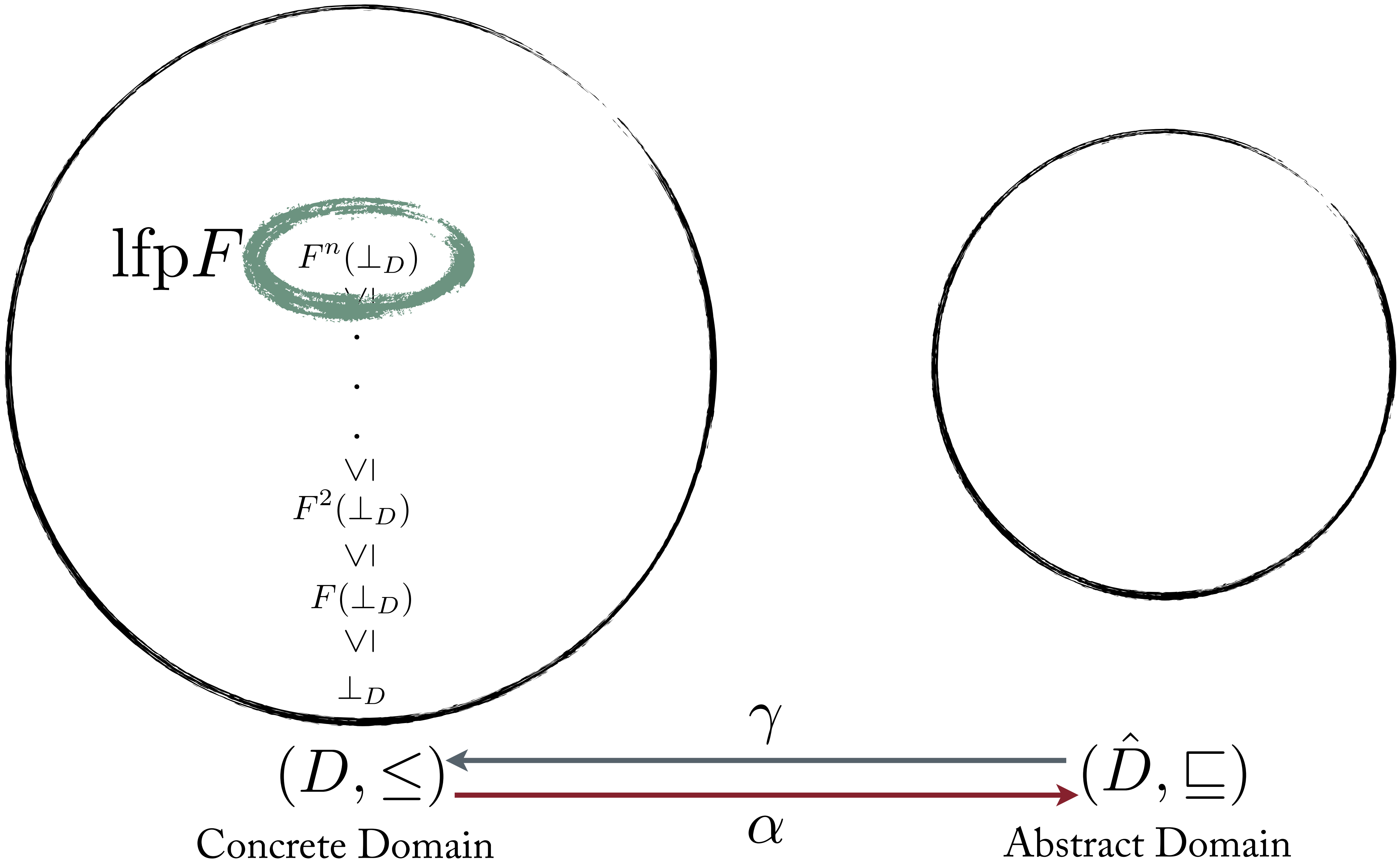
$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



$$F : D \to D$$

Concrete Semantic Function

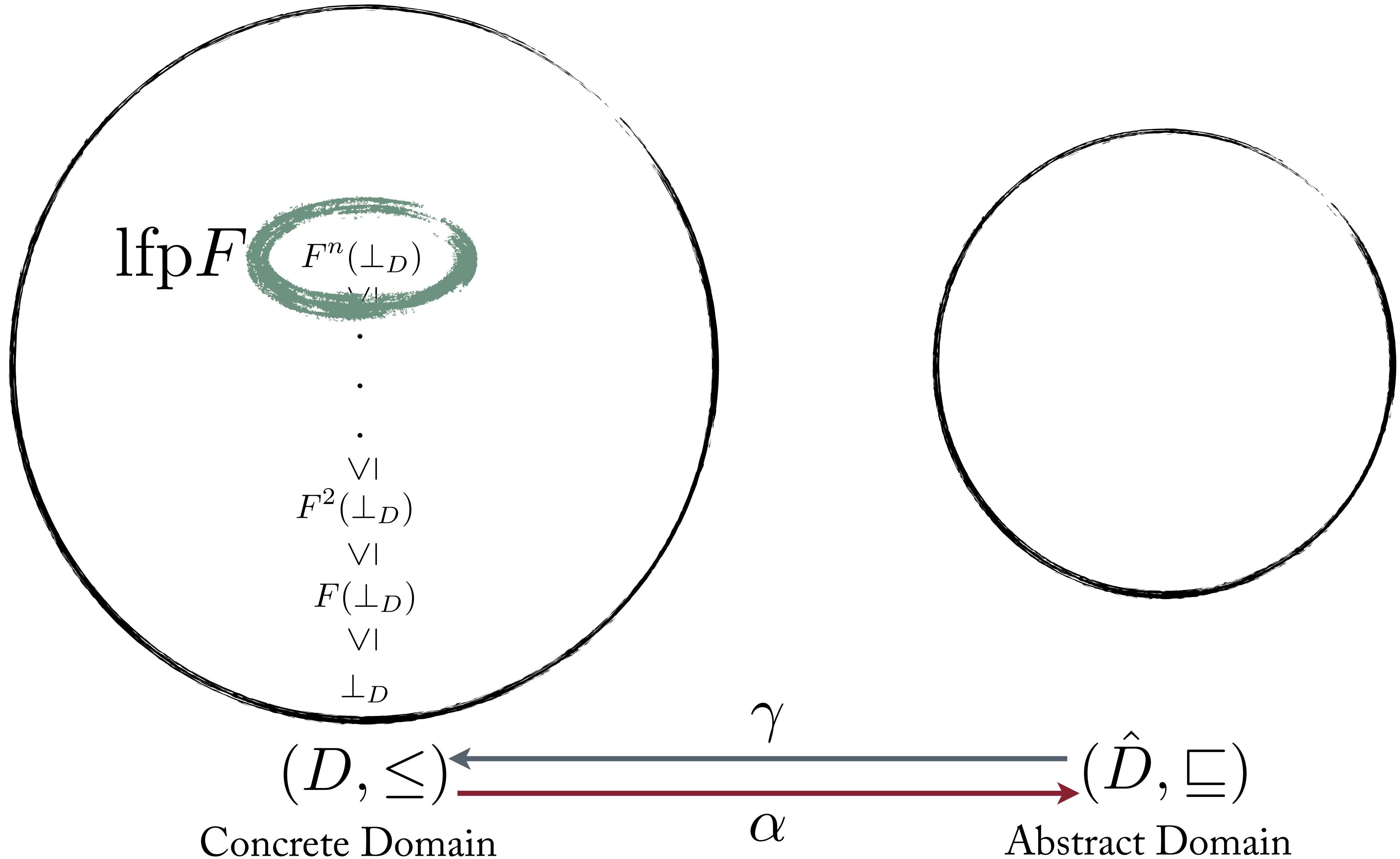
$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$



$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

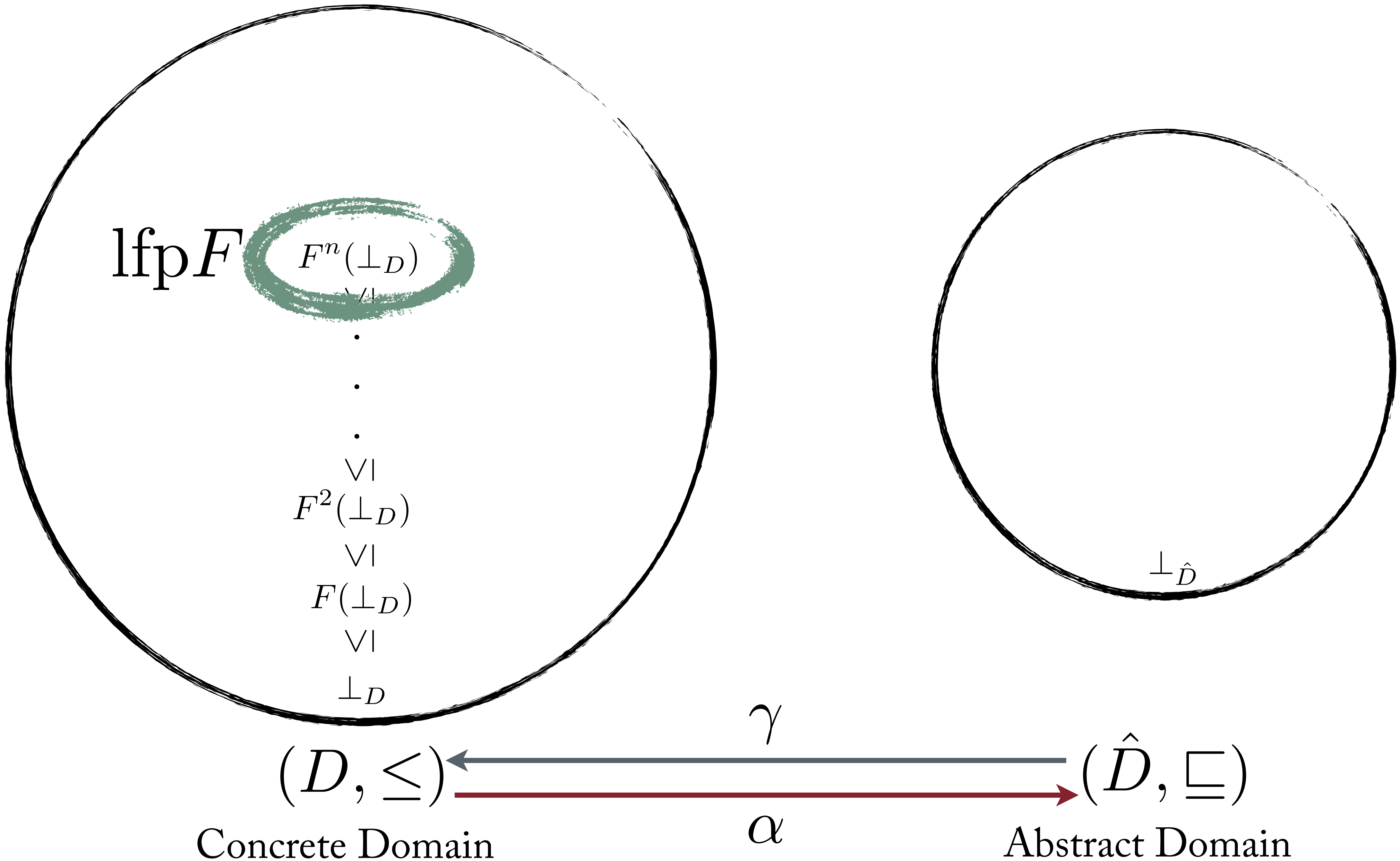


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$



$$F : D \rightarrow D$$

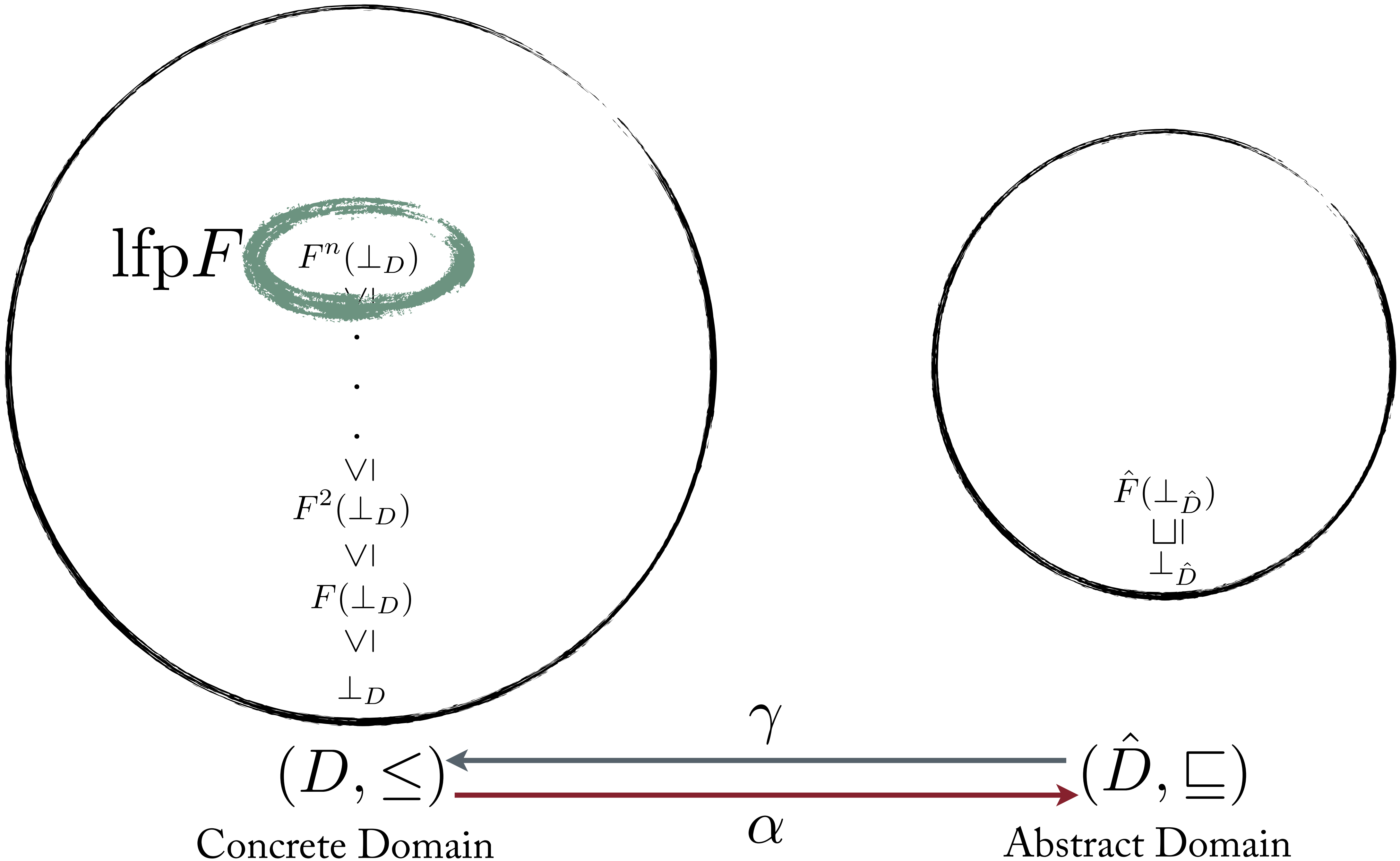
Concrete Semantic Function

$$\hat{F} : \hat{D} \rightarrow \hat{D}$$

Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

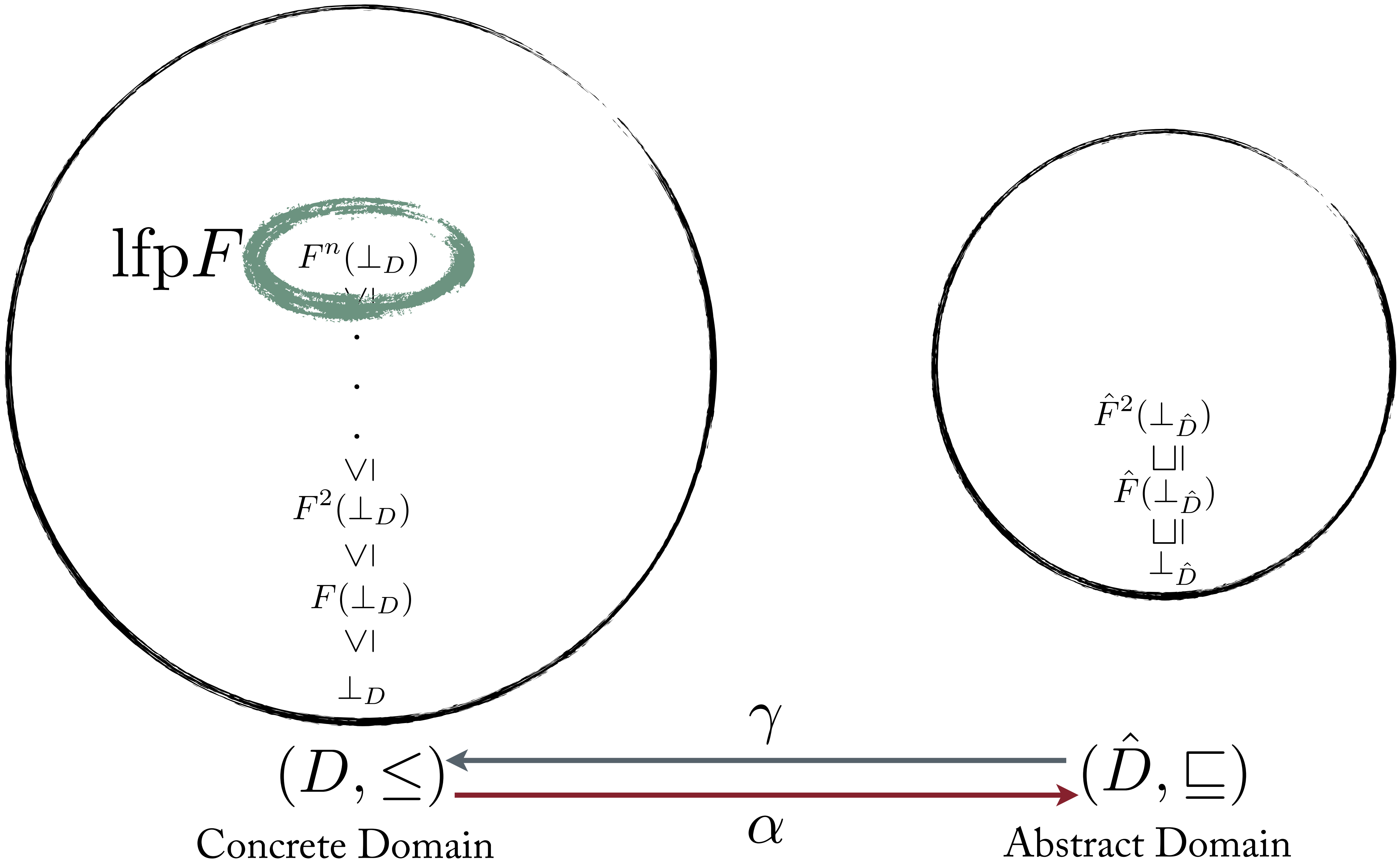


$F : D \rightarrow D$
Concrete Semantic Function

$\hat{F} : \hat{D} \rightarrow \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

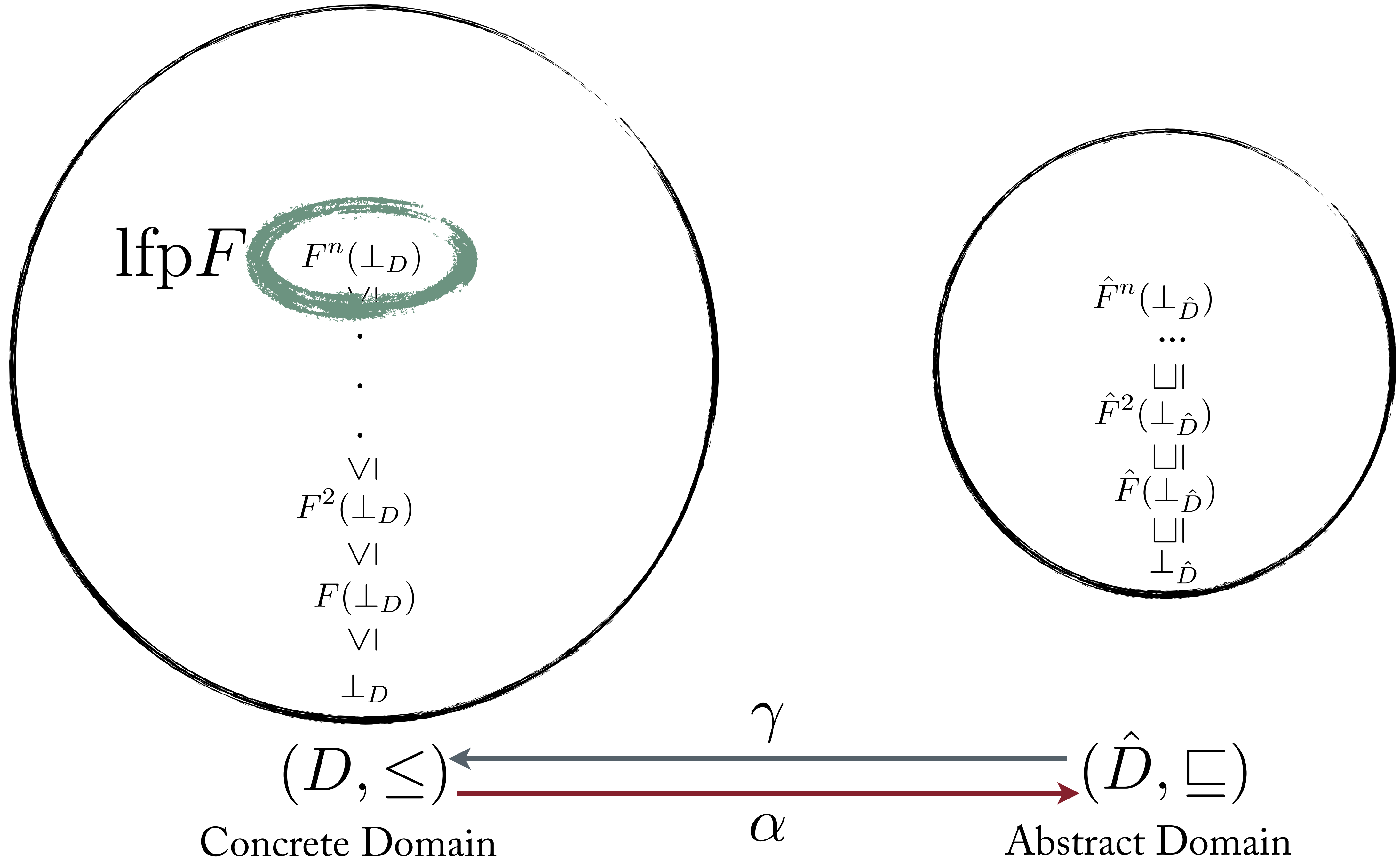


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

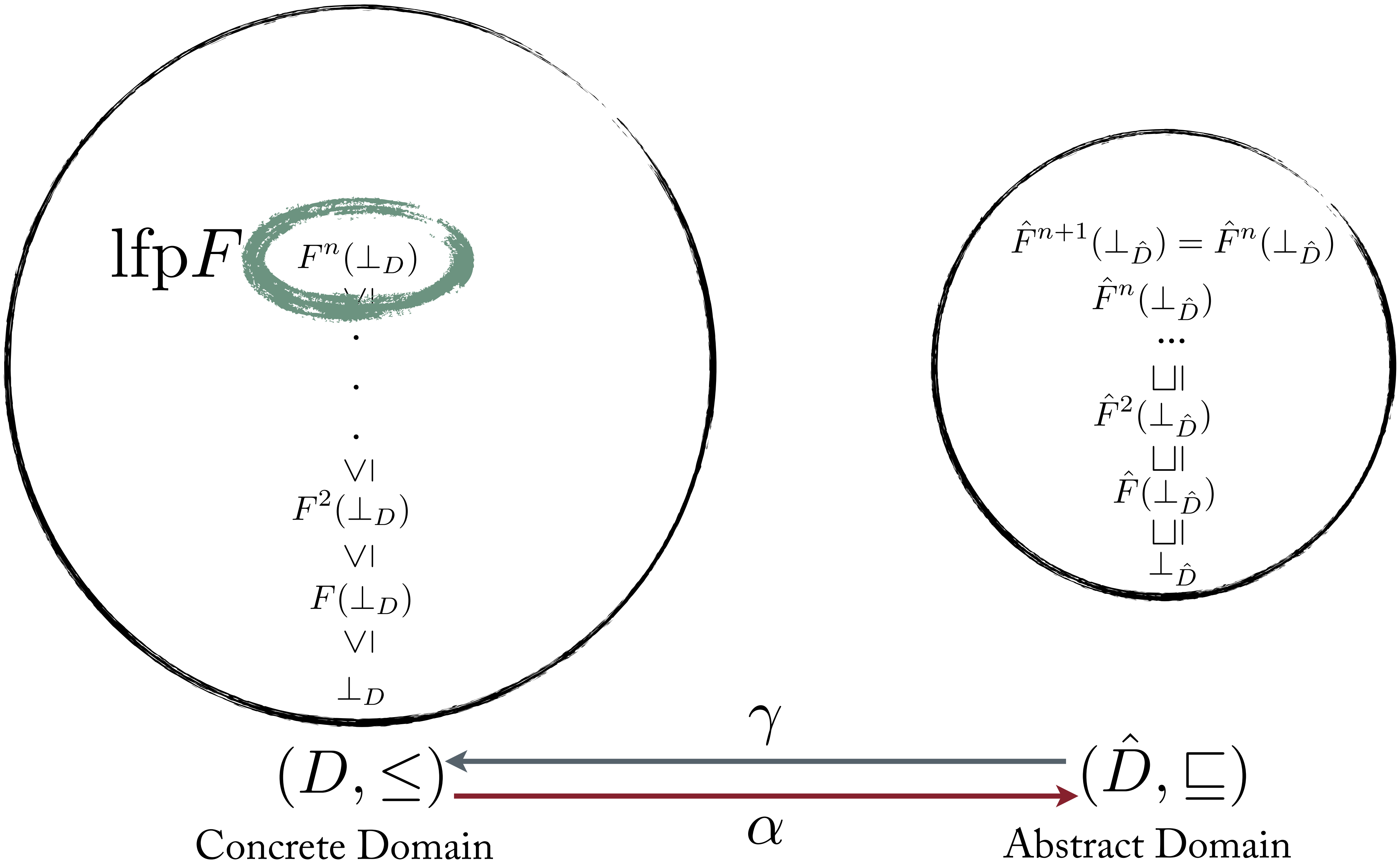


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

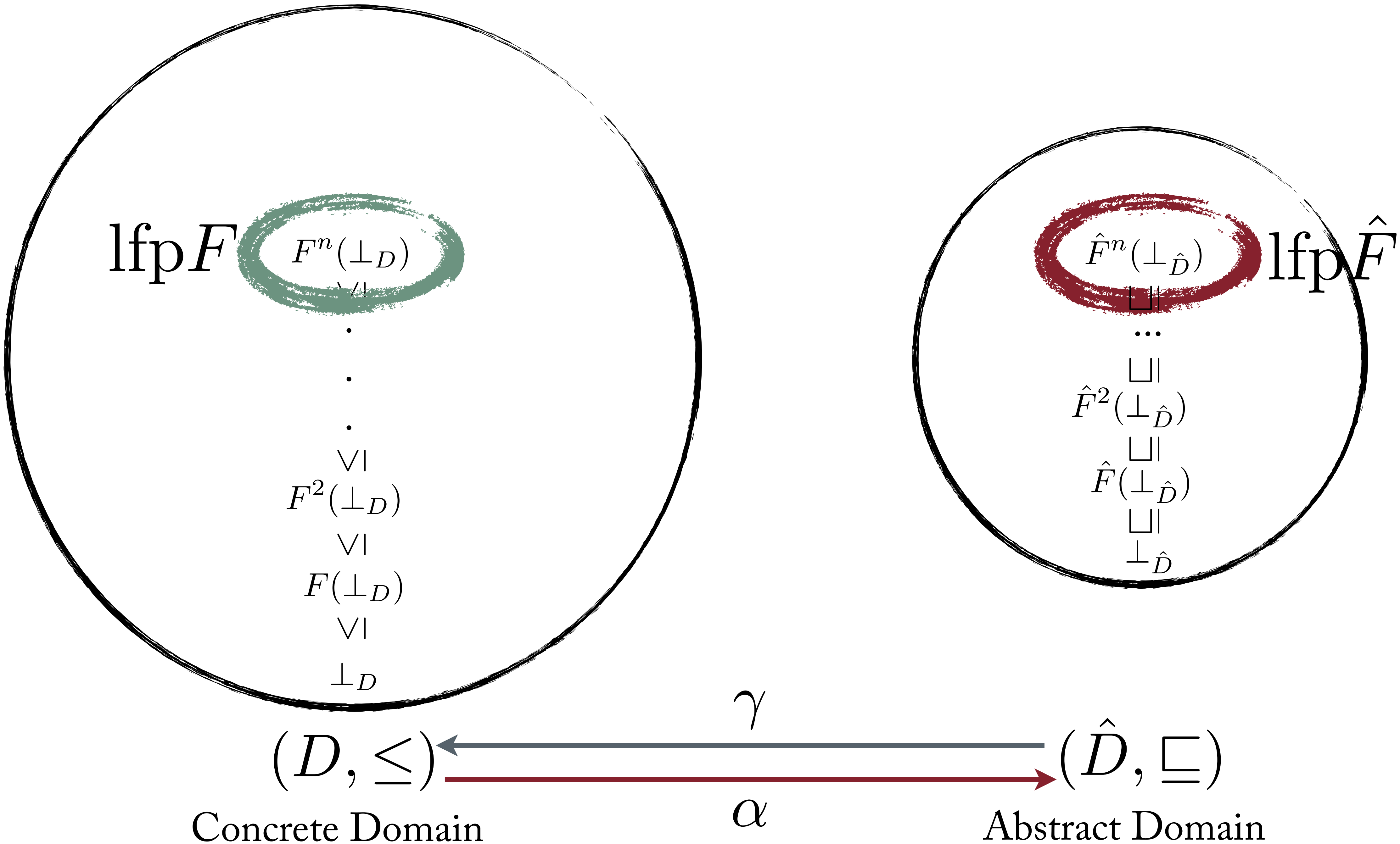


$F : D \rightarrow D$
Concrete Semantic Function

$\hat{F} : \hat{D} \rightarrow \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

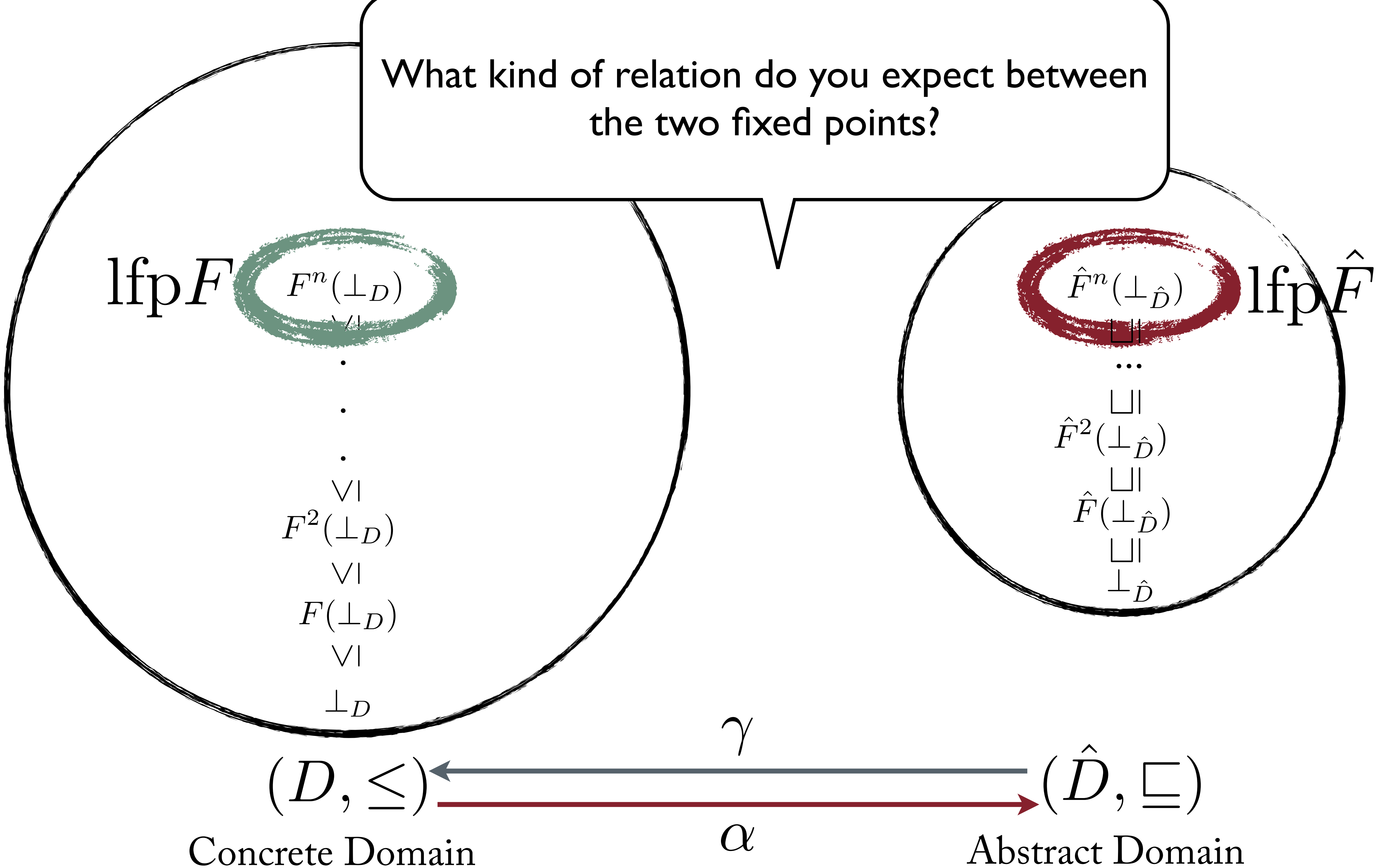


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\mathrm{lfp}\ F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\mathrm{lfp}\ \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

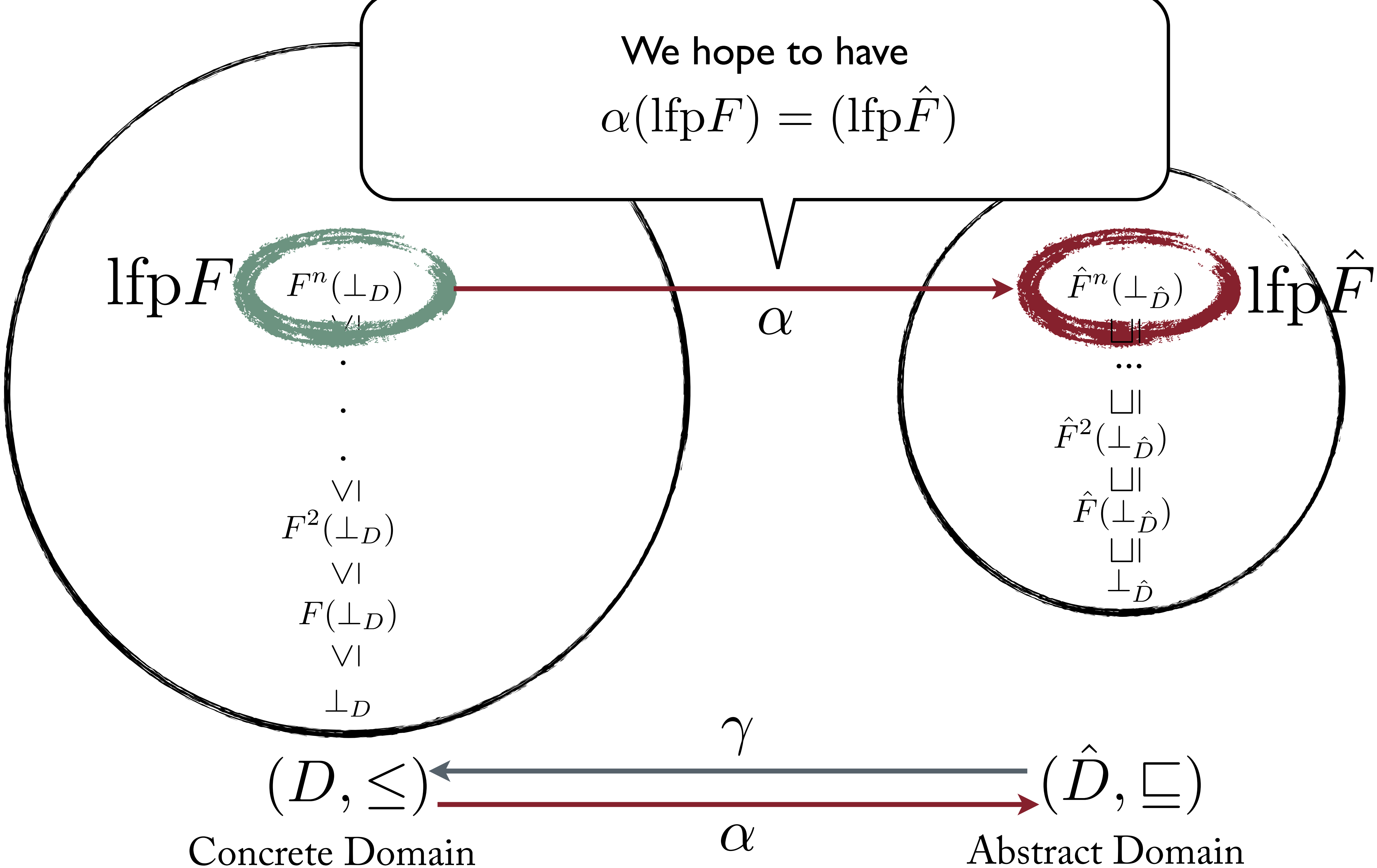


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

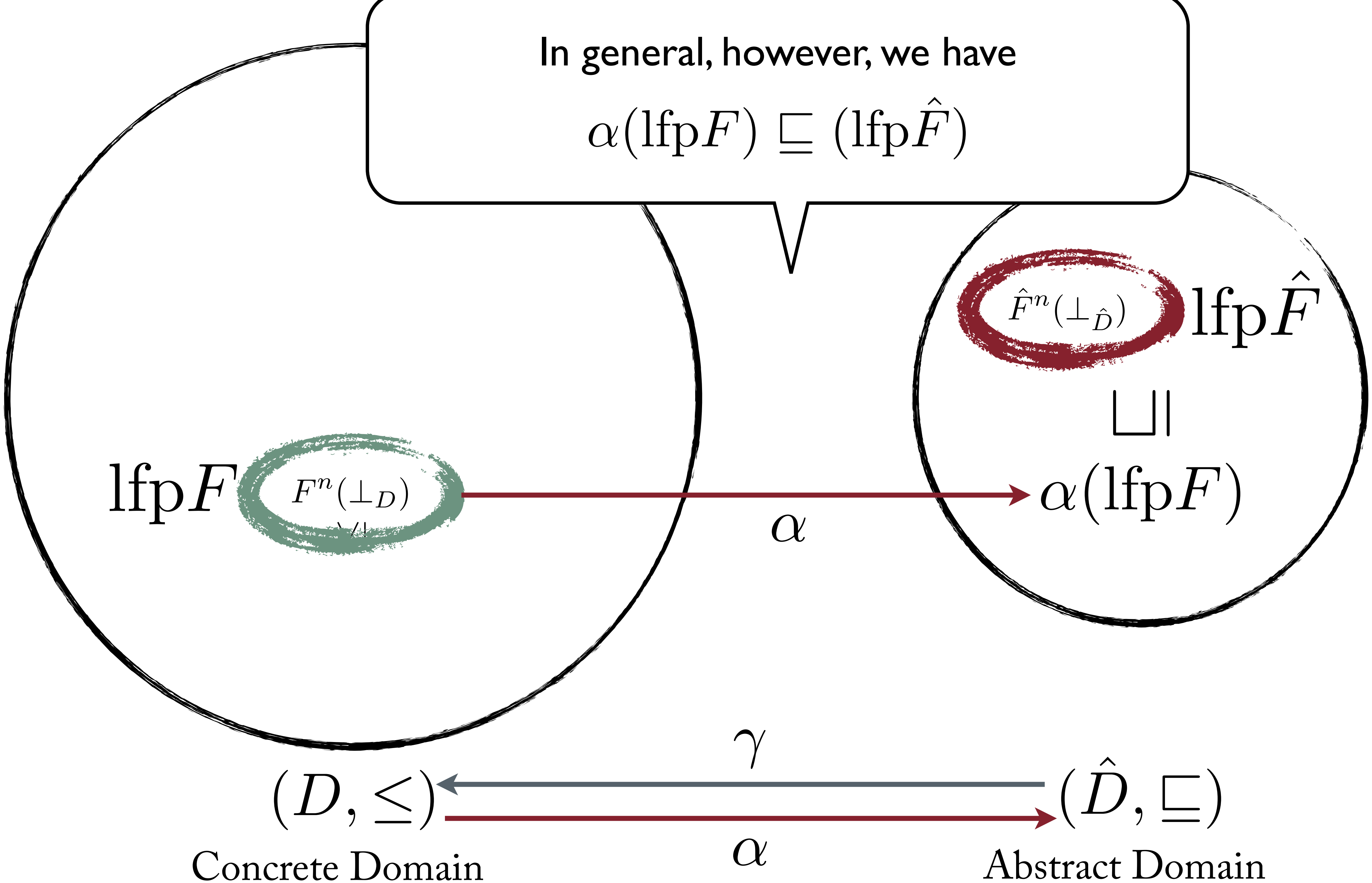


$F : D \rightarrow D$
Concrete Semantic Function

$\hat{F} : \hat{D} \rightarrow \hat{D}$
Abstract Semantic Function

$$\text{lfp } F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\text{lfp } \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

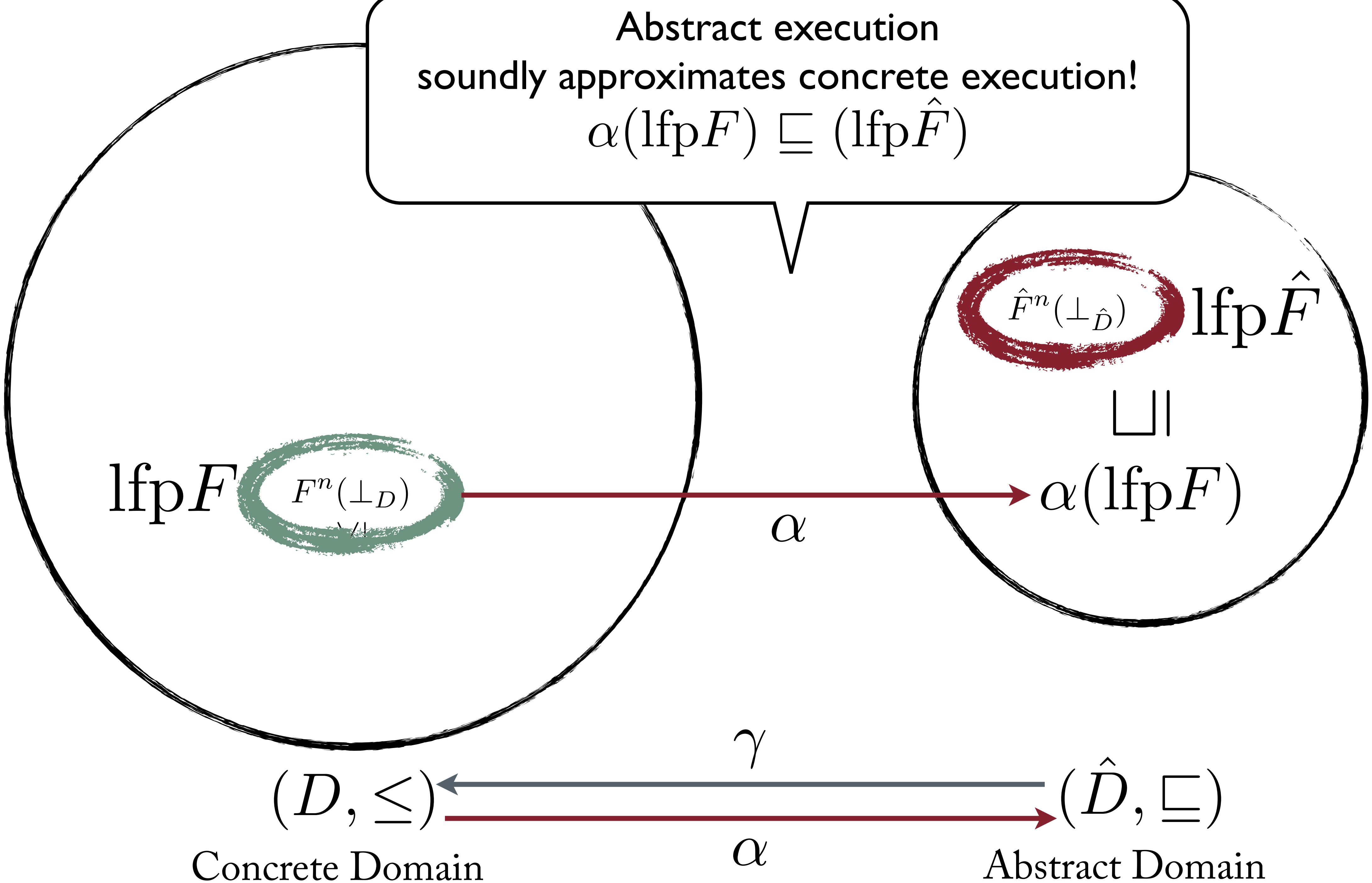


$F : D \to D$
Concrete Semantic Function

$\hat{F} : \hat{D} \to \hat{D}$
Abstract Semantic Function

$$\mathrm{lfp}\ F = \bigcup_{i \in \mathbb{N}} F^i(\perp_D)$$

$$\mathrm{lfp}\ \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\perp_{\hat{D}})$$

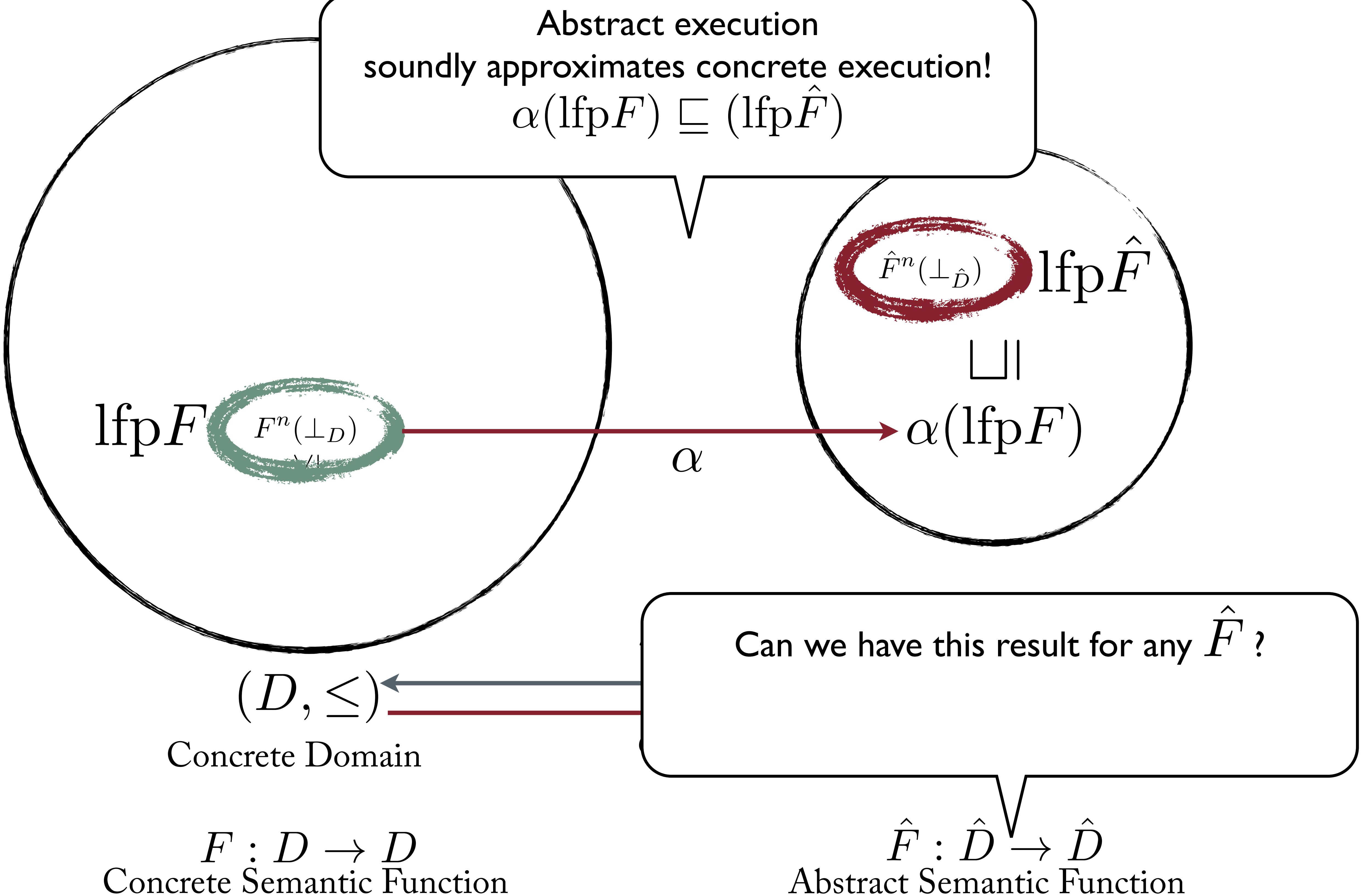


$F : D \rightarrow D$
Concrete Semantic Function

$\hat{F} : \hat{D} \rightarrow \hat{D}$
Abstract Semantic Function

$$\mathrm{lfp}\ F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\mathrm{lfp}\ \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

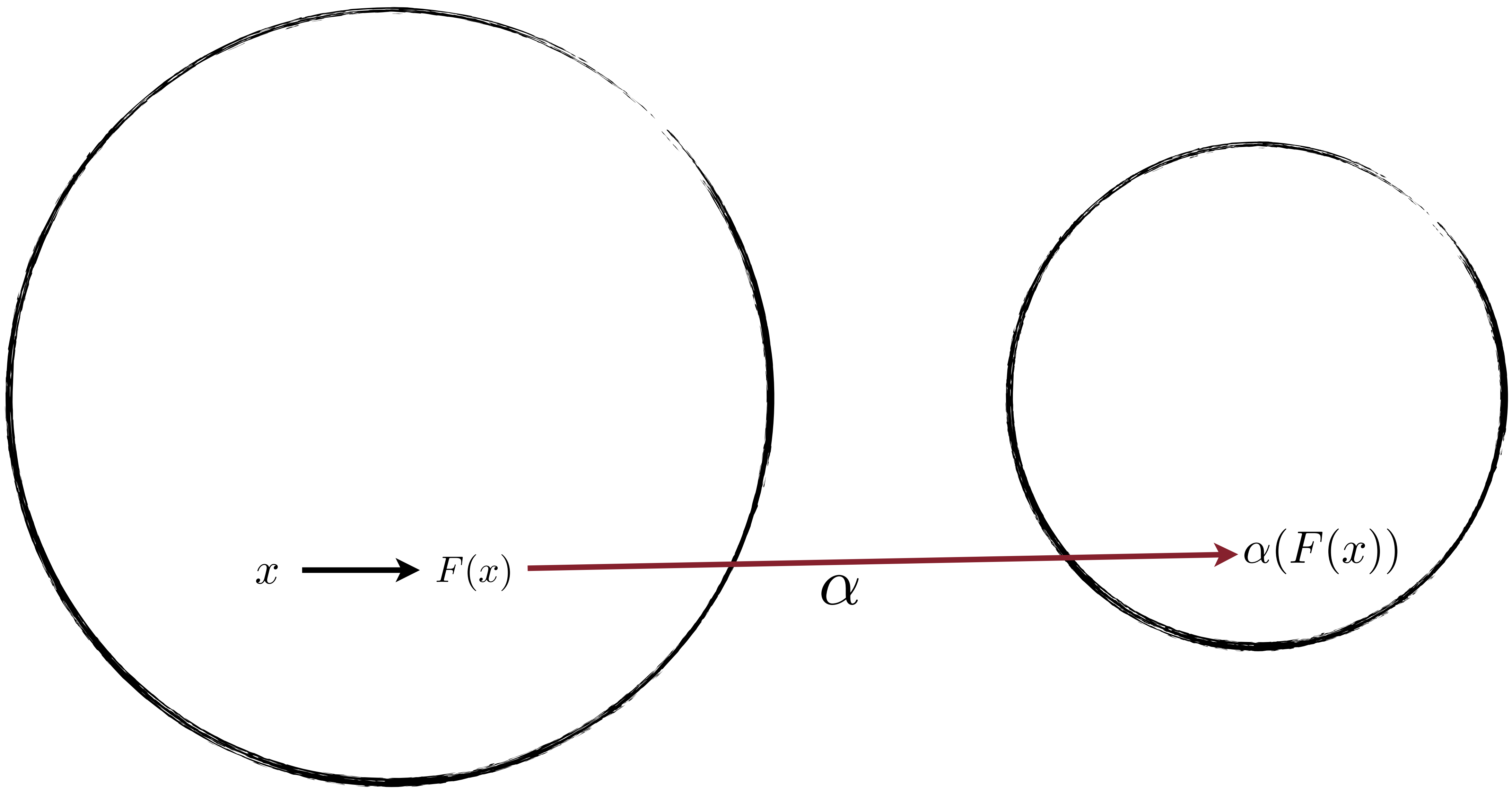


$$\mathrm{lfp}\ F = \bigcup_{i \in \mathbb{N}} F^i(\bot_D)$$

$$\mathrm{lfp}\ \hat{F} = \bigcup_{i \in \mathbb{N}} \hat{F}^i(\bot_{\hat{D}})$$

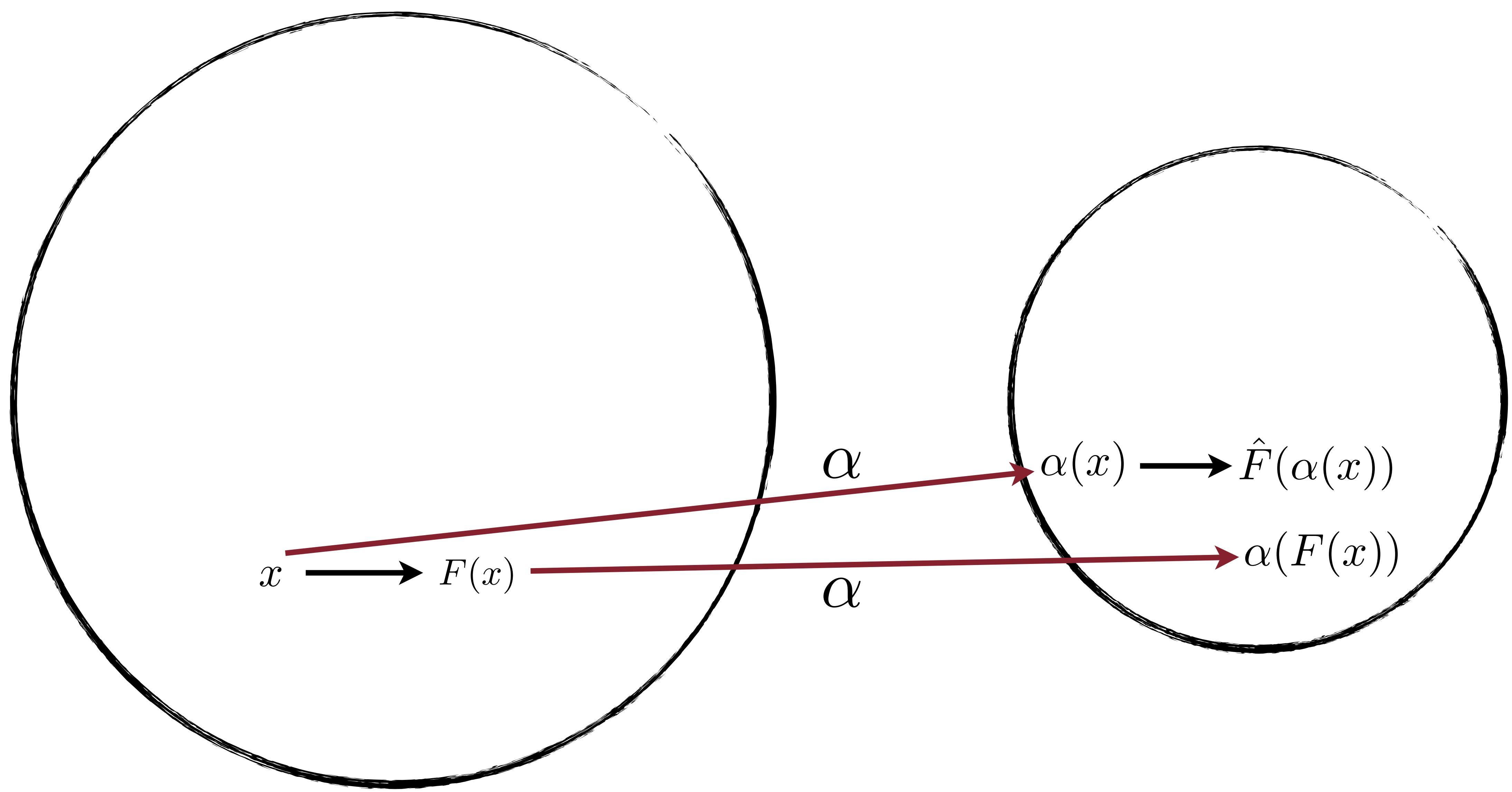
A condition for $\hat{F} : \hat{D} \to \hat{D}$ to have $\alpha(\text{lfp}F) \sqsubseteq (\text{lfp}\hat{F})$

1) monotone function.
2) $\forall x \in D : \alpha \circ F(x) \sqsubseteq \hat{F} \circ \alpha(x)$



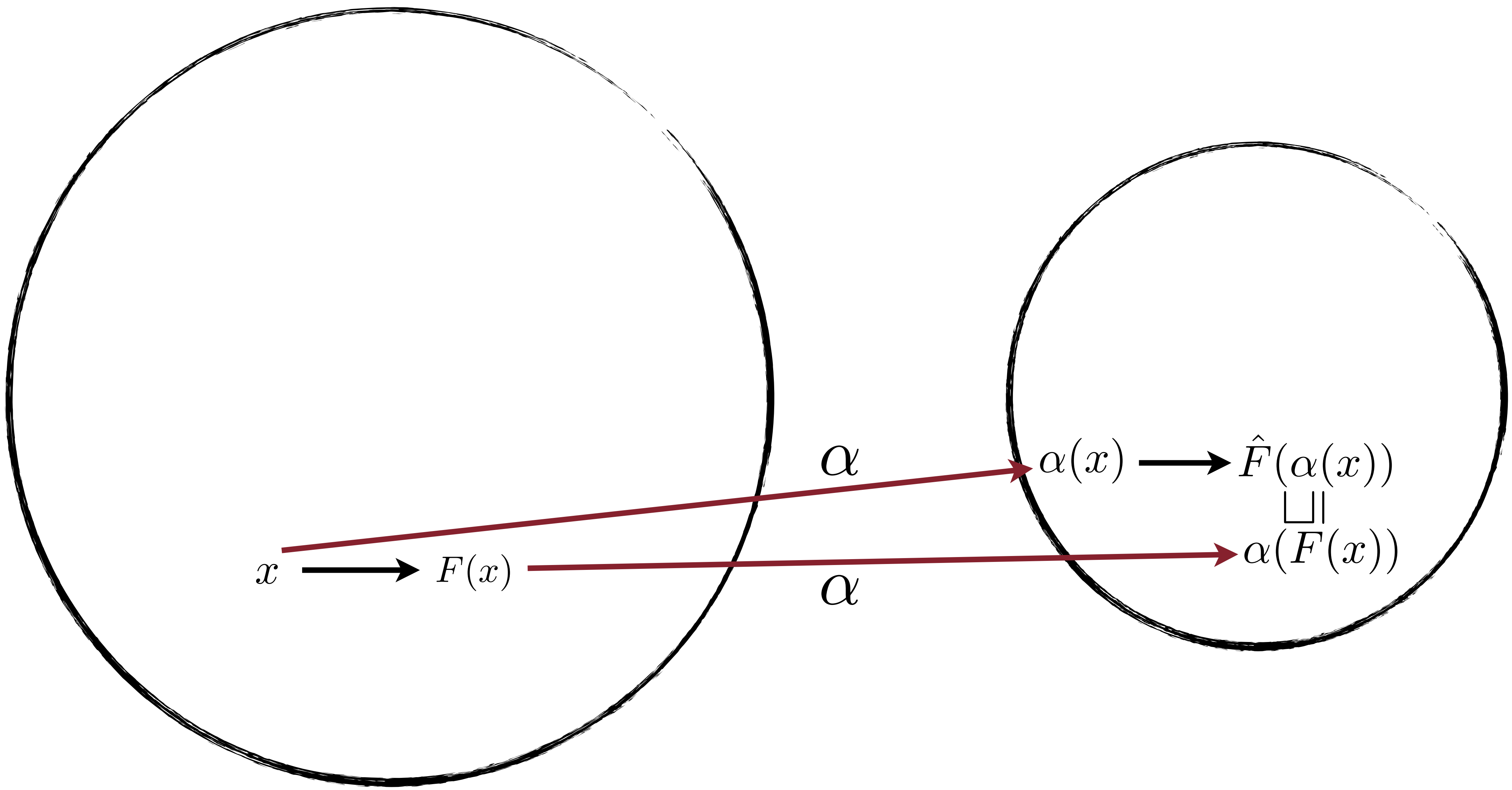
Condition for $\hat{F} : \hat{D} \rightarrow \hat{D}$ to have $\alpha(\text{lfp}F) \sqsubseteq (\text{lfp}\hat{F})$

1) monotone function.
2) $\forall x \in D : \alpha \circ F(x) \sqsubseteq \hat{F} \circ \alpha(x)$



Condition for $\hat{F} : \hat{D} \to \hat{D}$ to have $\alpha(\text{lfp}F) \sqsubseteq (\text{lfp}\hat{F})$

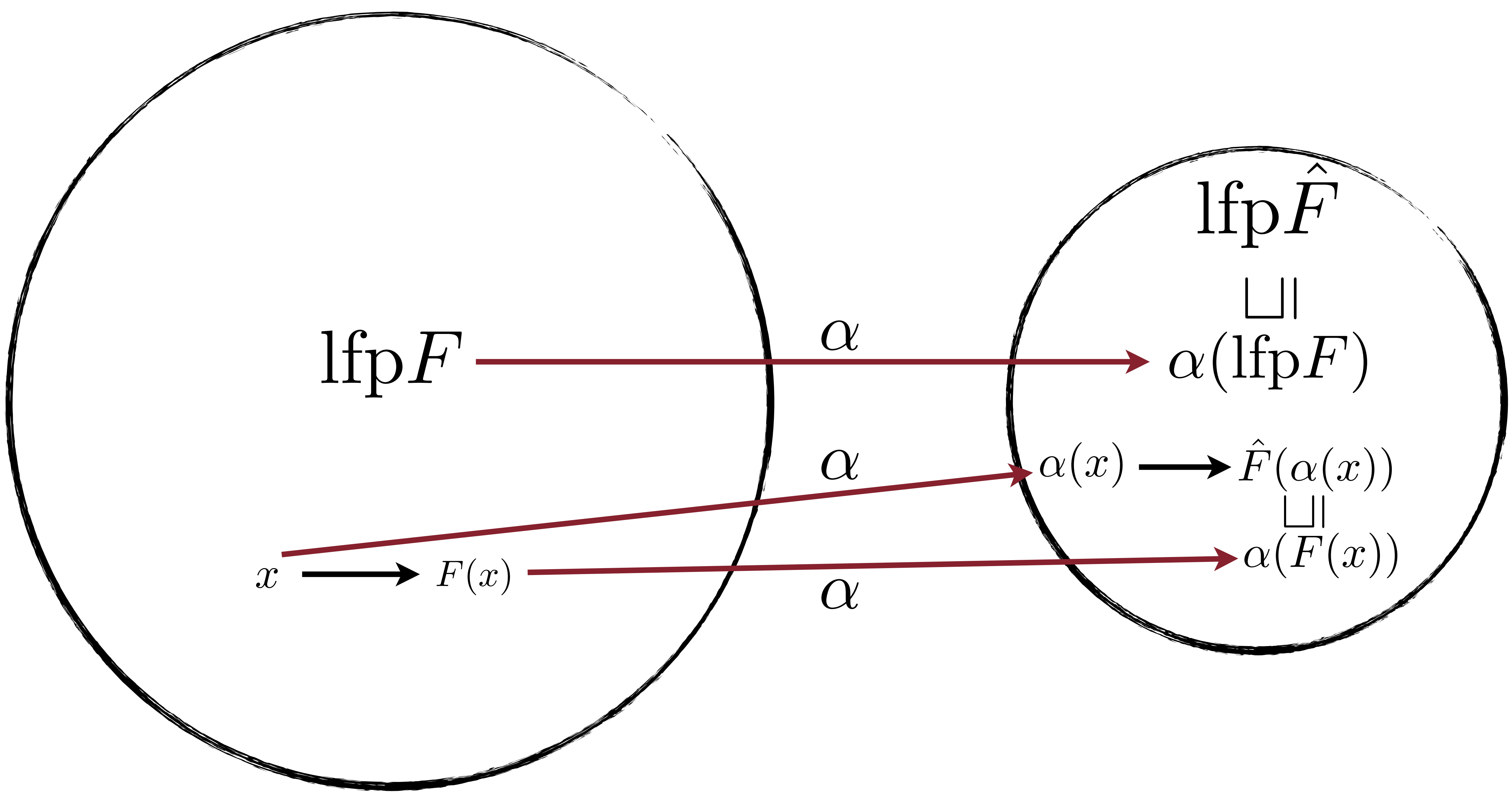
1) monotone function.
2) $\forall x \in D : \alpha \circ F(x) \sqsubseteq \hat{F} \circ \alpha(x)$



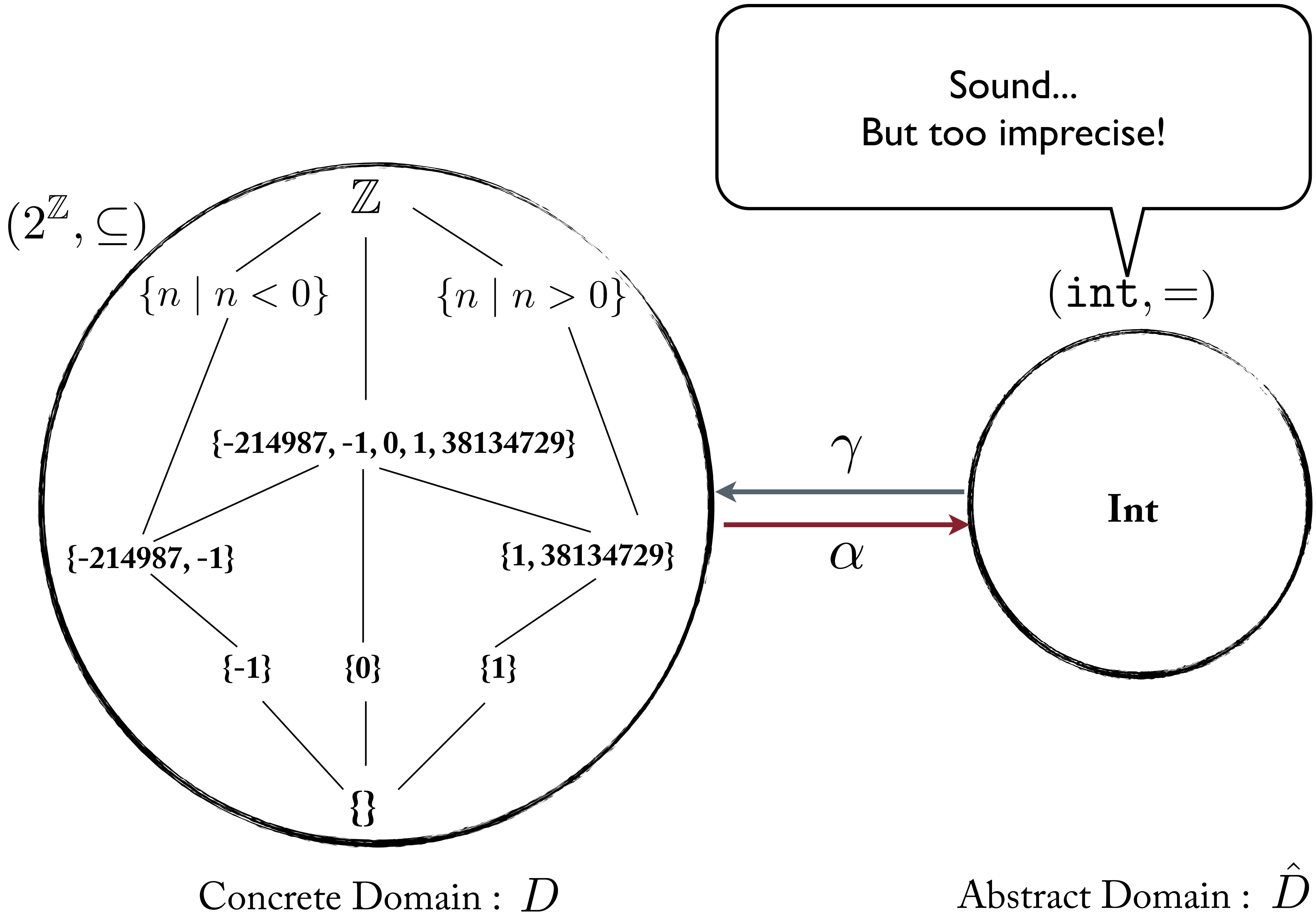
Condition for $\hat{F} : \hat{D} \rightarrow \hat{D}$ to have $\alpha(\text{lfp}F) \sqsubseteq (\text{lfp}\hat{F})$

1) monotone function.

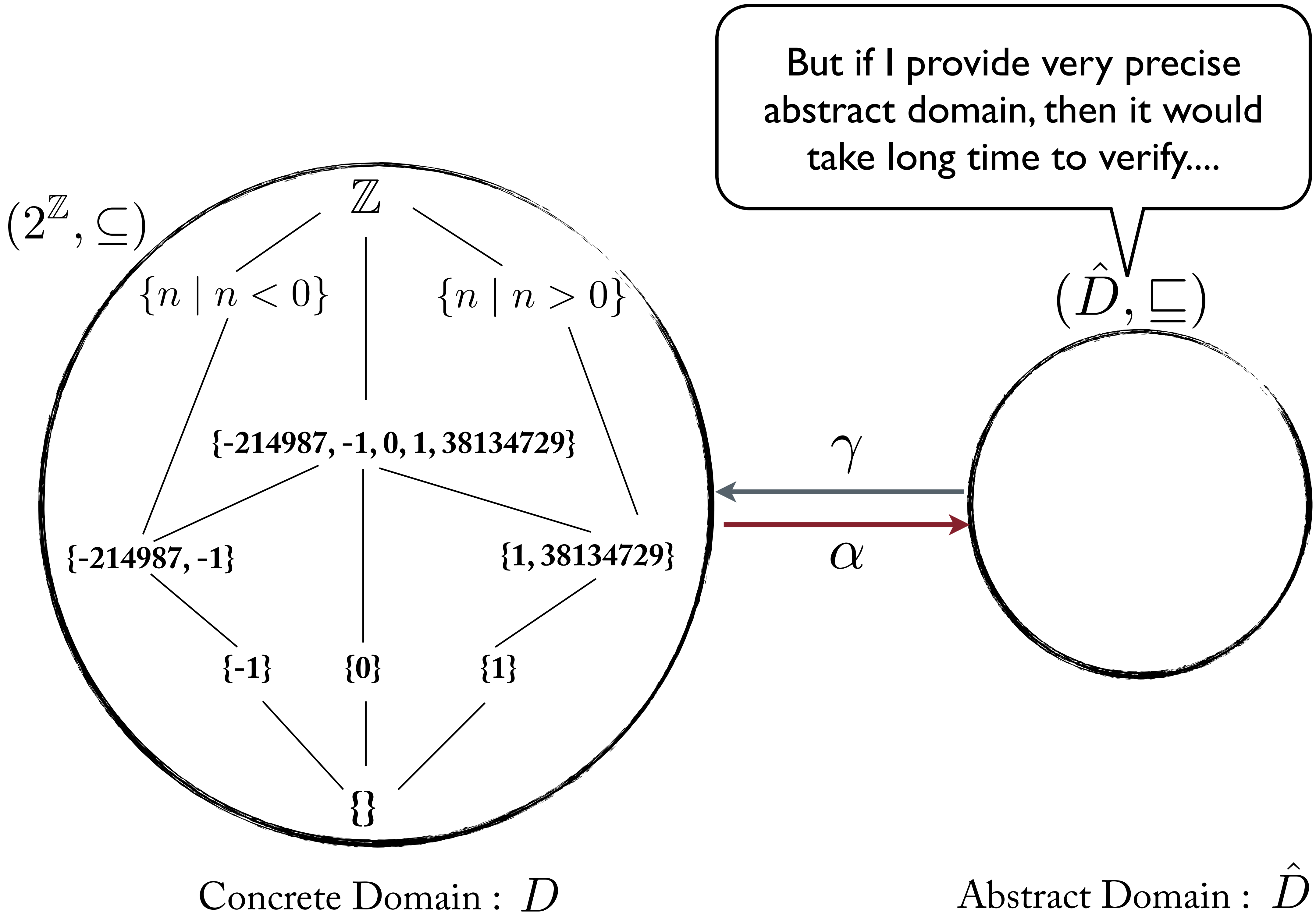
2) $\forall x \in D : \alpha \circ F(x) \sqsubseteq \hat{F} \circ \alpha(x)$



Abstraction, Abstraction, and Abstraction Refinement



Sound...
But too imprecise!
$(2^{\mathbb{Z}}, \subseteq)$
$\mathbb{Z}$
$\{n \mid n < 0\}$
$\{n \mid n > 0\}$
{-214987, -1, 0, 1, 38134729}
{-214987, -1}
{1, 38134729}
{-1}
{0}
{1}
{}
$(\texttt{int}, =)$
γ
α
Int
Concrete Domain : D
Abstract Domain : $\hat{D}$

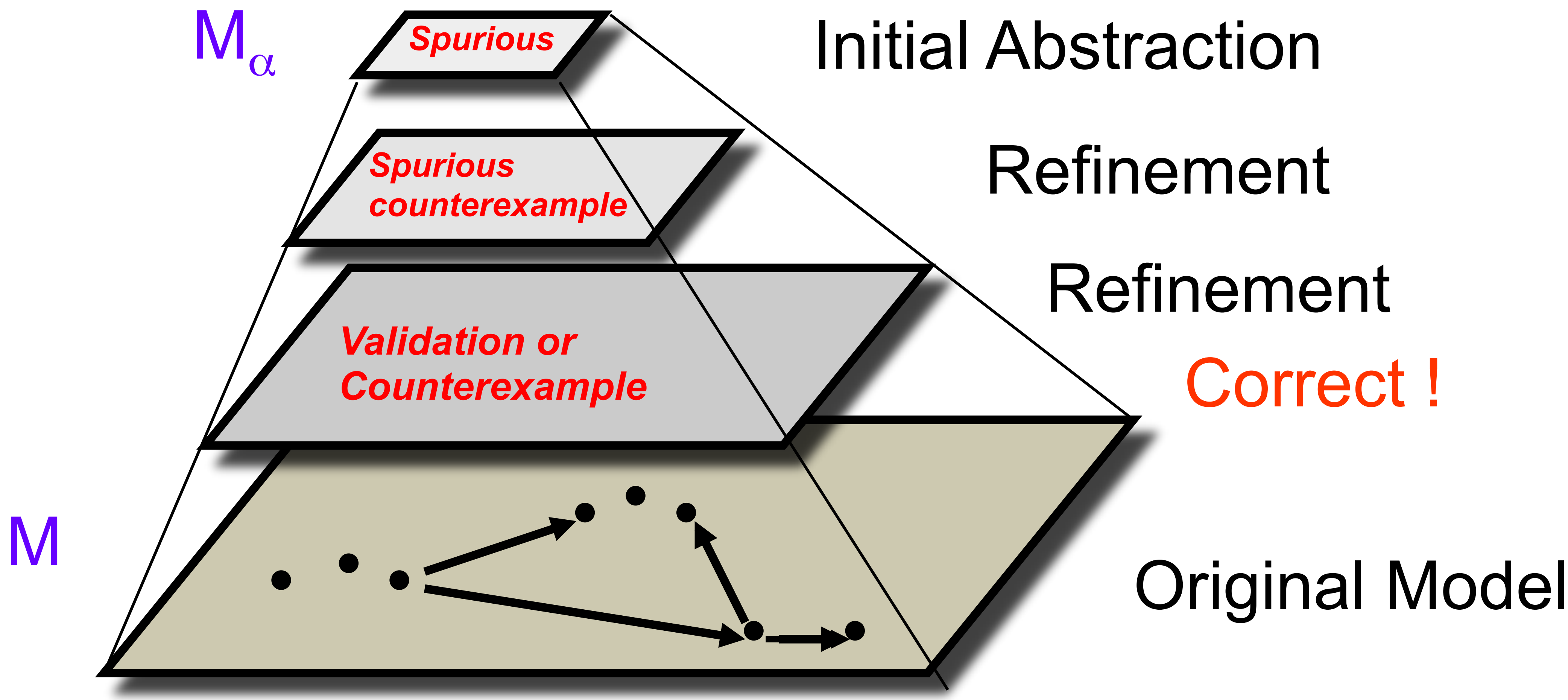


But if I provide very precise abstract domain, then it would take long time to verify....
$(2^{\mathbb{Z}}, \subseteq)$
$\mathbb{Z}$
$\{n \mid n < 0\}$
$\{n \mid n > 0\}$
{-214987, -1, 0, 1, 38134729}
{-214987, -1}
{1, 38134729}
{-1}
{0}
{1}
{}
$(\hat{D}, \sqsubseteq)$
γ
α
Concrete Domain : D
Abstract Domain : $\hat{D}$

"The purpose of **abstraction** is not to be vague, but to create a new semantic level in which one can be absolutely precise."

- Edsger W. Dijkstra

Automatic Abstraction Refinement



Thank you