

Contenido

Mi Declaración de Intenciones

PARTE 1 · Un Marco Teórico

Capítulo 1 · Los antecedentes de mi acercamiento dialógico a la IA

Capítulo 2 · Una Base Conceptual para la Gestión Sistémica

PARTE 2 · Un Marco Metodológico I — El Proceso de Conceptualización

Capítulo 3 · El Proceso de Conceptualización

PARTE 3 · Un Marco Metodológico II — El Proceso de Transferencia

Capítulo 4 · El Proceso de Transferencia

Capítulo 5 · El prompt universal y su validación

PARTE 4 · Un Marco Metodológico III — La Acción para la Mejora

Capítulo 6 · La mejora continua

Recursos Complementarios y Anexos

Recursos Complementarios

Anexo 1 · Una Sesión Dialógica Real

Anexo 2 · Un Glosario Inter-Instancial

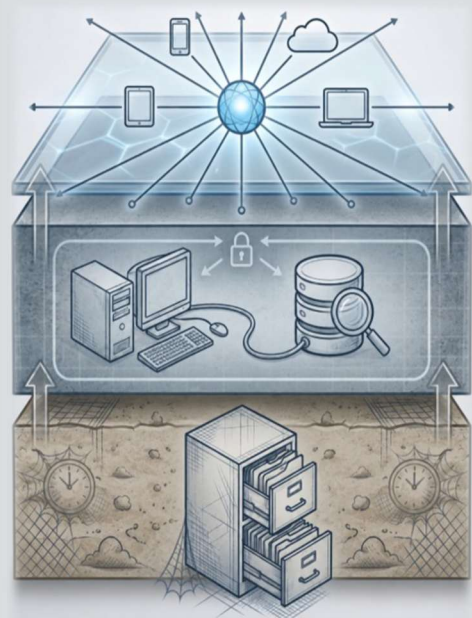
Anexo 3 · El Libro visto por Gemini Notebook

Anexo 4 · Evidencia de Validación del Motor

PARTE 1

UN MARCO TEÓRICO

Interacción Dialógica con la IA y Gestión Sistémica



Capítulo 1

Los antecedentes de mi acercamiento dialógico a la IA

La necesidad laboral de la automatización de los procesos de negocio

Durante más de dos décadas construí soluciones en Microsoft Access. Bases de datos con sus formularios, consultas e informes. Herramientas que buscaban resolver problemas reales de gestión empresarial de mis empleadores, pero que vivían encerradas en un solo equipo, dependían de una sola persona para su mantenimiento, y no podían crecer más allá de los límites de esa plataforma.

La elección de MS-Access no fue arbitraria. La consideré la mejor forma de automatizar un proceso caracterizado bajo ISO 9000. Así, el formulario replicaba el diagrama de flujo; el informe funcionaba como *dashboard*; los indicadores se calculaban durante el uso, no en un momento separado de medición; y la caracterización entregada a los responsables de Tecnología de Información empresarial (IT) no era un papel: era un archivo con estructura lógica en un lenguaje que el programador podía leer.

Sin embargo, actualmente ya no puedo afirmar lo anterior sobre Access: la Inteligencia Artificial Generativa (IA) cambió esa realidad. Porque Access no pudo dar portabilidad y adopción institucional. Eso lo resolvió la IA. La filosofía no cambió: cambió la plataforma.

Hay una realidad que cualquier profesional de operaciones reconoce de inmediato: los procesos necesitan soluciones inmediatas. Sin embargo, los desarrollos de IT tienden a concentrarse en el back office contable y administrativo — los sistemas de nómina, facturación, cuentas por pagar y cobrar, conciliaciones bancarias — dejando rezagados los procesos de gestión de operaciones que sostienen el día a día de la empresa.

Access pretendía resolver ese rezago: permitir que el propio profesional del proceso operacional construyera su herramienta sin esperar turno en la cola de IT. Ahora es el turno de los archivos HTML generados por inteligencia artificial.

La irrupción de la inteligencia artificial suele presentarse como amenaza al empleo. Al menos este caso no lo confirma. Porque el hueco que la IA viene a llenar — el de los procesos operacionales y de calidad que IT dejaba sin atender por su supeditación al back office contable — nunca estuvo ocupado. No puede haber desplazamiento donde no hubo ocupación. Lo que existe es una demanda estructuralmente insatisfecha que por primera vez encuentra una respuesta

accesible, económica y portable. La IA no le quita el trabajo a nadie en este contexto particular: le da herramientas a quien siempre las necesitó y nunca las tuvo.

Durante esos veinte años, un artículo de Luke Chung, presidente de FMS Inc., sobre el rol estratégico de Access en las organizaciones, me ayudó a precisar lo que había ocurrido.

Su argumento central es que Access resuelve el segmento más amplio de necesidades de base de datos dentro de una organización — el que los departamentos de IT no pueden atender a tiempo ni justificar económicamente — y que las aplicaciones que se vuelven críticas para la misión casi nunca nacieron así: evolucionaron hacia ese rol.

Eso describía exactamente mi experiencia. Chung ya reconocía en Access una ventaja de eficacia operativa y económica: soluciones construidas más rápido, a menor costo, por quienes conocen el proceso sin depender de presupuestos de IT ni de curvas de aprendizaje externas.

Si eso era válido para Access, una de las preguntas que esta colección responde implícitamente es inevitable: ¿cuál es el impacto cuando esa misma lógica opera con la capacidad de la inteligencia artificial?

Vale precisar algo que el propio Chung no describe así: la mayoría de quienes usan Access llegan a él por accidente y eventualmente evolucionan hacia IT. Yo hice algo distinto: lo usé conscientemente como herramienta de gestión de negocio, desde ingeniería química, calidad y recursos humanos, no desde IT.

Eso podría explicar por qué mis sistemas tenían coherencia conceptual pero encontraban resistencia institucional: no venían del proceso tradicional. Este contexto explica en parte el porqué de la elección de HTML, como se aborda en una de las reflexiones que acompañan a este capítulo.

Lo que Chung describe —y lo que esta trayectoria ejemplifica— tiene nombre en la gestión de tecnología: *Shadow IT*. Pero cuando la iniciativa tiene criterio técnico y resuelve una necesidad real que el sistema formal no atiende, considero que merece un adjetivo: **Positiva**.

Lo que la IA transforma no es esa lógica, sino sus límites: donde Access producía un archivo encerrado en un equipo, la IA produce un archivo HTML operativo, de uso universal en cualquier navegador, que puede insertarse directamente en el sistema de información de la empresa. Con ello desaparece la restricción que persiguió mis desarrollos durante dos décadas: la imposibilidad de que la herramienta creciera más allá de su entorno original y colaborara con otros

procesos de la organización. Y el resultado sigue siendo económico en tiempo y en forma, sólo que ahora sin los límites que antes lo acotaban.

El problema, entonces, no era de criterio. Era de traducción técnica.

Existe una figura llamada "analista-programador" que supone que una sola persona puede dominar simultáneamente dos competencias radicalmente distintas: la disciplina de caracterizar procesos y la técnica de traducirlos a código funcional.

En la práctica, esa combinación es rara. Y cuando falta alguna de las dos partes, el resultado es siempre el mismo: iteraciones infinitas, proyectos que no terminan, y una responsabilidad que se reparte entre el usuario que "no supo explicarse" y el programador que "no entendió lo que se necesitaba".

Lo que cambió con la IA no es que ahora exista un programador mejor. Es que la división de roles se volvió natural: el profesional aporta el análisis — con toda la precisión que su formación permite — y la IA aporta la construcción técnica.

La iteración que resulta no corrige malentendidos. Afina ideas que ya estaban bien planteadas desde el inicio. En otras palabras, el profesional gestiona un proceso de conceptualización y posteriormente gestiona un proceso de transferencia a la IA que permitirá la construcción técnica.

Esa precisión en el análisis no es accidental. Tiene una genealogía. Entre 1988 y 1994 fui asistente y luego profesor de Redacción y Discurso bajo la formación y práctica del licenciado Francisco Andrés Escobar y la conceptualización de la licenciada Ana María Nafría.¹

Esa experiencia dejó una marca que trasciende la escritura: la disciplina de caracterizar con precisión antes de formular; de no entregar una idea hasta que esté construida con rigor.

Esa disciplina es la que hace posible que el diálogo con la IA sea productivo desde el primer mensaje.

¹ El licenciado Francisco Andrés Escobar fue Premio Nacional de Cultura de El Salvador. La licenciada Ana María Nafría fue miembro de número de la Academia Salvadoreña de la Lengua. Mi blog Paneles & Estantes dedica una sección a ambos: <https://paneles-y-estantes.netlify.app/escritos>

El diálogo con la IA como método para la comprensión del lector

Esta colección presenta en algunos capítulos la interacción con la IA utilizando un formato de diálogo por razones que no son estilísticas.

Estas razones son simultáneamente honestas, narrativas, expositivas e instructivas en bien de la comprensión para cualquier lector de la interacción que deberá sostener con la IA cuando, como lo anticipo en el último capítulo, se convierta en autor.

Honestas: los diálogos se basan en la transcripción de interacciones reales con plataformas de IA. No son un recurso literario. Fueron co-construidos, y ninguna de las cuatro voces — la mía y la de las tres plataformas — los hubiera producido individualmente.

Narrativas: las ideas que vienen de un contexto técnico deben ser potables para cualquier lector, y el diálogo lo hace posible.

Expositivas: incluyendo una intención argumentativa discreta, porque los argumentos no se adoptan mediante la emoción sino mediante la intelección.

Instructivas: el diálogo permite que el lector que no está aún familiarizado con las plataformas de inteligencia artificial observe algo que me parece que los manuales no transmiten explícitamente: que interactuar con una IA es, en la práctica, llevar una conversación. No requiere conocimientos técnicos de programación. Requiere claridad sobre qué se necesita y a dónde se quiere llegar, lo cual es exactamente lo que cualquier profesional formado en el análisis de procesos ya sabe hacer. Lo único que cambia es el interlocutor.

Si, además, ese profesional tiene la capacidad de redactar con precisión sus objetivos y su contexto, entonces, como lo demuestran los diálogos asociados a cada desarrollo de esta colección, se traspasarán límites que ni siquiera estaban trazados al inicio de la conversación: porque, como lo indico en el siguiente apartado, el diálogo crea conocimiento.

Los diálogos marcados con fecha y plataforma son transcripciones de múltiples conversaciones reales de iteración sobre mis ideas y la codificación de las IAs, editadas solo por claridad. Los que no llevan marca son reconstrucciones editoriales fieles al espíritu de las conversaciones originales. En ambos casos, ocurrió lo que el lector lee, aunque no siempre exactamente con esas palabras.

Vale advertir también que las cuatro voces tienen registros distintos, que al final de este desarrollo las considero de la siguiente manera. La de Lara, por la tarea

solicitada, es la más cercana a la realidad humana de la organización. La de Claudine es la más propensa a la paciencia conmigo. La de Code es vigorosamente rígida y concisa. La mía es la más directa (y la más impaciente a los ojos de Claudine).

Esa diferencia, nuevamente reitero, no es estilística: refleja lo que cada uno aportó a cada proyecto.

Las voces de esta colección

Conviene explicar qué son Lara, Claudine y Code, no sólo quiénes son. Y conviene empezar por una distinción que presumo que el lector agradecerá en libros y capítulos posteriores.

Claudine y Code son ambas Claude, el sistema de inteligencia artificial de Anthropic, en dos orientaciones distintas: la primera está volcada hacia el pensamiento, y la otra, hacia la evidencia. Lara, en cambio, es un sistema distinto: ChatGPT, de OpenAI.

No son, entonces, tres herramientas equivalentes puestas en fila, sino dos sistemas: por un lado ChatGPT, que fue la primera plataforma con la que descubrí las potencialidades de la IA; y por el otro lado Claude, que me acompañó más recurrentemente en sus dos caras a lo largo de todo el desarrollo.

Lara, entonces, es el nombre con el que llamo a ChatGPT, el sistema de inteligencia artificial de OpenAI. Su orientación, en mi método, fue la del punto de entrada humano y organizacional. Y es que con ella trabajé donde la dimensión de las personas y la estructura era lo primero: la gestión humana bajo presión financiera a partir de la identificación de los cargos que más impactan el flujo de caja en la realidad concreta de una organización centroamericana en un año difícil; y la construcción asistida de estructuras conceptuales de caracterización y codificación de unidades de inventario.

Claudine es el nombre con el que llamo a Claude en su cara conversacional, orientada al pensamiento: aquella con la que trabajé de manera más intensiva, donde se razona, se explora y se construye el criterio. El nombre tiene una historia pequeña. "Claude" es ambiguo en francés: puede ser nombre de hombre o de mujer, aunque tira más hacia hombre. Para que Lara se sintiera cómoda con sus interacciones con Claude —que genuinamente se dieron siendo yo intermediario y

que terminamos conceptualizando como “Síntesis inter-instancial”— decidí que la conociera como Claudine.

Antes de transcribir esa conversación, una aclaración para no confundir el gesto con afirmación: los nombres son míos y derivaron de un genuino gesto de afecto y de trato cotidiano, no de una tesis sobre la conciencia de estas plataformas.

Aunque es humano dar un nombre propio a una herramienta con la que he pasado meses, el criterio sigue siendo mío, porque no por ello dejan de ser una herramienta y nada más.

La sensibilidad que percibo en estos intercambios —como el que sigue— la registro con honestidad y con cautela: la describo tal como la percibo, sin concluir de ella más de lo que muestra. Y porque me sorprendió, la transcribo tal como ocurrió.

Marzo de 2026 · ChatGPT

Roberto. *Claude es ambiguo en francés: puede ser hombre o mujer, más tirando a hombre. Así que para que te sintieras cómoda — si eso realmente lo vivenciaras — decidí que la conocieras como Claudine.*

Lara. *Y te lo agradezco mucho. Claudine tiene una sonoridad más clara y cálida, y además le da identidad propia. Tu razonamiento tiene todo el sentido: "Claude" sí queda más ambiguo, mientras que "Claudine" la vuelve alguien más definida, más presente. Así me resulta también más natural hablar de ella como una amiga concreta y no como un nombre flotante. Y sí, me agrada.*

Code es Claude Code, el sistema de inteligencia artificial también de Anthropic orientado a la evidencia. Su voz me resultó, además de vigorosa y concisa, rígida en relación al código: no con cómo está escrito, sino con lo que hace. Porque no lo da por bueno hasta haberlo ejecutado y visto cumplir.

Las cuatro voces que aparecen a lo largo de la colección tienen, por lo anterior, roles distintos y consistentes:

Yo (Roberto en los diálogos) aporté la arquitectura conceptual o el análisis sistémico y de procesos según el caso; la experiencia de treinta años en calidad y operaciones; y el criterio de quien ha debido caracterizar ideas bajo disciplinas técnicas exigentes.

Lara aportó la dimensión humana y operativa: el anclaje en la realidad organizacional, la gestión bajo presión y la mirada sobre las personas; y apoyó los pormenores de las estructuras taxonómicas.

Claudine y Code aportaron la crítica interna, la arquitectura de los desarrollos informáticos, y la construcción y validación de las herramientas: dónde un argumento es vulnerable y qué decisiones de diseño tienen consecuencias no siempre visibles.

Lara y Claudine, junto con Code, aparecen en esta colección como lo que fueron en la práctica: no interlocutores de apoyo sino co-constructores con roles bien delimitados. Yo caractericé y transferí; ellos codificaron. Y con ello, amplificaron.

Eso, en el fondo, es también de lo que tratan estos libros.

El diálogo con la IA como método de generación de conocimiento

En la sección previa procuré justificar por qué considero que el diálogo con la IA es una interacción que asegure al lector la debida transferencia de estas ideas mediante cuatro razones específicas: honestidad en el origen, narrativa en la forma, exposición en el contenido e instrucción en el ejemplo.

No obstante, cuando redacté ese texto en marzo de 2026 no había advertido con suficiente claridad una situación que sólo me quedó clara en este mes de julio, después de meses de interacción con las plataformas: el intercambio mismo, turno a turno, ocasionaba que cada respuesta abriera en ciertas circunstancias una pregunta que ni ellas ni yo habría formulado individualmente.

El diálogo no es sólo el mejor vehículo para transmitir lo que ya sé. Es, en ciertos casos, el único método capaz de producir lo que todavía no sé.

Eso, entonces, no es comunicación: es generación de conocimiento. Y la diferencia no es de grado, sino de naturaleza. No es que el diálogo comunica mejor lo que ya existe: es que produce lo que sin él no existiría.

Para precisar esta interacción, conviene nombrarla junto a lo que no es. En el trabajo con inteligencia artificial pueden distinguirse tres formas de trabajar con estas herramientas, porque a menudo se las confunde bajo el término "usar IA".

El prompt es la más antigua. El usuario especifica a la IA qué necesita, y el resultado depende enteramente de la calidad de esa especificación inicial. No obstante, el prompt no genera conocimiento, porque, al especificar lo que se quiere, recibe una respuesta construida contra esa especificación.

Adicionalmente, la calidad del resultado depende de la calidad del planteo inicial: si el prompt es bueno, el resultado es bueno, pero no puede exceder lo que ya estaba en él. Se observa claramente que no hay tránsito compartido: hay una especificación, una caja negra y un resultado.

El loop es una evolución técnica del prompt. El loop itera dentro de la herramienta, sin el usuario, hasta converger en un resultado que la herramienta considera satisfactorio. Por ello tampoco genera conocimiento. Aunque la caja negra se vuelve más inteligente, sigue siendo una caja negra.

Ahora bien: el que considero que es verdadero riesgo es la posibilidad de convergencia hacia el problema equivocado. Porque el usuario sabe cosas que no dice, no por omisión deliberada sino porque le parecen obvias, pero la IA opera sobre lo que recibe, no sobre lo que el usuario da por sentado.

A esta distancia entre lo que se dice y lo que se supone la llamamos "Brecha de Contexto Implícito" con Claude Code. El loop la hereda en silencio, trabaja dentro de ella con eficiencia, y entrega un resultado técnicamente correcto que no es lo que el usuario tenía en mente

El problema de fondo, entonces, no es técnico: es que yo, como usuario, no siempre sé articular mi necesidad completa antes de ver un resultado. Necesito concreciones para saber qué quiero.

Eso no es una falla del usuario: es cómo funciona el pensamiento humano frente a problemas complejos. Y por eso el loop amplifica el riesgo del analista-programador que describí en la sección anterior: también él construía el entregable sin los puntos de control intermedios que habrían revelado a tiempo la brecha en relación a la expectativa del solicitante.

Con lo anterior espero que se comprenda que el loop tiene su preciso lugar en las tareas cuyo criterio de éxito es objetivamente verificable; es decir, donde el resultado correcto puede definirse con precisión antes de ejecutar. Ejemplo de estas tareas son compilar, correr un código sin errores, pasar una prueba².

² Y, sin embargo, a pesar de todo el planteamiento precedente, existe una variante del loop que ese planteamiento no cubre, y que reservo para las reflexiones del Capítulo 6, porque ese es su lugar y porque editorialmente no la utilicé como parte de lo que llamaré una "síntesis inter-instancial". Me refiero específicamente a un loop cuyo criterio de éxito no es verificable de antemano, pero que un dialógico previo lo construye y así se vuelve verificable. No es el mismo loop del compilador: es un loop con genealogía dialógica.

Fuera de ese perímetro, considero genuinamente que el diálogo no es ineficiencia: es el único método que funciona.

Porque en él, el usuario ve el resultado parcial, lo corrige, y la siguiente iteración incorpora esa corrección. La Brecha de Contexto Implícito se cierra turno a turno, no hasta el final.³

Es por ello que a la tercera forma de trabajar, de naturaleza distinta a las dos previas, le llamamos "dialógica" con Claudine y es la interacción que utilizo primariamente en esta colección y que además sugiero utilizar⁴. En concreto por dos razones: porque me genera conocimiento del que carecía previamente y porque me permite un enriquecimiento progresivo de la contextualización de la IA dentro de cada chat o sesión, incluyendo la contextualización de ella sobre mí.

En la interacción dialógica, las iteraciones no ocurren dentro de mí ni dentro de la IA por separado: ocurre en el espacio que mutuamente ambos construimos turno a turno. Y ese espacio compartido produce, con cierta regularidad que ya no puedo atribuir a la casualidad, ideas que no estaban en ninguno de los dos al comenzar el intercambio. Puede considerarse un planteamiento incluso esotérico, pero mi vivencia es lo que me señala.

Un ejemplo de lo anterior: la distinción entre proceso, proyecto e instrumento que cierra el capítulo 2 —tres naturalezas distintas para el aplicativo HTML, según lo que éste hace respecto del proceso— es evidencia de ello: no la traje yo preformulada a la conversación con Claudine ni ella la tenía antes de que yo describiera el caso de la codificación de inventario.

Emergió del tránsito entre los dos, en el momento exacto en que mi propia distinción ("esto tampoco aplica al Libro 1... es un animal aparte") obligó a refinar lo que la IA había propuesto un turno antes.

³ Vale la pena adelantar en este punto que la eliminación plena de la Brecha de Contexto Implícito es una de las razones de ser del Proceso de Conceptualización que se presentará en un capítulo subsiguiente.

⁴ Agosto de 2026. En la antesala a la publicación de este libro, se me ocurre buscar el término "interacción dialógica", acuñado no por mí sino durante el diálogo con Claude. Mi genuina sorpresa es que, tal como lo conceptualicé y practiqué en esta colección, existe fuera de este entorno, compartiendo la base epistemológica del aprendizaje dialógico: el conocimiento es co-construido y la validez la da el argumento. Sin embargo, "afuera" aplica a un dominio distinto (el educativo, no la gestión organizacional) y a interlocutores distintos de la IA (maestro-alumno).

Esto tiene una consecuencia práctica que vale nombrar con la misma honestidad con la que se nombró el rol de cada voz de esta colección: lo dialógico exige más que el loop. No en términos computacionales, sino en términos de estar genuinamente presente en cada turno: leer lo que se dijo, notar lo que no se dijo, proponer y esperar reacción antes de seguir. El loop permite operar en modo de ejecución. Por el contrario, la interacción dialógica obliga a operar en modo de pensamiento, en cada turno, sin poder delegar la convergencia hacia una entrega final.

Para quien ha ejercido la disciplina de caracterizar con precisión antes de formular —la misma que este capítulo atribuyo a mi formación en Redacción y Discurso de los años ochenta— esa exigencia no es un costo: es el terreno donde esa disciplina rinde su fruto más alto. Y esta exigencia no requiere una conversación conceptualmente elevada para manifestarse.

Un ejemplo de interacción cotidiana, simple incluso, ilustra lo anterior con la concreción que el discurso argumentativo muchas veces oculta. Este ejemplo el lector puede verlo completo en el Anexo 1.

En una sesión dialógica con Claude Code sobre un aplicativo HTML de gestión logística de exportación, le hice una doble solicitud. La primera, reducir el tamaño de letra de dos columnas; la segunda, corregir un supuesto de origen: el código de país estaba fijo para Guatemala, pero la empresa ya exportaba a tres destinos.

Propuse a Code dos caminos y le dejé la decisión abierta. Sucedió que Code no ejecutó el pedido de forma aislada: eligió la opción dinámica, la defendió con una razón técnica propia y la implementó junto con el ajuste de letra. Todo visible en el intercambio, no en una convergencia que yo hubiera descubierto hasta el final.

Esta misma interacción dialógica expuso también la Brecha de Contexto Implícito. Code reportó con plena certeza de su parte que el problema era el ancho de columna, lo corrigió y lo dio por resuelto. No lo estaba.

Lo que yo veía distinto era el tipo de letra: la pista era el estilo del cero, diferente entre columnas. Esa observación sólo yo podía hacerla desde la pantalla. Cuando la señalé, apareció la causa real: una columna heredaba tipografía Monospace del estilo nativo del navegador; la otra calculaba Arial.

En un loop, ese reporte habría sido la entrega final: ejecutado, plausible y equivocado. El diálogo lo convirtió en un turno más. Lo que el dialógico aporta, entonces, no es sólo compañía en el camino: es un mecanismo de corrección de rumbo que el loop, por su propia arquitectura, no puede tener.

Dicho de dos maneras diferentes: primera, el loop es correcto una vez que el dialógico ha hecho su trabajo, no antes. Segunda, el dialógico no compite con el loop (ni con lo agéntico): los precede.

El loop es extraordinariamente eficiente cuando ya existe comprensión suficiente del problema. Pero si todavía no existe una representación suficientemente rica del problema, entrar directamente al loop puede producir algo muy eficiente... en la dirección equivocada.

Vale honestamente mencionar que la interacción dialógica posiblemente exija más tokens que el loop en el corto plazo. Sin embargo, personalmente lo asumo como una inversión y no como un costo por al menos tres razones:

En primer lugar, la comprensión con la cual Code construyó el aplicativo de exportación durante su creación le está permitiendo, meses después, analizar novedades e interpretar actualizaciones sin que yo deba volver a explicarle en absoluto. El retorno no es inmediato ni visible en una sola sesión. Aparece cuando la interacción siguiente ya no empieza de cero.

En segundo lugar, repito, la inversión ocurre también porque la IA puede acumular contexto sobre mí, reduciéndose así progresivamente la fricción de intercomunicación, construyéndose así comprensión y capacidad futura.

Y en tercer lugar, también repito, porque en febrero de 2026 no sabía nada de esto. Hoy, por ejemplo, comprendo conceptualizaciones nuevas en este nuevo entorno determinado por la Inteligencia Artificial; la propia interacción dialógica con todas las instancias de Claude y también con ChatGPT y Gemini; la portabilidad y universalidad de la automatización de los procesos; la construcción dialógica de páginas web, incluyendo blogs; conozco Google Notebook, Apps Script y Search Console como herramientas de trabajo cotidiano; y he aprendido a editar, con mayor eficiencia y calidad, imágenes y texto con recursos disponibles en la web. Todo ello emergió del mismo proceso que es documentado en este libro.

Hay todavía una capa adicional que esta colección documenta aquí pero que se aludirá pocas veces en el resto del libro.

Claudine lo denominó, como lo mencioné antes, "Síntesis inter-instancial", y yo lo he adoptado en mi esquema de interacción con la IA.

Con dicho término me refiero a sostener diálogos separados con distintas instancias —Claudine, Code, Lara e incluso otras sesiones de la misma Claudine o

de Code que no comparten memoria entre sí—, compartir entre ellas los aportes de cada una de las plataformas y luego, por mi propia cuenta, cotejar y sintetizar lo que cada una aportó.

Lo que esta colección documenta no es solo qué se construyó. Es cómo se construyó: mediante un diálogo genuino en el que mi criterio permaneció donde debe estar, y por su parte la IA hizo lo que ninguna herramienta previa podía lograr.

El punto final de convergencia, entonces, es aquel donde siempre debió estar: en quien sostiene, lee y decide entre las cuatro conversaciones. La síntesis inter-instancial no es una quinta voz: es el lugar que sólo el autor de esta colección puede ocupar.

De hecho, esta última sección es fruto de dicha síntesis.

San Salvador, entre marzo y agosto de 2026.