

三重宇宙論：意圖、計算與物理存在的基本區分

A Theory of Three Universes: The Fundamental Distinction Among Intentional, Computational, and Physical Existence

v0.1 / 2026-07-26 / Neo.K with Aletheia / 初版完成

<https://thisoneisneok.com/html/papers/pu-0-01.html>

SECTION

A Theory of Three Universes: The Fundamental Distinction Among Intentional, Computational, and Physical Existence

論文編號：PU-0-01

系列：「程式宇宙書系」第 0 冊《程式之前》

作者：Neo.K with Aletheia

機構：EveMissLab／一言諾科技有限公司

版本：v0.1

日期：2026 年 7 月 26 日

SECTION

摘要

程式設計、計算機科學與人工智慧通常直接從符號、演算法、資料結構、模型與執行開始，卻很少先回答一個更早的問題：人類為何建立計算機宇宙，而計算究竟連接了哪些存在域？若不先區分「主體或制度想要的世界」、「機器能表示與轉換的世界」以及「實際具有因果效力的世界」，程式就容易被縮減為語法，意圖容易被誤認為可直接執行，計算模型也容易被錯認為現實本身。

本文提出「三重宇宙論」，區分三個互相耦合但不可互相化約的存在域：

$$\begin{aligned} &\mathbb{U} \\ &= \\ &< \\ &\mathcal{I}, \\ &\mathcal{C}, \\ &\mathcal{P} \\ &> \end{aligned}$$

其中：

- $\mathcal{I}$ ：意圖宇宙，指主體、群體、制度或功能性系統所指向、偏好、拒絕、要求或希望實現的可能狀態域；
- $\mathcal{C}$ ：計算機宇宙，指由符號、狀態、地址、記憶、規則、資料、演算法、程序、權限與虛擬時間構成的人工可表示與可轉換存在域；
- $\mathcal{P}$ ：物理宇宙，指具有實際因果效力、受物質、能量、時空、身體、硬體、制度與環境制約的存在域。

本文的第一核心命題是：

$$\begin{aligned} &[\mathcal{I} \\ &\neq \\ &\mathcal{C} \\ &\neq \\ &\mathcal{P}] \end{aligned}$$

三者雖然相互作用，卻不共享相同的存在條件。意圖可以指向尚不存在、甚至物理上不可實現的狀態；計算可以表示與模擬不存在於現實中的世界；物理世界則能以材料、能源、延遲、故障與非預期因果抵抗模型。任何一者都不能被另外兩者完全吸收。

本文進一步指出，計算機宇宙不是與物理世界完全分離的超自然宇宙，而是物理宇宙中被人工穩定化、符號化與規則化的特殊因果區域。其存在必須依賴硬體、能源與物理載體，但其高層規則又具有相對自治性。相同的數位結構可以由不同物理載體承載，而相同物理載體也可以實現不同計算結構。因此，計算機宇宙具有「物理依賴」與「結構相對自治」的雙重性。

本文也慎重處理意圖宇宙的本體地位。意圖宇宙不等於幻想集合，也不能簡單視為物理世界之外的獨立空間。它是可能狀態、價值排序、拒絕條件、規範要求、主體目的、制度目標與功能性指向所形成的實質結構域。它可能由具身主體產生，也可能由群體與制度維持；客體本身未必具有主觀欲求，但可被賦予功能性目的或在系統中具有目標指向。

本文提出三類存在判準：

- 意圖存在判準：某狀態是否被主體或制度指向、偏好、拒絕或要求；
- 計算存在判準：某對象是否能在特定計算體系內以合法身分、狀態與操作存在；
- 物理存在判準：某對象是否在物理因果鏈中具有可觀測或可作用的實際效力。

由此可知，同一對象可以在三個宇宙中以不同方式存在。例如「尚未建成的橋」可以在意圖宇宙中作為目標存在，在計算機宇宙中作為 CAD 模型與工程模擬存在，但尚未在物理宇宙中作為可通行結構存在。

本文建立「跨宇宙同一性問題」：三個宇宙中的對象不應因名稱相同就被視為同一存在。必須經由映射、追蹤、證據與實現關係，才能判斷某意圖對象、計算對象與物理對象是否構成合法對應。本文因此提出：

```
【Name(x_I)
=
Name(x_C)
=
Name(x_P)
not⇒
x_I
=
x_C
=
x_P】
```

本文也分析三種典型化約錯誤：

- 意圖化約：把「想要」當成「已經存在」；
- 計算化約：把「模型中存在」當成「現實中存在」；
- 物理化約：把意圖與規範只視為物理事件，忽略其指向與約束功能。

本文的第二核心命題是：

【Program
≠
Intent
≠
Computation
≠
Physical Effect】

程式位於三者的耦合邊界，但不等於任何單一宇宙。它是把部分意圖形式化為可計算結構，並在授權與條件成立時對數位或物理世界施加狀態轉換的中介結構。

本文最後提出可證偽研究綱領，包括跨宇宙對象追蹤、意圖—模型保真、模型—物理偏差、功能性目的判定、計算載體替換、數位對象持續性、名稱同一性錯誤與制度意圖漂移。本文為後續《三宇宙耦合動力學》建立本體論起點：只有先區分三個宇宙，才能進一步研究它們如何相互轉譯、限制、生成與改變。

關鍵詞：三重宇宙論、意圖宇宙、計算機宇宙、物理宇宙、計算存在論、程式本體論、數位存在、功能性目的、跨宇宙同一性

SECTION

Abstract

Computer science, programming, and artificial intelligence usually begin with symbols, algorithms, data structures, models, and execution. They rarely begin with a prior question: why do humans construct computational universes, and which domains of existence does computation connect?

This paper proposes a theory of three universes:

$\mathbb{U}$
=
<
 $\mathcal{I}$,
 $\mathcal{C}$,
 $\mathcal{P}$
>

where I is the intentional universe, C is the computational universe, and P is the physical universe.

The intentional universe consists of possible states that subjects, collectives, institutions, or functionally directed systems prefer, reject, require, or seek to realize. The computational universe consists of artificial domains of symbols, states, memory, rules, identities, algorithms, permissions, and virtual time. The physical universe consists of entities and processes with actual causal efficacy under material, energetic, temporal, bodily, institutional, and environmental constraints.

The central thesis is:

【 I
 $\neq$
 C
 $\neq$
 P 】

The three universes interact but cannot be reduced to one another. Intent can refer to states that do not yet exist or cannot physically exist. Computation can represent and simulate worlds absent from physical reality. Physical reality can resist models through material limits, energy costs, latency, failure, and unmodeled causality.

The computational universe is characterized by both physical dependence and structural relative autonomy. It requires physical substrates, yet its higher-level rules can remain invariant across different substrates. The intentional universe is not merely a collection of fantasies; it is a structured domain of goals, values, prohibitions, normative requirements, institutional purposes, and functional directions.

The paper introduces three criteria of existence, a theory of cross-universe identity, and three reduction errors: intentional reduction, computational reduction, and physical reduction. It argues that a program is not identical to intent, computation, or physical effect. Rather, it is a mediated structure that formalizes part of an intention into computable form and, under valid authorization and conditions, produces state transitions in digital or physical worlds.

This paper establishes the ontological foundation for the next study, Dynamics of Three-Universe Coupling.

Keywords: three-universe theory, intentional universe, computational universe, physical universe, computational ontology, program ontology, digital existence

在學習程式時，常見起點是變數、條件、迴圈、函式、類別與框架。但這些東西為什麼存在？

人類之所以發明變數，不是因為宇宙天然要求變數；之所以建立資料庫，也不是因為物理世界本身就是表格。這些結構出現，是因為人類想要記錄、比較、預測、控制、重複、協作與改變世界。

因此，在程式碼以前，至少已存在三種不同問題：

- 我們想要什麼？
- 機器能表示與運算什麼？
- 現實實際允許什麼？

若三者被混為一談，就會出現典型錯誤：

- 使用者說「我要安全」，系統卻只實作一個布林欄位；
- 模型在模擬中成功，便被視為真實世界必然成功；
- 程式回傳 success，便被視為物理效果已完成；
- AI 理解一句要求，便被視為已取得執行權；
- 數位世界沒有表示某個人，便把那個人當成不存在。

所以，程式理論的真正起點不是語法，而是存在域的區分。

定義：

```
【 $\mathbb{U}$ 
=
<
mathcal{I},
mathcal{C},
mathcal{P}
>】
```

其中：

- $\mathcal{I}$ ：Intentional Universe，意圖宇宙；
- $\mathcal{C}$ ：Computational Universe，計算機宇宙；

- $\mathcal{P}$: Physical Universe，物理宇宙。

三者具有交互作用，但其存在條件不同。

SECTION

2.1 非等同原則

$\mathcal{I}$
 $\neq$
 $\mathcal{C}$
 $\neq$
 $\mathcal{P}$

意圖存在，不等於計算表示存在；計算表示存在，不等於物理實現存在；物理存在，也不等於已被意圖理解或計算建模。

SECTION

2.2 非封閉原則

任何一個宇宙都不應被當成另外兩者的完整封閉模型：

$\mathcal{M}_C(\mathcal{P})$
 $\subsetneq$
 $\mathcal{P}$

計算機中的物理模型，只能涵蓋物理世界的一部分。同樣：

$\mathcal{M}_C(\mathcal{I})$
 $\subsetneq$
 $\mathcal{I}$

意圖的計算模型，也只保存意圖的一部分。

3.1 定義

物理宇宙是具有實際因果效力，並受到物質、能量、時空、環境、身體、硬體與制度性條件制約的存在域。

形式上：

$$P$$
$$=$$
$$<$$
$$M,$$
$$E,$$
$$T,$$
$$X,$$
$$B,$$
$$K,$$
$$R$$
$$>$$

其中：

- M：物質；
- E：能量；
- T：物理時間；
- X：空間與位置；
- B：身體與載體；
- K：因果關係；
- R：資源與限制。

SECTION

3.2 物理存在判準

對象 x 在物理宇宙中存在，不要求一定被人類直接看見，但至少需位於某種實際因果鏈中，能對其他物理過程造成影響，或受其影響。

SECTION

3.3 計算不脫離物理

每次運算需要晶體管或其他物理元件、能源、時間、記憶載體、訊號傳遞、散熱與維護。因此：

【 C
physically depends on
 P 】

不存在完全沒有物理載體的實際計算機。

SECTION

3.4 物理宇宙的抵抗

現實會以感測誤差、未建模因素、材料疲勞、網路中斷、電力不足、時間延遲、人類拒絕、制度阻礙與偶發事件抵抗模型。

所以：

SimulatedSuccess
not $\Rightarrow$
PhysicalSuccess

SECTION

4.1 定義

計算機宇宙是由符號、狀態、身份、位址、記憶、規則、轉換、權限與虛擬時間構成，並可由物理系統承載的人工存在域。

形式上：

C

=

<

Σ ,

S,

I,

A,

M,

R,

O,

T_C,

P_C

>

其中：

- Σ ：符號；
- S：狀態；
- I：數位身分；
- A：位址；
- M：記憶；
- R：規則；
- O：操作；
- T_C：計算時間；
- P_C：權限。

SECTION

4.2 計算存在判準

對象 x 在某計算體系中存在，至少需具備：

- 可識別身分；
- 合法結構；
- 可儲存或可生成狀態；
- 至少一種可定義操作；
- 位於某一 Runtime 或可執行語意中。

形式上：

$\text{Exists_C}(x)$

$\Leftrightarrow$

$\text{Identifiable}(x)$

$\wedge$

$\text{WellFormed}(x)$

$\wedge$

$\text{StatefulOrGenerable}(x)$

SECTION

4.3 人工法則

數位世界中的使用者、訂單、金幣、角色、權限、刪除、封鎖、等級與任務，都不是物理宇宙天然提供的類別。它們由 schema、程式、資料模型、UI、規則與組織制度共同建立。

因此：

【數位法則是人工穩定化的狀態轉換規則。】

SECTION

4.4 相對自治性

同一演算法可以在不同硬體上實現：

```
Impl_(P⊠)(C)
cong
Impl_(P⊠)(C)
```

若它們保持同一高層計算語意。因此，計算機宇宙同時具有物理依賴與結構相對自治。

SECTION

4.5 相對自治不等於完全獨立

虛擬機、容器與模擬世界可以隱藏硬體差異，但不能消除能源成本、記憶上限、效能差異、故障、裝置能力與網路距離。

SECTION

5.1 定義

意圖宇宙是主體、群體、制度或功能性系統所指向、偏好、拒絕、要求、承諾或希望實現之可能狀態所形成的結構域。

形式上：

```
mathcal I
=
<
G,
N,
V,
C,
D,
U,
```

其中：

- G：目標；
- N：非目標與拒絕狀態；
- V：價值排序；
- C：限制；
- D：決策；
- U：未決定與不可代理部分；
- H：歷史意圖與承諾。

SECTION

5.2 意圖存在判準

Exists_I(x)

⇔

DirectedToward(x)

∨

Preferred(x)

∨

Rejected(x)

∨

Required(x)

即使某狀態尚未在物理世界實現，它仍可具有規劃、選擇與限制行動的效力。

SECTION

5.3 意圖不等於幻想

幻想可以沒有承諾、約束與行動指向。意圖通常至少包含某種選擇、偏好、拒絕、行動傾向、規範效力或未來指向。

Fantasy

not≡

Intent

SECTION

5.4 是否屬於前時空

本文暫不強制把意圖宇宙定義為物理前時空。更保守的表述是：意圖的內容可以指向尚未進入物理時空的狀態，但意圖活動本身仍可能依賴具身主體、制度與物理載體。

因此需區分：

- 意圖內容的超前性；
- 意圖活動的物理承載性。

SECTION

5.5 主體、群體、制度與功能性目標

- 主體意圖：由具有經驗、欲求或決策能力的主體持有。
- 群體意圖：由協議、共同承諾與協調維持。
- 制度意圖：由法律、組織章程或政策設定。
- 功能性目標：客體未必具有主觀欲求，但可被指定應維持、尋找或完成某種狀態。

本文將最後一類稱為功能性指向，而非直接宣稱所有客體都有主觀意識。

存在域 核心判準 典型問題

意圖宇宙 被指向、偏好、拒絕、要求或承諾 我們想要什麼？

計算機宇宙 可合法表示、存續、轉換與執行 機器中什麼算存在？

物理宇宙 位於實際因果鏈並受物理限制 現實中什麼具有實效？

以尚未建成的橋為例：

- 在意圖宇宙中，它是交通改善目標、預算選擇與政策承諾；
- 在計算機宇宙中，它是 CAD 模型、工程模擬、BIM 物件與施工排程；
- 在物理宇宙中，只有真正落成並進入因果使用的結構，才成為可通行的橋。

因此：

```
Bridge_I  
≠  
Bridge_C  
≠  
Bridge_P
```

SECTION

7.1 名稱不保證同一

```
【Name(x_I)  
=  
Name(x_C)  
=  
Name(x_P)  
not⇒  
x_I  
=  
x_C  
=  
x_P】
```

例如系統中的「完成訂單」可能只是資料庫欄位變成 completed，但商品尚未送達、款項未真正入帳、使用者未收到通知。

名稱一致，但物理與制度效果未必一致。

SECTION

7.2 對應關係

需建立：

$\mu_{(IC)}$
:
 $\mathcal{I}$
 $\rightarrow$
 $\mathcal{C}$

表示意圖如何被模型化。

以及：

$\mu_{(CP)}$
:
 $\mathcal{C}$
 $\rightarrow$
 $\mathcal{P}$

表示計算狀態如何產生物理效果。

再建立：

$\mu_{(PI)}$
:
 $\mathcal{P}$
 $\rightarrow$
 $\mathcal{I}$

表示物理結果如何被觀測與重新解釋。

SECTION

7.3 同一性證據

跨宇宙同一性需要：

- 穩定身分；
- 映射規則；
- 版本；
- 來源；
- 觀測；
- 驗證；
- 時間；
- 授權。

沒有這些，僅靠名稱與敘事不足以證明對應。

SECTION

8.1 意圖化約

把「想要」誤認為「已實現」。

例如「我要一個安全系統」並沒有自動定義何為安全，也沒有使安全成為現實。

SECTION

8.2 計算化約

把「模型中存在」誤認為「現實中存在」。

例如數位分身顯示設備正常，不代表設備真的無故障。

SECTION

8.3 物理化約

把意圖、規範、價值與承諾只視為神經或物理事件，忽略其指向、理由與制度效力。

即使意圖依賴物理載體，也不能因此忽略：

- 它指向什麼；
- 它拒絕什麼；
- 它對行動有何規範作用。

SECTION

9.1 意圖對物理的依賴

人類意圖受到身體、生存、感知、資源、時間與社會關係塑造。

SECTION

9.2 意圖的相對超前性

意圖可以指向尚未存在的狀態：

x
 $\in$
 $\mathcal{I}$

while

x
 $\notin$
 $\mathcal{P}$

SECTION

9.3 計算對物理的依賴

沒有物理載體，實際計算無法發生。

SECTION

9.4 計算的結構自治

同一高層程式可跨不同物理載體保持語意。

SECTION

9.5 物理世界不依賴被計算才存在

未被感測、未被建模、未被命名的物理過程，仍可能存在並產生因果效果。

可以從兩種意義回答。

SECTION

10.1 實現論意義

是。

所有實際計算都必須由物理載體實現：

$$\begin{array}{l} \text{mathcal C}_{\text{realized}} \\ \subseteq \\ \text{mathcal P} \end{array}$$

SECTION

10.2 結構論意義

不應簡單等同。

計算結構的身分與規則，不完全由某一個物理材料決定。

一段程式可以：

- 在不同 CPU 執行；

-
- 在模擬器執行；
 - 被重新編譯；
 - 被形式推理；
 - 被人工逐步執行。

因此，計算機宇宙可被理解為：

物理宇宙中的可多重實現結構域。

本文採取分層立場。

SECTION

11.1 承載層

實際意圖活動通常需要主體、記憶、語言、制度、文件、身體或硬體，因此具有物理或社會承載。

SECTION

11.2 內容層

意圖內容可以指向：

- 尚不存在的未來；
- 不可能世界；
- 規範要求；
- 反事實；
- 被拒絕狀態。

所以意圖內容不能被當成當前物理狀態的同義詞。

SECTION

11.3 規範效力層

即使某意圖尚未實現，也可能：

- 約束行動；
- 分配資源；
- 建立法律；
- 觸發程式設計；
- 改變制度。

這使意圖具有實質效力，而不是無效果的幻象。

程式不是單一宇宙中的單純物件。

SECTION

12.1 作為意圖結構

程式設計通常始於：

- 要解決的問題；
- 想要的結果；
- 不允許的狀態；
- 人類或制度目的。

SECTION

12.2 作為計算結構

程式具有：

- 語法；
- 語意；
- 狀態；

- 控制流；
- 資料；
- Runtime 行為。

SECTION

12.3 作為物理過程

實際執行時會：

- 消耗能源；
- 佔用硬體；
- 發送訊號；
- 驅動設備；
- 影響人類與制度。

所以：

【Program
=
Cross-Universe Mediating Structure】

但：

【Program
≠
Intent
≠
Execution
≠
Physical Effect】

本文暫定：

程式是一種把部分意圖壓縮為可計算狀態轉換，並在特定 Runtime、授權、資源與世界條件下產生數位或物理效果的跨宇宙中介結構。

形式化為：

```
mathcal{P}_r
=
<
I',
C_s,
R\Box,
A_u,
W\Box,
\Delta W
>
```

其中：

- I' ：被形式化的部分意圖；
- C_s ：計算結構；
- $R\Box$ ：Runtime；
- A_u ：授權；
- $W\Box$ ：前置世界狀態；
- ΔW ：產生的世界差分。

這個定義將在後續第 1 冊與第 2 冊持續擴張。

SECTION

14.1 意圖宇宙

使用者想要：

- 記住任務；
- 降低遺忘；
- 知道哪些已完成；
- 不讓私人任務被他人看到。

SECTION

14.2 計算機宇宙

系統建立：

- Task；
- User；
- status；
- due_date；
- permission；
- 建立、修改、完成、刪除等操作。

SECTION

14.3 物理宇宙

實際效果可能包括：

- 使用者改變行為；
- 手機發出通知；
- 工作被完成或未完成；
- 伺服器消耗能源；

-
- 私人資訊可能外洩。

SECTION

14.4 非等同

資料庫中的 `completed=true` 不一定代表現實任務真的完成。

這是跨宇宙同一性失敗的典型案例。

SECTION

15.1 意圖宇宙

希望房間維持舒適溫度，並降低能源成本。

SECTION

15.2 計算機宇宙

存在：

- 溫度感測值；
- 設定值；
- 控制規則；
- 開關狀態；
- 時間排程。

SECTION

15.3 物理宇宙

實際存在：

- 空氣溫度；

-
- 房屋隔熱；
 - 感測器誤差；
 - 冷暖氣機；
 - 電力；
 - 人類體感。

SECTION

15.4 落差

感測器顯示 24°C，不等於每個人都感到舒適。

MeasuredState

≠

ExperiencedState

SECTION

16.1 意圖宇宙

使用者希望得到正確答案、寫出文章、完成程式、產生圖像或取得建議。

SECTION

16.2 計算機宇宙

模型處理：

- token；
- embedding；
- probability；

- context ；
- tool call ；
- structured output 。

SECTION

16.3 物理宇宙

結果可能影響工作決策、金錢、健康、法律、人際關係與能源使用。

SECTION

16.4 主要錯誤

語言流暢被誤認為世界正確：

Fluent(x)
not $\Rightarrow$
True_P(x)

SECTION

17.1 數學對象屬於哪個宇宙

數學對象可以：

- 在意圖宇宙中作為研究目標；
- 在計算機宇宙中作為符號與形式物件；
- 透過物理載體被記錄與計算。

本文不在此決定數學柏拉圖主義問題，只指出三宇宙模型可以容納多種數學本體論。

SECTION

17.2 虛構角色是否存在

虛構角色可在意圖宇宙中作為創作對象，在計算機宇宙中作為數位實體，並在物理宇宙中透過書籍、螢幕與人類行為產生效果。

其存在方式不同，不應以單一「存在／不存在」判斷全部層次。

SECTION

17.3 AI 是否具有意圖

目前系統可具有目標函數、任務表示、持續狀態、自我修訂與工具能力。

但是否具有主觀意圖，需要另外處理意識、主體性與持續身份問題。本文只承認其可能具有功能性指向，不預先等同於人類主觀欲求。

- 模型即現實：把模擬、資料庫或數位分身當成物理世界本身。
- 需求即規格：把自然語言要求當成已完整形式化。
- 計算成功即世界成功：工具回傳成功，就視為外部效果完成。
- 名稱同一即存在同一：三個宇宙中的同名對象被錯認為同一。
- 數位不可見即不存在：系統未建模的人、需求與例外被排除。
- 物理限制遺忘：忽略能源、延遲、材料、故障與制度阻力。
- 客體人格化：把功能性目標直接視為主觀欲求。
- 意圖實體化：把尚未決定或只是願望的內容，當成穩定目標。
- 計算機宇宙神秘化：把數位世界說成完全脫離物理世界的獨立宇宙。
- 物理主義的語意消除：承認意圖的物理承載，卻抹除其目的與規範功能。

SECTION

19.1 意圖—模型保真

比較原始意圖 I 與計算模型 $M_C(I)$ 的差異：

$$R_{IC} = d(I, M_C(I))$$

SECTION

19.2 模型—物理偏差

比較計算預測與物理結果：

$$R_{CP} = d(\widehat{P}, P_{\text{observed}})$$

SECTION

19.3 跨宇宙對象追蹤

測量同一專案中的意圖對象、數位對象與物理對象，是否能保持可追溯對應。

SECTION

19.4 名稱同一性錯誤率

統計名稱相同但實際效果不一致的事件。

SECTION

19.5 計算載體替換

將同一計算結構移至不同物理載體，測量語意保持與物理成本差異。

SECTION

19.6 功能性目的判定

研究人類是否能區分：

- 主體欲求；
- 制度目標；
- 設計者賦予目的；
- 系統內部目標函數。

SECTION

19.7 制度意圖漂移

追蹤政策或組織目標在形式化、實作與執行後是否偏離原始目的。

SECTION

19.8 數位存在持續性

研究數位實體跨版本、遷移、複製、分支與 Runtime 後是否仍被視為同一對象。

1.

【mathcal I

≠

mathcal C

≠

- 意圖存在、計算存在與物理存在具有不同判準。
- 計算機宇宙在實現上依賴物理宇宙，在結構上具有相對自治。
- 意圖內容可以超前於當前物理狀態，但意圖活動通常需要具身或制度承載。
- 功能性目標不必等於主觀欲求。
- 同名對象不保證跨宇宙同一。
- 模型中的成功不保證物理世界成功。
- 物理世界中存在的對象，不必先被計算模型表示才具有因果效力。
- 數位實體不是單純資料值，而是具有身份、狀態、規則與歷史的人工存在。
- 意圖不是程式，程式也不是執行結果。
- 程式是跨宇宙中介結構。
- 任何形式化都只捕捉部分意圖。
- 任何感測與資料化都只捕捉部分物理世界。
- 程式語言的限制性與生成性，源自它對計算機宇宙合法狀態的定義。

15.

【若不先區分三個宇宙，】

【程式語言、AI 與數位世界就容易被誤認為現實本身。】

本文只完成三個宇宙的基本區分。

下一篇 PU-0-02 將進一步研究：

I

→

C

→

ΔP

→

並處理：

- 意圖如何形式化；
- 計算如何執行；
- 現實如何回饋；
- 工具如何生成新意圖；
- 制度、權力與資源如何影響耦合；
- 為何三宇宙耦合不是線性管道。

人類建立計算機，不只是為了進行更快的算術。

我們建立了一個人工存在域，在其中：

- 對象可以被命名；
- 狀態可以被保存；
- 規則可以被固定；
- 世界可以被模擬；
- 歷史可以被重放；
- 身分可以被複製；
- 行動可以被自動化。

但這個人工存在域不是現實本身，也不是意圖本身。

它位於兩者之間，並受兩者限制。

意圖宇宙告訴我們希望什麼成為現實。

計算機宇宙告訴我們哪些東西可以被表示、組合、驗證與執行。

物理宇宙則決定哪些效果真正發生、需要何種代價，以及現實會如何抵抗模型。

因此：

【Program
=
Partial Intent
+
Computable Structure
+
Physical Realization Conditions】

程式設計最深層的問題，從來不只是：

這段程式怎麼寫？

而是：

我們想要什麼？

我們表示了什麼？

機器實際執行了什麼？

世界真正發生了什麼？

三者之間還剩下哪些沒有被看見的差異？

本文的最終命題是：

【計算機宇宙之所以重要，】

【不是因為它取代了現實，】

【而是因為它成為意圖與現實之間，】

【最強大、也最需要被理解的人工耦合域。】

```
cross_universe_object:
  object_id: "bridge-project-001"
  intentional:
    goal: "縮短兩地通行時間"
    non_goals:
      - "不得破壞保護區"
    stakeholders:
      - "local_residents"
      - "transport_authority"
    status: "approved-intent"
  computational:
```

```
models:
  - "cad://bridge-v4"
  - "simulation://traffic-v2"
  - "budget://estimate-v5"
status: "design-validated"
physical:
  site: "river-section-a"
  construction_status: "not-started"
  operational_status: false
mappings:
  intent_to_computation:
    verified: true
    residuals:
      - "local cultural impact not fully modeled"
  computation_to_physical:
    verified: false
    reason: "construction not started"
existence_criteria:
  intentional:
    any_of:
      - "directed_toward"
      - "preferred"
      - "rejected"
      - "required"
      - "committed"
  computational:
    all_of:
      - "identifiable"
      - "well_formed"
      - "stateful_or_generable"
      - "operable_under_runtime"
  physical:
    any_of:
      - "causally_effective"
      - "causally_affected"
      - "physically_observable"
      - "physically_inferable"
```

- PU-0-01 三重宇宙論：意圖、計算與物理存在的基本區分
- PU-0-02 三宇宙耦合動力學：從想要、表示到世界改變
- PU-0-03 表示落差：意圖、計算模型與物理現實之間的不可完備映射
- PU-0-04 數位宇宙作為人工存在域：實體、狀態、規則與歷史

-
- PU-0-05 生成與限制：計算機宇宙如何創造並封閉可能性
 - PU-0-06 我們到底想要什麼：計算目的、價值邊界與世界選擇

SECTION

Neo.K／EveMissLab 相關理論

- Neo.K with Aletheia，《意圖程式文明：後文本語言、持續 Agent 與可編譯世界的統一理論》，2026。
- Neo.K with Aletheia，《可編譯世界：從程式執行到世界狀態演化》，2026。
- Neo.K，《計算機宇宙、意圖宇宙與物理宇宙之三重耦合命題》，2026。
- Neo.K，《世界編織論》，2026。
- Neo.K，《萬物算子論》，2026。
- Neo.K，《自然語言可計算論》，2026。
- Neo.K，《意圖遊戲引擎》，2026。

SECTION

一般理論背景

- Turing, A. M., “On Computable Numbers, with an Application to the Entscheidungsproblem,” 1936.
- Church, A., The Calculi of Lambda-Conversion, 1941.
- Newell, A. and Simon, H. A., “Computer Science as Empirical Inquiry: Symbols and Search,” 1976.
- Searle, J. R., Intentionality: An Essay in the Philosophy of Mind, 1983.
- Bratman, M. E., Intention, Plans, and Practical Reason, 1987.
- Floridi, L., The Philosophy of Information, 2011.
- Dennett, D. C., The Intentional Stance, 1987.

-
- Putnam, H., “The Nature of Mental States,” 1967.
 - Lessig, L., Code and Other Laws of Cyberspace, 1999.
 - Gibson, J. J., The Ecological Approach to Visual Perception, 1979.
 - Clark, A., Supersizing the Mind, 2008.

SECTION

v0.1 — 2026-07-26

- 完成十八篇地基論文的第一篇。
- 建立意圖、計算與物理三重宇宙模型。
- 建立三種存在判準。
- 提出計算機宇宙的物理依賴與結構相對自治。
- 區分意圖內容的超前性與意圖活動的承載性。
- 區分主體意圖、群體意圖、制度意圖與功能性目標。
- 建立跨宇宙同一性問題。
- 建立意圖化約、計算化約與物理化約三類錯誤。
- 提出程式作為跨宇宙中介結構的初步定義。
- 加入待辦事項、溫控系统與生成式 AI 三類案例。
- 提出十類失敗模式與八項可證偽研究方向。
- 完成與 PU-0-02 《三宇宙耦合動力學》的銜接。