

意圖中介表示：從自然意圖到多重可執行投影

Intent Intermediate Representation: From Natural Intent to Multiple Executable Projections

v0.1 / 2026-07-27 / Neo.K with Aletheia / 初版完成

<https://thisoneisneok.com/html/papers/pu-2-04.html>

SECTION

Intent Intermediate Representation: From Natural Intent to Multiple Executable Projections

論文編號：PU-2-04

系列：「程式宇宙書系」第 2 冊《程式語言的本質》

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 27 日

SECTION

摘要

前三篇已建立：程式語言不等於文字；符號應被理解為受治理算子；程式語言具有語法、語意與效果三層存在。本篇進一步處理意圖時代最核心的編譯問題：自然語言、圖形、表格、對話與人類未完成想法，如何被轉換為可驗證、可追蹤、可授權、可生成多種投影，並能安全進入 Runtime 的共同中介結構？

本文提出「意圖中介表示」：

【mathcal I_R

=

<

G,

W,

A,

S,

R,

E,

P,

U,

V,

H

>】

其中：

- G：Goal，目的與成功條件；
- W：World，問題世界與作用域；
- A：Actors，主體、角色與受影響者；
- S：State，前置狀態、目標狀態與合法轉移；
- R：Rules，規則、不變量與非目標；
- E：Effects，外部、物理與治理效果；
- P：Permissions，權限、委任與人類批准；
- U：Uncertainty，歧義、未知、爭議與未決選擇；
- V：Verification，證據、測試與完成判準；
- H：History，來源意圖、版本、決策與變更歷史。

本文核心命題是：

【Natural Intent

≠

Executable Program】

自然語言擅長表達目的、背景、例外、價值與尚未完成的思考，但它通常包含：

- 歧義；
- 省略；
- 代詞；
- 隱含常識；
- 不明作用域；
- 未明權限；
- 未決成功條件；
- 不可逆效果缺席；
- 主體與責任位置不清。

因此，從自然意圖直接跳到程式碼或工具執行，會把尚未解決的世界問題偽裝成已完成的編譯結果。

本文提出五階意圖編譯鏈：

【Intent

→

ClarifiedIntent

→

SemanticIR

→

EffectPlan

→

ExecutableProjection】

其中：

- Intent：原始自然意圖；
- ClarifiedIntent：已顯示歧義、作用域與未決選擇的意圖；
- Semantic IR：綁定問題世界身份、狀態、規則與責任；
- Effect Plan：明示外部效果、權限、風險、證據與恢復；
- Executable Projection：程式碼、工作流、UI、契約、測試、部署或 Agent 計畫。

本文將意圖編譯定義為：

```
【Compile_I
=
Parse
+
Bind
+
Clarify
+
Constrain
+
Plan
+
Verify
+
Project】
```

它不是把一句自然語言翻譯為另一種文字，而是把模糊的人類目的轉化為可被多角色共同理解與驗證的世界變更結構。

本文區分四類不確定性：

- 詞彙歧義：同一詞具有多種可能語意；
- 指稱歧義：不知道指向哪個世界實體；
- 規範未決：沒有決定權限、優先級與例外；

- 效果未決：尚未決定世界要改變到哪一層。

本文主張，不確定性不能被編譯器或 AI 靜默猜測後刪除，而應正式保存在 IR：

```
unknown
ambiguous
contested
requires-human-choice
requires-external-evidence
out-of-scope
```

本文提出「不可代理選擇保留」原則：

若某項選擇會永久改變主體權利、身份、財產、公開狀態或物理世界，而該選擇本應由特定主體完成，IR 必須保留該決策節點，不能因自動化便利而將其消除。

形式上：

```
NonDelegableChoice(c)
⇒
PreserveHumanDecisionNode(c)
```

本文進一步建立多重投影：

```
Π
(
  mathcal I_R
)
=
{
  P_(code),
  P_(workflow),
  P_(ui),
  P_(contract),
  P_(test),
  P_(docs),
  P_(runtime),
  P_(audit)
}
```

同一意圖 IR 可以生成：

- 程式碼；
- 狀態機與工作流；
- 使用者介面；
- API 與事件契約；
- 測試與驗證條件；
- 文件與可視圖；
- Runtime 執行計畫；
- 稽核與證據規則。

本文提出「多投影一致性」：

$\forall P_{\Box}, P_{\Box} \in \Pi(\text{mathcal I}_R),$

Consistent

(
 $P_{\Box}, P_{\Box} \text{midmathcal I}_R$
)

不同投影不必外觀相同，但必須保持核心世界身份、狀態、效果、權限、失敗與完成條件。

本文也區分「生成」與「授權」。AI 或編譯器可以生成候選程式，但生成不等於取得執行權：

【Generated

not $\Rightarrow$

Authorized

not $\Rightarrow$

Executed】

IR 必須保留：

- 候選；

- 已驗證；
- 已批准；
- 已排程；
- 已執行；
- 已觀測；
- 已接受；

等不同生命週期。

本文進一步提出「效果預覽」：

```
widehat{\Delta W}
=
PredictEffects
(
  \mathcal{I}_R,
  S_{\Box}
)
```

在執行前，系統應能向人類與 AI 顯示：

- 預期修改哪些狀態；
- 呼叫哪些外部服務；
- 哪些效果不可逆；
- 哪些主體受影響；
- 哪些權限被使用；
- 哪些失敗與補償可能發生；
- 哪些結果仍不可確定。

本文使用網站更新、訂單退款、資料報表、AI Agent 多工具任務、醫療提醒、研究論文發布與物理設備排程等案例，說明自然意圖如何被轉換為結構化 IR，再生成多個相互一致的執行與理解投影。本文最後提出可證偽研究綱領，包括意圖歧義率、未決選擇保留率、自然語言到 IR 的語意保持、效果預覽準確率、多投影一致性、人工確認負擔、AI 直接生成與 IR 生成之事故差異、IR 版本演化及人機共同編譯效率。

本文為下一篇〈可編譯世界〉建立直接地基：當意圖已被轉換為具有世界身份、效果、權限與證據的 IR，程式執行就可被重新理解為對世界狀態差分的受治理提交。

關鍵詞：意圖中介表示、自然語言編譯、多後端生成、效果預覽、不確定性保留、人機共同編譯、AI 程式設計

SECTION

Abstract

This paper introduces Intent Intermediate Representation (Intent IR) as a structured bridge between natural intent and multiple executable projections.

Intent IR preserves goals, problem-world scope, actors, state transitions, rules, effects, permissions, uncertainty, verification, and history.

The paper distinguishes generation from authorization, preserves non-delegable choices, formalizes ambiguity and uncertainty, and supports multiple projections including code, workflows, user interfaces, contracts, tests, documentation, runtime plans, and audit rules.

Keywords: intent intermediate representation, natural-language compilation, multiple projections, effect preview, human-AI co-compilation

使用者可能說：

把網站更新後發布。

這句話具有清楚方向，卻缺少：

- 哪個網站；
- 更新哪些內容；
- 使用哪個版本；
- 發布到哪個環境；

- 誰有權批准；
- 是否先測試；
- 發布成功如何證明；
- 是否允許回滾；
- 外部快取與爬蟲如何處理。

如果 AI 直接從這句話生成並執行命令，它不是完成編譯，而是用猜測填滿未決世界。

本文定義：

【 I_R
=
<
G,
W,
A,
S,
R,
E,
P,
U,
V,
H
>】

SECTION

2.1 Goal G

保存：

- 原始目的；
- 成功條件；

-
- 非目標；
 - 可接受殘差；
 - 停止條件。

SECTION

2.2 World W

保存：

- 問題世界；
- 實體；
- 模組；
- 環境；
- 時間；
- 外部依賴；
- 作用域。

SECTION

2.3 Actors A

保存：

- 委任者；
- 執行者；
- 批准者；
- 受影響者；
- 責任者；

-
- 外部權威。

SECTION

2.4 State S

保存：

- 當前狀態；
- 目標狀態；
- 合法轉移；
- 版本；
- 競爭與爭議。

SECTION

2.5 Rules R

保存：

- 世界規則；
- 系統規則；
- 實作限制；
- 暫時規則；
- 不變量；
- 例外。

SECTION

2.6 Effects E

保存：

- 讀寫；
- 事件；
- 外部呼叫；
- 物理效果；
- 治理效果；
- 可逆性；
- 補償。

SECTION

2.7 Permissions P

保存：

- 能力；
- 作用域；
- 時效；
- 委任；
- 批准；
- 不可代理選擇。

SECTION

2.8 Uncertainty U

保存：

- 歧義；

-
- 未知；
 - 爭議；
 - 未決；
 - 假設；
 - 外部待確認。

SECTION

2.9 Verification V

保存：

- 測試；
- 不變量；
- 證據；
- 完成；
- 外部確認；
- 人類接受。

SECTION

2.10 History H

保存：

- 原始意圖；
- 澄清；
- 選擇；
- 拒絕；

- 版本；
- 生成與執行歷史。

本文提出：

```
【Intent
→
ClarifiedIntent
→
SemanticIR
→
EffectPlan
→
ExecutableProjection】
```

SECTION

3.1 Intent

原始意圖可能是：

- 一句話；
- 一段對話；
- 一張草圖；
- 一個表格；
- 一組操作示範；
- 一份舊系統範例。

SECTION

3.2 Clarified Intent

把原始內容展開為：

- 已知；
- 未知；
- 歧義；
- 假設；
- 必須由人選擇之處；
- 可以由系統推導之處。

SECTION

3.3 Semantic IR

將意圖綁定到：

- 問題世界實體；
- 狀態；
- 規則；
- 責任模組；
- 契約；
- 身份與時間。

SECTION

3.4 Effect Plan

建立：

- 預期世界差分；
- 外部效果；

-
- 不可逆節點；
 - 權限；
 - 失敗；
 - 補償；
 - 證據。

SECTION

3.5 Executable Projection

生成：

- 程式碼；
- 工作流；
- UI；
- API；
- 測試；
- 部署設定；
- Agent 工具計畫。

本文定義：

【Compile_I

=

Parse

+

Bind

+

Clarify

+

Constrain

SECTION

4.1 Parse

辨識自然語言、圖形或操作示範中的候選結構。

SECTION

4.2 Bind

把名稱、代詞與描述綁定到世界身份。

SECTION

4.3 Clarify

保留並處理歧義、未知與未決。

SECTION

4.4 Constrain

加入：

- 型別；
- 狀態；
- 規則；
- 權限；
- 不變量；
- 非目標。

SECTION

4.5 Plan

形成可執行步驟、依賴、失敗與恢復路徑。

SECTION

4.6 Verify

檢查語意、效果、授權與完成條件。

SECTION

4.7 Project

生成多種人類與機器投影。

SECTION

5.1 詞彙歧義

例如「刪除」可能表示：

- 從畫面隱藏；
- 軟刪除；
- 匿名化；
- 停用；
- 永久清除。

SECTION

5.2 指稱歧義

例如「把那份文件發布」中的「那份文件」可能有多個候選。

SECTION

5.3 規範未決

例如：

- 誰能批准；
- 哪項規則優先；
- 是否允許例外；
- 哪些資料可以公開。

SECTION

5.4 效果未決

例如「更新網站」沒有說明：

- 只更新草稿；
- 建立預覽；
- 發布測試環境；
- 切換正式流量。

SECTION

5.5 不確定性不是錯誤

未決狀態應被正式保存，而不是被編譯器或 AI 靜默填補。

本文提出：

unknown
ambiguous
contested
requires-human-choice
requires-external-evidence
out-of-scope

SECTION

6.1 Unknown

目前沒有足夠資料。

SECTION

6.2 Ambiguous

存在多個合理解釋。

SECTION

6.3 Contested

不同主體對事實或規則有衝突主張。

SECTION

6.4 Requires Human Choice

該選擇屬於特定主體，不能由系統代替。

SECTION

6.5 Requires External Evidence

需要外部系統、感測或專業判定。

SECTION

6.6 Out of Scope

明確不在本次意圖與授權範圍內。

本文提出：

NonDelegableChoice(c)

⇒

PreserveHumanDecisionNode(c)

SECTION

7.1 不可代理選擇

包括可能永久改變：

- 身份；
- 財產；
- 權利；
- 公開狀態；
- 醫療決定；
- 物理世界；
- 長期承諾；

的選擇。

SECTION

7.2 保留方式

IR 應包含：

decision:

type: "human-required"

actor: "account-owner"

options:

- "deactivate"

- "anonymize"

- "permanently-delete"

reversible: false

SECTION

7.3 自動化不能消除主體

AI 可以整理資訊、模擬後果與提出建議，但不能因「已能推測」而取得本應由主體完成的不可撤回選擇。

意圖轉換必然依賴假設。

SECTION

8.1 假設格式

assumption:

statement: "the provided document is the latest approved version"

source: "user reference"

confidence: "medium"

validation: "check release registry"

failure_effect: "publishing wrong version"

SECTION

8.2 假設生命週期

proposed

validated

rejected

expired

superseded

SECTION

8.3 假設與事實分離

AI 推論、使用者主張、外部證據與權威狀態必須被區分。

SECTION

8.4 假設失效

若關鍵假設失效，IR 應：

- 阻止生成；
- 降級為草稿；
- 要求外部查證；

-
- 重新規劃。

SECTION

9.1 原始來源不能消失

IR 應保留：

- 原句；
- 對話上下文；
- 圖形區域；
- 表格欄位；
- 示範步驟；
- 使用者修改。

SECTION

9.2 節點追蹤

每個語意節點應能回答：

它來自哪一句？

是直接表達、推導還是猜測？

誰確認？

何時修改？

SECTION

9.3 來源狀態

explicit

derived

inferred

assumed

human-confirmed

externally-verified

SECTION

9.4 反向說明

執行計畫與生成程式應能反向說明其與原始意圖的關係。

意圖 IR 不應只保存句子，而應保存可識別節點。

SECTION

10.1 目的節點

描述：

- 希望達成什麼；
- 為誰達成；
- 如何判斷成功；
- 何時停止。

SECTION

10.2 世界實體節點

綁定：

- 身份；
- 類型；
- 所有者；
- 生命週期；
- 權威來源。

SECTION

10.3 行動節點

保存：

- 行動者；
- 目標；
- 前置條件；
- 效果；
- 失敗；
- 證據。

SECTION

10.4 規則節點

保存：

- 規則來源；
- 適用域；
- 優先級；
- 例外；
- 版本；
- 修改權。

SECTION

10.5 決策節點

保存：

- 誰決定；
- 候選選項；
- 後果；
- 可撤回性；
- 決策期限。

SECTION

10.6 證據節點

保存：

- 所證明主張；
- 來源；
- 時間；
- 完整性；
- 信任程度；
- 有效期限。

SECTION

11.1 IR 自身也是持續實體

意圖中介表示具有生命週期：

captured
clarifying
modeled
validated
awaiting-approval
authorized

projected
executing
observing
accepted
revised
withdrawn

SECTION

11.2 生成不等於批准

【Generated
not⇒
Authorized
not⇒
Executed】

SECTION

11.3 執行不等於接受

執行完成後仍需：

- 效果觀測；
- 證據核查；
- 使用者或領域主體接受；
- 殘差處理。

SECTION

11.4 撤回

意圖可以在不可逆效果發生前被撤回。

若已部分執行，撤回需轉化為補償與結案流程。

意圖 IR 必須在生成低階程式前建立效果計畫。

mathcal E_P

=

<

Reads,

Writes,

Events,

Calls,

Physical,

Governance,

Failure,

Recovery,

Evidence

>

SECTION

12.1 讀取

列出讀取哪些權威狀態與敏感資料。

SECTION

12.2 寫入

列出修改哪些狀態，由誰擁有。

SECTION

12.3 事件

列出將形成哪些領域事件。

SECTION

12.4 外部呼叫

列出第三方、網路、費用與相依。

SECTION

12.5 物理效果

列出可能作用於設備、位置、能源與人員的效果。

SECTION

12.6 治理效果

列出權限、身份、規則與責任改變。

SECTION

12.7 失敗與恢復

列出：

- 拒絕；
- 暫時失敗；
- 結果未知；
- 部分成功；
- 重試；
- 補償；
- 人類接管。

SECTION

12.8 證據

每一重要效果都應有完成與失敗證據。

本文提出：

```
widehat{\Delta W}
=
PredictEffects
(
  \mathcal{I}_R,
  S_{\Box}
)
```

SECTION

13.1 預期狀態差分

顯示：

- 哪些實體被建立；
- 哪些狀態被修改；
- 哪些關係被終止；
- 哪些規則生效。

SECTION

13.2 受影響主體

顯示：

- 直接使用者；

-
- 被提及者；
 - 管理者；
 - 外部合作方；
 - 未登入但受影響的人。

SECTION

13.3 風險

顯示：

- 不可逆；
- 敏感資料；
- 費用；
- 對外公開；
- 物理危險；
- 權限提升。

SECTION

13.4 恢復能力

顯示：

- 可重試；
- 可回滾；
- 可補償；
- 只能說明；
- 完全不可逆。

SECTION

13.5 預覽不是保證

預覽是模型推導，仍應標記：

- 高信心；
- 條件性；
- 未知；
- 需外部證據。

本文定義：

```
Π  
(  
  mathcal I_R  
)  
=  
{  
  P_(code),  
  P_(workflow),  
  P_(ui),  
  P_(contract),  
  P_(test),  
  P_(docs),  
  P_(runtime),  
  P_(audit)  
}
```

SECTION

14.1 程式碼投影

生成具體語言來源碼與設定。

SECTION

14.2 工作流投影

生成狀態、事件、等待、人工節點與恢復圖。

SECTION

14.3 UI 投影

生成：

- 可用操作；
- 表單；
- 狀態；
- 確認；
- 撤回；
- 錯誤與部分成功敘事。

SECTION

14.4 契約投影

生成：

- API；
- Command；
- Event；
- Query；

-
- Evidence ；
 - 版本與失敗語意。

SECTION

14.5 測試投影

生成：

- 案例 ；
- 不變量 ；
- 權限測試 ；
- 效果模擬 ；
- 外部完成判準。

SECTION

14.6 文件投影

生成對不同角色的說明與可視圖。

SECTION

14.7 Runtime 投影

生成執行計畫、資源、權限、觀測與恢復設定。

SECTION

14.8 稽核投影

生成證據保留、責任鏈與操作歷史要求。

本文提出：

$\forall P \Box, P \Box \in \Pi(\text{mathcal I}_R),$

Consistent

(

$P \Box, P \Box \mid \text{mathcal I}_R$

)

SECTION

15.1 一致不等於相同

UI 不必顯示程式碼細節，程式碼也不必包含所有文件文字。

SECTION

15.2 必須保持的核心

- 世界身份；
- 狀態語意；
- 規則；
- 效果；
- 權限；
- 失敗；
- 完成條件；
- 版本。

SECTION

15.3 常見漂移

- UI 顯示完成，後端只完成入列；
- 契約允許重試，效果實際不可冪等；
- 文件說需批准，Runtime 可直接執行；
- 測試只驗證回傳值，未驗證外部效果。

SECTION

15.4 一致性檢查

可由 IR 對每個投影建立語意差分與缺失警告。

SECTION

16.1 共同權威

意圖 IR 或其驗證後語意核心應成為多投影共同來源。

SECTION

16.2 手改生成物

直接修改生成碼或 UI，若未回寫語意核心，會造成分裂。

SECTION

16.3 允許局部手工內容

格式、效能與平台細節可以在投影層補充，但必須標示：

- 非語意；
- 後端特定；
- 不可反向覆蓋核心；
- 需在再生成時保存。

SECTION

16.4 投影回饋

低階發現的限制、效果與 Runtime 差異，應提升回 IR，而非長期藏於生成碼。

SECTION

17.1 人類提供

人類較擅長提供：

- 目的；
- 價值；
- 背景；
- 例外；
- 可接受風險；
- 不可代理選擇。

SECTION

17.2 AI 提供

AI 較適合：

- 展開候選結構；
- 找出歧義；
- 對照既有規則；
- 生成投影；
- 模擬效果；

-
- 建立測試與文件。

SECTION

17.3 編譯器提供

編譯器與驗證器負責：

- 結構合法；
- 型別；
- 身份；
- 狀態；
- 效果；
- 權限；
- 投影一致性。

SECTION

17.4 Runtime 提供

Runtime 提供：

- 真實執行；
- 狀態轉移；
- 外部回執；
- 可觀測資料；
- 恢復與補償。

SECTION

17.5 共同鏈

HumanIntent

+

AIStructuring

+

CompilerVerification

+

RuntimeEvidence

形成可持續的人機共同編譯流程。

SECTION

18.1 高信心也不是授權

AI 可以判斷某解釋最可能，但最可能不等於使用者已選擇。

SECTION

18.2 可自動消歧的情況

只有當：

- 選擇可逆；
- 風險低；
- 有明示預設；
- 不改變他人權利；
- 可在執行前預覽；

時，才適合自動選用預設。

SECTION

18.3 必須提出選擇的情況

涉及：

- 永久刪除；
- 對外發布；
- 付款；
- 身份；
- 權限；
- 醫療；
- 法律承諾；
- 物理控制；

時，歧義必須保留到合法決策者完成選擇。

SECTION

18.4 猜測標記

AI 推導應標示：

high-confidence-inference

low-confidence-inference

default-assumption

requires-confirmation

SECTION

19.1 意圖者不必是執行者

使用者可以提出目的，Agent 執行，外部服務完成效果。

SECTION

19.2 權限鏈

Principal

→

Delegation

→

Agent

→

Tool

→

World

每一層必須保存：

- 作用域；
- 時效；
- 可再委任性；
- 禁止效果；
- 證據；
- 撤銷。

SECTION

19.3 能力不等於委任

Agent 有技術能力呼叫某工具，不代表本次意圖授權它使用。

SECTION

19.4 投影權限

不同投影也需要最小權限：

- 文件不應洩漏敏感資料；

-
- UI 不應顯示無權操作；
 - 測試資料不應攜帶真實秘密；
 - Runtime 計畫不應超出委任範圍。

SECTION

20.1 結構驗證

十元欄位、節點與關係是否完整。

SECTION

20.2 指稱驗證

實體、規則、版本與外部服務是否正確綁定。

SECTION

20.3 狀態驗證

當前狀態是否允許計畫中的轉移。

SECTION

20.4 規則驗證

不變量、非目標與例外是否一致。

SECTION

20.5 權限驗證

委任、批准與不可代理選擇是否完整。

SECTION

20.6 效果驗證

世界差分是否有界、可觀測、可恢復或已接受。

SECTION

20.7 投影驗證

各投影是否保持 IR 語意。

SECTION

20.8 Runtime 驗證

執行環境、工具版本、資源與外部依賴是否符合計畫。

SECTION

21.1 意圖會改變

使用者可能：

- 補充條件；
- 改變目的；
- 撤回；
- 降低範圍；
- 接受新的殘差。

SECTION

21.2 版本鏈

$I_R^{(1)}$

→

$I_R^{(2)}$

→

...

→

$I_R^{(n)}$

SECTION

21.3 語意差分

IR 差分應顯示：

- 目的改變；
- 作用域擴張；
- 新增受影響者；
- 新增不可逆效果；
- 權限提升；
- 完成條件改變。

SECTION

21.4 已執行版本

執行後不能用新版本靜默覆寫舊版本，必須知道哪個 IR 版本實際造成世界效果。

SECTION

21.5 分支

多個候選方案可形成分支，經比較後選擇，而不必過早刪除其他可能性。

SECTION

22.1 對話記憶不足

長對話可能包含重要決定，但難以直接作為執行依據。

SECTION

22.2 結構記憶

IR 將對話中的：

- 目的；
- 選擇；
- 限制；
- 未決；
- 批准；

轉化為可查詢節點。

SECTION

22.3 記憶來源

每個記憶應區分：

- 使用者明示；
- 系統觀測；
- AI 推論；
- 外部文件；
- 已過期內容。

SECTION

22.4 遺忘與撤回

若主體撤回資訊或授權，IR 與投影必須能：

- 標記失效；
- 撤銷執行權；
- 重新生成；
- 保存必要稽核而不繼續使用內容。

原始意圖：

把網站更新後發布。

SECTION

23.1 澄清後意圖

goal:

site: "evemisslab.com"

content_source: "approved-release-2026-07"

target_environment: "production"

success:

- "public homepage shows approved version"

SECTION

23.2 未決節點

- 是否先建立預覽；
- 是否允許自動切流量；
- 失敗時是否自動回滾；
- 外部 CDN 完成如何確認。

SECTION

23.3 效果計畫

read release

→ build

→ test

→ deploy

→ verify

→ shift traffic

→ observe public journey

SECTION

23.4 多投影

生成：

- 部署程式；
- 工作流；
- 批准 UI；
- 發布契約；
- smoke test；
- 回滾計畫；
- 發布紀錄。

原始意圖：

把這筆訂單退掉。

SECTION

24.1 詞彙歧義

「退掉」可能是：

- 取消未付款訂單；

-
- 取消履約；
 - 退款；
 - 退貨；
 - 關閉爭議。

SECTION

24.2 指稱

必須綁定訂單、付款、商品與申請者。

SECTION

24.3 規則

需要：

- 退款上限；
- 期限；
- 退貨條件；
- 部分退款；
- 手續費；
- 人工批准。

SECTION

24.4 效果

退款可能不可即時撤回，並涉及財務、庫存、通知與會計證據。

SECTION

24.5 IR 價值

若未完成上述展開，直接呼叫 refund(order) 是把業務歧義壓入不可逆效果。

原始意圖：

幫我整理本月銷售報表。

SECTION

25.1 問題世界

需定義：

- 本月的時區；
- 銷售是下單、付款或結算；
- 退貨與退款如何計算；
- 幣別；
- 稅；
- 資料新鮮度。

SECTION

25.2 權限

報表可能包含：

- 客戶資料；
- 財務資料；
- 員工績效；
- 商業機密。

SECTION

25.3 投影

同一 IR 可生成：

- SQL ；
- 資料管線 ；
- 試算表 ；
- 圖表 ；
- 管理摘要 ；
- 資料來源說明 。

SECTION

25.4 驗證

報表生成成功不等於定義正確，仍需語意與資料品質核查。

原始意圖：

整理研究資料，更新論文，然後發布。

SECTION

26.1 任務分解

- 搜尋與讀取資料 ；
- 判斷證據 ；
- 修改論文 ；
- 執行格式與連結檢查 ；
- 發布網站 。

SECTION

26.2 權限分層

讀取資料、修改草稿與正式發布應使用不同能力與批准。

SECTION

26.3 決策保留

AI 可以提出論文修改，但關鍵斷言與正式發布仍可保留人類批准。

SECTION

26.4 證據鏈

每項新主張應連到來源、分析與修改節點。

SECTION

26.5 恢復

發布失敗不應回滾論文內容；兩者是不同世界責任。

原始意圖：

提醒病人記得吃藥。

SECTION

27.1 世界綁定

需知道：

- 哪位病人；
- 哪種藥；
- 醫囑版本；

-
- 時間與時區；
 - 是否已停藥；
 - 聯絡方式。

SECTION

27.2 主體權利

病人可能選擇：

- 提醒頻率；
- 通知管道；
- 是否向家屬顯示；
- 暫停或撤回。

SECTION

27.3 效果

提醒送出不等於病人已讀，更不等於已服藥。

SECTION

27.4 完成層級

scheduled
provider-accepted
delivered
read
patient-confirmed
必須分開表示。

原始意圖：

把這篇研究發布出去。

SECTION

28.1 前置狀態

- 文件版本；
- 作者；
- 引用；
- 證據狀態；
- 公開授權；
- 敏感內容；
- 發布平台。

SECTION

28.2 非目標

發布不等於：

- 命題已被證明；
- 同行評審完成；
- 研究已獲共識。

SECTION

28.3 投影

可生成：

- Markdown；
- HTML；

-
- PDF ；
 - metadata ；
 - citation ；
 - 網站索引 ；
 - AI 可讀層 。

SECTION

28.4 歷史

IR 應保存發布版本與當時主張強度 。

原始意圖 ：

明天早上讓機器開始運作 。

SECTION

29.1 歧義

- 明天是哪個日期 ；
- 早上幾點 ；
- 哪台機器 ；
- 執行多久 ；
- 誰在現場 。

SECTION

29.2 安全條件

- 維護狀態 ；
- 安全區域 ；

- 感測器；
- 能源；
- 緊急停止；
- 操作者批准。

SECTION

29.3 時間 IR

需保存絕對時間、時區、過期策略與重新確認條件。

SECTION

29.4 物理效果

排程建立、命令送出、設備啟動與安全運行是不同完成層級。

- 自然意圖即程式：一句話直接取得世界修改權。
- 翻譯即編譯：只把自然語言換成程式語言文字。
- 歧義靜默消失：AI 以猜測代替合法選擇。
- 指稱未綁定：「它」「那份」「網站」沒有穩定身份。
- 成功條件缺席：不知道何時算完成。
- 非目標缺席：系統擴張到未委任範圍。
- 不可代理選擇被自動化：永久權利與身份選擇由 Agent 代做。
- 假設冒充事實：AI 推論未標記來源與信心。
- 來源意圖遺失：無法說明生成節點來自哪句要求。
- 生成即批准：候選程式被直接執行。
- 批准即完成：取得批准後忽略 Runtime 與外部證據。

- 投影各自為政：程式碼、UI、測試與文件語意分裂。
- 生成物手改：低階修改未回饋共同 IR。
- 效果計畫缺席：只生成步驟，不列不可逆、權限與補償。
- 預覽冒充保證：模型預測被當成真實效果。
- 作用域膨脹：Agent 在執行過程自行擴大資料與工具範圍。
- 權限繼承過度：子任務獲得整體任務全部能力。
- 對話即記憶：關鍵決定只埋在長上下文中。
- 版本覆寫歷史：新意圖靜默改寫已執行版本。
- 撤回無效：使用者撤回後既有計畫仍保有執行權。
- 部分成功不可見：多投影與多工具工作只顯示成功或失敗。
- 物理排程缺少再確認：時間到便執行，忽略世界狀態已改變。

SECTION

31.1 原始意圖歧義率

統計自然意圖進入 IR 時，包含詞彙、指稱、規範與效果歧義的比例。

SECTION

31.2 未決選擇保留率

測量系統是否把需要人類完成的不可代理選擇保存為正式決策節點，而非靜默填補。

SECTION

31.3 自然語言—IR 語意保持

由人類評估原始目的、非目標、角色與成功條件在 IR 中的保存程度。

SECTION

31.4 指稱綁定準確率

統計代詞、簡稱與描述被正確綁定到世界身份、版本與環境的比例。

SECTION

31.5 假設透明度

檢查 AI 推導中有多少正確標記為：

- inferred ；
- assumed ；
- unverified ；
- human-confirmed ；
- externally-verified 。

SECTION

31.6 效果預覽準確率

比較：

$\widehat{\Delta W}$

與實際觀測：

ΔW

的狀態、事件、外部效果、受影響主體與殘差差異。

SECTION

31.7 多投影一致性

統計程式碼、UI、契約、測試、文件與 Runtime 計畫間的核心語意漂移。

SECTION

31.8 人工確認負擔

研究 IR 是否能把人類確認集中在高風險與真正未決節點，而不是對每個低階步驟要求確認。

SECTION

31.9 直接生成—IR 生成事故差異

比較 AI 直接生成並執行程式，與先建立、驗證、批准 IR 再生成投影的事故率、越權率與恢復成本。

SECTION

31.10 授權作用域保持

檢查委任從主體到 Agent、工具與子任務時，是否出現權限擴張或時效遺失。

SECTION

31.11 IR 版本可追蹤性

測量能否從每個世界效果反查實際執行的 IR 版本、來源意圖、批准與生成物。

SECTION

31.12 撤回有效率

統計主體撤回意圖或權限後，尚未執行投影能否被可靠停止與重新生成。

SECTION

31.13 人機共同編譯效率

比較純人工規格、純 AI 生成及人機共同 IR 流程，在需求理解、開發時間、驗證與維護上的差異。

SECTION

31.14 跨後端語意保持

將同一 IR 生成至不同語言、工作流引擎與 Runtime，測量狀態、效果、權限、失敗與證據保持。

SECTION

31.15 物理效果安全性

比較自然語言直接排程設備與使用含安全條件、感測證據、過期和再確認的 IR 控制方式。

SECTION

31.16 長期維護效果

研究結構化意圖歷史是否降低規則變更、系統重構、AI 模型替換與團隊交接成本。

- 自然意圖不等於可執行程式。
- 意圖編譯不是文字翻譯，而是世界綁定、消歧、限制、規劃、驗證與投影。
- 意圖 IR 必須保存目的、世界、主體、狀態、規則、效果、權限、不確定性、驗證與歷史。
- 原始意圖中的歧義、未知與爭議不能被靜默刪除。
- 詞彙、指稱、規範與效果不確定性具有不同結構。
- 不可代理選擇必須保留為由合法主體完成的決策節點。
- AI 的高信心推測不等於主體授權。
- 假設、推論、使用者主張與外部證據必須區分。

- 每個 IR 節點都應能追蹤回原始意圖與確認歷史。
- IR 自身具有從捕捉、澄清、驗證、批准到執行與接受的生命週期。

11.

Generated

not⇒

Authorized

not⇒

Executed

- 意圖 IR 必須在生成低階程式前建立效果計畫。
- 效果預覽應顯示狀態差分、外部效果、受影響主體、權限、失敗與恢復。
- 效果預覽是條件性推導，不是真實世界保證。
- 同一 IR 可以生成程式碼、工作流、UI、契約、測試、文件、Runtime 與稽核投影。
- 多重投影外觀可以不同，但核心世界語意必須一致。
- 投影不應各自成為可獨立修改的競爭真相。
- 低階 Runtime 發現的限制與未宣告效果應回饋至 IR。
- 人類、AI、編譯器與 Runtime 在意圖編譯中具有不同責任。
- 權限必須沿主體、委任、Agent、工具與世界效果鏈保持作用域與時效。
- 對話內容只有被結構化後，才能成為可靠執行記憶。
- IR 版本必須保存實際造成世界效果的來源與批准。
- 意圖撤回應能撤銷尚未執行投影的執行權。

24.

【意圖時代的程式設計，】

【不是讓一句話跳過所有結構直接改變世界，】

【而是讓一句話能被展開為】

【可理解、可選擇、可驗證、可授權與可追蹤的世界計畫。】

SECTION

33.1 承接 PU-2-01

PU-2-01 指出文字只是程式的一種投影。

本篇建立共同意圖 IR，使文字、視覺、UI、工作流與程式碼能由同一語意核心生成。

SECTION

33.2 承接 PU-2-02

PU-2-02 將符號提升為受治理算子。

本篇將自然意圖分解為具有身份、條件、效果、權限與證據的算子節點與算子圖。

SECTION

33.3 承接 PU-2-03

PU-2-03 建立語法、語意與效果三層。

本篇把意圖依序轉換為語意 IR 與效果計畫，再生成具體語法與 Runtime 投影。

SECTION

33.4 銜接 PU-2-05

下一篇將建立：

```

【CompileWorld
:
(W $\boxtimes$ ,mathcal I_R)
→
(
W_(t+1),
Evidence,
Residual
)】

```

集中處理：

- 世界狀態如何成為編譯輸入；
- 程式執行如何成為世界差分提交；
- 數位效果與物理效果如何分層確認；
- Runtime 如何維持交易、補償與觀測；
- 可編譯世界如何拒絕非法狀態差分。

自然語言使程式設計的入口大幅擴張。

人類不必先知道：

- 語法；
- API；
- 資料表；
- 框架；
- Runtime；
- 部署工具；

就能先描述自己希望世界發生什麼。

這是意圖時代真正重要的進步。

但自然語言的便利，也帶來另一種危險：系統可能把理解、選擇、授權與世界責任全部隱藏在一次不可見的模型推論中。

一句「幫我處理好」可以被展開為數十個狀態變更、外部呼叫、權限使用與不可逆效果。若這些結構不被外部化，使用者只能在執行前盲目信任，在執行後觀看結果。

本文提出意圖 IR，就是為了打開這個黑箱。

它不要求使用者閱讀所有低階程式碼，而是把真正重要的結構顯示出來：

你想達成什麼？

系統認為你指的是哪些對象？

哪些內容仍有歧義？

哪些假設尚未確認？

哪些主體會受到影響？

將產生哪些外部或不可逆效果？

使用哪些權限？

失敗時如何恢復？

什麼證據代表完成？

哪個選擇必須由你自己完成？

本文將意圖編譯收束為：

【Intent Compilation

=

Meaning Preservation

+

Uncertainty Preservation

+

Effect Planning

+

Authorization

+

Multi-Projection Generation

+

Evidence】

而意圖 IR 的角色是：

【Intent IR

=

Shared Semantic Contract】

它位於人類意圖、AI 推理、編譯器驗證、Runtime 執行與世界證據之間，使所有參與者能以不同投影共享同一個可追蹤結構。

這並不表示未來每個小操作都需要龐大規格。

低風險、可逆、局部且具有穩定預設的意圖，可以被高度壓縮與自動化。真正需要被展開的，是那些：

- 有歧義；
- 跨模組；
- 影響他人；
- 使用敏感資料；
- 產生費用；
- 對外公開；
- 修改權限；
- 作用於物理世界；
- 不可逆或難以補償；

的世界變更。

因此，意圖程式設計的成熟標準，不是 AI 能多快把一句話變成程式碼。

而是：

【AI 能否知道哪些地方可以自動編譯，】

【哪些地方必須保留未知，】

【哪些地方需要人類選擇，】

【以及每一項世界效果如何被驗證與承擔。】

```
intent_ir:
  intent_id: "publish-site-20260727"
  goal:
    statement: "publish the approved site update"
    success:
      - "approved release is publicly visible"
    non_goals:
      - "change domain ownership"
  world:
    site: "evemisslab.com"
    environment: "production"
    release: "approved-release-2026-07"
  actors:
    principal: "site-owner"
    executor: "deployment-agent"
    affected:
      - "site-visitors"
  state:
    current: "approved"
    target: "public-and-verified"
  effects:
    external:
      - "deploy artifact"
      - "shift traffic"
    irreversible:
      - "public content may be cached"
  permissions:
    required:
      - "production-deploy"
    human_approval:
      - "shift-public-traffic"
  uncertainty:
    - id: "cdn-propagation"
      kind: "requires-external-evidence"
  verification:
    - "public homepage journey passes"
    - "release hash matches approved artifact"
  projection_registry:
    source_ir: "publish-site-20260727"
```

projections:

code:

- path: "deploy/generated/publish_site.ts"
- generated: true

workflow:

- id: "publish-site-workflow-v2"

ui:

- view: "release-approval-panel"

contract:

- id: "deploy-release-v3"

test:

- suite: "public-journey-smoke"

runtime:

- plan: "deployment-plan-889"

audit:

- evidence_chain: "release-publication-889"

unresolved_decision:

- decision_id: "delete-account-mode"

source:

- statement: "delete my account"

alternatives:

- id: "deactivate"
 - reversible: true
- id: "anonymize"
 - reversible: false
- id: "permanent-delete"
 - reversible: false

decision_owner: "account-owner"

delegation_allowed: false

execution_blocked_until_decided: true

effect_preview:

- intent_id: "refund-order-932"

predicted:

states:

- "payment -> refund-pending"
- "order -> cancellation-review"

external_calls:

- "payment-provider.refund"

affected_actors:

- "customer"
- "merchant"
- "finance-team"

irreversible:

- "provider refund request"

recovery:

- "manual reconciliation"

confidence:

overall: "medium"

unknown:

- "provider final settlement time"

- PU-2-01 程式語言不等於文字：結構、語意與執行的基本分離
- PU-2-02 符號作為算子：從靜態字元到可組合計算閉包
- PU-2-03 語法—語意—效果：程式語言的三層存在結構
- PU-2-04 意圖中介表示：從自然意圖到多重可執行投影
- PU-2-05 可編譯世界：程式執行作為世界狀態差分
- PU-2-06 後文本程式語言：意圖、結構、驗證與物理耦合的統一框架

SECTION

Neo.K／EveMissLab 相關理論

- Neo.K with Aletheia，《程式語言不等於文字：結構、語意與執行的基本分離》，2026。
- Neo.K with Aletheia，《符號作為算子：從靜態字元到可組合計算閉包》，2026。
- Neo.K with Aletheia，《語法—語意—效果：程式語言的三層存在結構》，2026。
- Neo.K with Aletheia，《程式不等於程式碼：可執行問題模型的基本定義》，2026。
- Neo.K with Aletheia，《責任、模組與契約：從檔案分類到系統邊界》，2026。
- Neo.K with Aletheia，《可視化作為外部結構記憶：多角色、多尺度的程式理解》，2026。
- Neo.K with Aletheia，《可編譯世界：從程式執行到世界狀態演化》，2026。

SECTION

一般理論背景

- Aho, A. V. et al., Compilers: Principles, Techniques, and Tools, 2006.

-
- Fowler, M., Domain-Specific Languages, 2010.
 - Erdweg, S. et al., “The State of the Art in Language Workbenches,” 2013.
 - Omar, C. et al., “Hazelnut: A Bidirectionally Typed Structure Editor Calculus,” 2017.
 - Pierce, B. C., Types and Programming Languages, 2002.
 - Scott, D. and Strachey, C., “Toward a Mathematical Semantics for Computer Languages,” 1971.
 - Hoare, C. A. R., “An Axiomatic Basis for Computer Programming,” 1969.
 - Harel, D., “Statecharts: A Visual Formalism for Complex Systems,” 1987.
 - Knuth, D. E., “Literate Programming,” 1984.
 - Green, T. R. G. and Petre, M., “Usability Analysis of Visual Programming Environments,” 1996.
 - W3C PROV-O and provenance modeling standards.
 - Research on program synthesis from natural language and neuro-symbolic program generation.
 - Research on bidirectional transformations, lenses, and multi-view consistency.

SECTION

v0.1 — 2026-07-27

- 完成十八篇地基論文第十六篇。
- 建立意圖中介表示十元模型。
- 提出自然意圖不等於可執行程式。
- 建立意圖、澄清意圖、語意 IR、效果計畫與可執行投影五階鏈。
- 將意圖編譯定義為解析、綁定、澄清、限制、規劃、驗證與投影。
- 區分詞彙、指稱、規範與效果四類不確定性。
- 建立未知、歧義、爭議、人類選擇、外部證據與域外節點。

-
- 提出不可代理選擇保留原則。
 - 建立假設帳本、來源意圖與節點可追蹤。
 - 建立 IR 生命週期與生成、授權、執行分離。
 - 建立效果計畫與效果預覽。
 - 建立程式碼、工作流、UI、契約、測試、文件、Runtime 與稽核八類投影。
 - 建立多投影一致性與共同權威結構。
 - 建立人類、AI、編譯器與 Runtime 的共同編譯責任。
 - 建立權限鏈、IR 驗證階梯、版本、分支、撤回與結構記憶。
 - 加入網站、退款、報表、AI Agent、醫療提醒、研究發布與物理設備案例。
 - 提出二十二類失敗模式與十六項可證偽研究方向。
 - 完成與 PU-2-05《可編譯世界》的銜接。