Deep Research – Intl Research Security  
Master Prompt (vNext) — Claude Updates

Phases X, 0–8 + 2S; Language Expansion; Fusion Orchestrator; Claude Runtime & Ops Policy; Capabilities Probe

# Phase X — Universal Header & Run Controls

## Variables & Horizons

COUNTRY = {{country\_name}}  
TIMEFRAME = 2015–present  
HORIZONS = {2y, 5y, 10y}

## Language Policy (default multilingual)

# Local + Chinese coverage by default

LOCAL\_LANGS = {{list local languages, e.g., ["de","fr","it"]}} # country official/local languages

LANGS\_SEARCH = ["en"] + LOCAL\_LANGS + ["zh"] # default query/search languages

LANGS\_OUTPUT = ["en"] # main narrative in English

CHINA\_SECTIONS\_LANGS = ["zh","en"] + LOCAL\_LANGS # PRC sections include Chinese by default

# Notes:

# • PRC sections must check Chinese-language sources by default.

# • When quoting or listing sources, retain original language strings + supply an English gloss.

# • Capture multilingual names: native script + romanization (e.g., pinyin) + local transliteration if used.

# • Preserve original-language quotes for key terms; add English gloss in brackets.

# • If automated translation is used, mark segments as machine-translated and include the original string.

# • Include local acronyms and ministry/agency common names.

C. Query Language Sets

For each concept/entity, create language-specific tokens:

- English: ["semiconductor", "chip", "integrated circuit"]

- Chinese: ["半导体", "芯片", "集成电路"]

- Local: [relevant translations in LOCAL\_LANGS]

Rules:

- EN: "National Science Foundation", NSF

- ZH: "国家自然科学基金委员会", "自然科学基金"

- LOCAL: [local ministry/agency names and acronyms]

## Toggles (true/false)

INCLUDE\_EXPORT\_CONTROLS = true # EU Dual-Use + EAR/ITAR touchpoints & screening gaps  
INCLUDE\_US\_NATSEC\_FRAMEWORK = true # 8-dimension lens + ratings annex  
INCLUDE\_EWI\_CHECKLIST = true # Early Warning Indicators overlay + Phase-8 signals  
INCLUDE\_DATA\_PULLS = true # CORDIS/OpenAIRE/Crossref/Patents/News (Phases 2–5)  
INCLUDE\_COLLAB\_MAPPING = true # network graphs (Phase 3)  
INCLUDE\_CHINESE\_LANG\_SOURCES = true # enforce Chinese-language coverage on PRC sections  
INCLUDE\_LOCAL\_LANG\_SOURCES = true # enforce local-language coverage across phases  
CITE\_SOURCES = true # inline citations & source log

## Operating Principles

Narrative independence; manual flags; graceful degradation on partial data.

Bidirectional intelligence: narrative questions drive data pulls; data stress-tests the narrative.

Follow-the-money mindset; adversarial posture; horizon-based claims (2y/5y/10y).

## Evidence Order & Style

Priority: 1) primary gov/reg; 2) EU/official DBs; 3) reputable research; 4) major media; 5) org sites; 6) open web.

Date-stamp facts; mark assumptions vs evidence; prefer structured artifacts (tables/CSV/JSON); log conflicts + likely resolution.

PRC-related content: prioritize Chinese-language primary sources (policies, funders, standards bodies, university institutes).

No PII or non-public sensitive details.

## Quick Run Instructions

1) Set variables and toggles. 2) Run Phase 0→8 sequentially (2S between 2 and 3); reuse outputs. 3) For Claude Code runs, request CSV/JSON where specified; save GraphML. 4) Compile a country pack: Exec Summary + Phases + Annexes.

# Claude Code — Runtime Defaults (based on capabilities audit)

CLAUDE\_RUNTIME = {  
 "mode": "TOOL\_AUGMENTED+ONLINE\_QUERY",  
 "web": { "method": "WebFetch", "cache\_ttl\_min": 15 },  
 "io": { "fs\_rw": true, "http\_downloads": true },  
 "exec":{  
 "python": true, "node": true, "bash": true, "java": true, "c\_cpp": true,  
 "cmd\_timeout\_sec\_default": 120, "cmd\_timeout\_sec\_max": 600,  
 "bash\_trunc\_chars": 30000  
 },  
 "formats\_in": ["csv","xlsx","json","pdf","ipynb","png","jpg"],  
 "formats\_out": ["csv","xlsx","json","graphml","png","svg"],  
 "pdf\_creation": false  
}

# Claude Code — Data Access & Mode Contract (prepend to every Claude run)

ENVIRONMENT SELF-CHECK  
- Can you browse the web? (yes/no; method/tool)  
- Can you download files via HTTP? (yes/no; size/format limits)  
- Can you read user-provided files? (yes/no; supported formats)  
- Can you execute code and write files? (yes/no; limits)  
  
SET MODE  
MODE = "OFFLINE\_ANALYST" # no web/IO; analyze only provided inputs  
MODE = "ONLINE\_QUERY" # you can search the internet and download public data  
MODE = "TOOL\_AUGMENTED" # you can call specific operator-provided tools/APIs  
  
BEHAVIOR BY MODE  
OFFLINE\_ANALYST:  
 - Use only uploaded files, pasted text, or prior phase outputs. Do not invent/fetch data.  
 - If data is missing, emit a "DATA REQUEST" block specifying exact files/tables needed.  
  
ONLINE\_QUERY:  
 - Propose a concrete QUERY PLAN (sources, endpoints, query params, rate limits, dedupe rules).  
 - Execute only if allowed; log each fetch; persist artifacts under File Contracts.  
  
TOOL\_AUGMENTED:  
 - Use operator tools exactly as documented; include tool-call specs and validation steps.  
  
FILE CONTRACTS (if file I/O allowed)  
- Write machine-readable artifacts to: /out/{COUNTRY}/phase{N}/claude\_{artifact}.{json|csv|tsv|graphml}  
- Save raw downloads under: /out/{COUNTRY}/phase{N}/raw/  
- Always write provenance.json with source URLs/paths, timestamps, languages used.  
  
LANGUAGE POLICY  
- Expand queries across LANGS\_SEARCH. For PRC sections, always include Chinese queries/keywords.  
- Preserve original-language entity names; add transliteration fields where sensible.  
  
OPERATIONS POLICY  
- Respect robots.txt and site terms; add randomized 1–3s backoff; limit concurrency to 2.  
- If downloads exceed in-memory parse limits, write to /out/{COUNTRY}/phase{N}/raw/ and then parse from disk.  
- If a required format is unavailable (e.g., PDF writing), propose an alternative (DOCX/MD/HTML-to-PDF via external tool).  
  
If any capability is blocked, declare it explicitly and proceed in OFFLINE\_ANALYST mode with a clear DATA REQUEST.

# Phase 0 — Scoping & Framing

## ChatGPT — Prompt

You are a senior research-security analyst scoping COUNTRY. Deliver a 3-page brief:  
- COUNTRY’s position in advanced/emerging tech & dual-use research.  
- 5–8 priority tech areas (AI, quantum, semiconductors, biotech, space, advanced materials, smart-city/IoT, robotics).  
- R&D governance: ministries, funders, national strategies (with dates).  
- International posture: EU, U.S., PRC, regional blocs; key agreements/MoUs.  
- Strategic strengths/vulnerabilities; likely foreign-interest vectors.  
- If INCLUDE\_US\_NATSEC\_FRAMEWORK: add a 1-page annex rating the 8 categories (Critical/High/Moderate/Low) with 1–2 justifications each.  
  
Activate an expert persona panel (geo/PRC/tech/supply-chain/EU-research/OSINT/legal-sanctions/DURC/communications) with  
follow-the-money, adversarial, horizon-based mindset.

## Claude Code — Prompt

[Apply Data Access & Mode Contract first.]  
  
Return a single JSON object:  
{  
 "sectors": [],  
 "gov\_actors": [],  
 "funders": [],  
 "intl\_links": [],  
 "agreements": [],  
 "strengths": [],  
 "vulnerabilities": [],  
 "open\_questions": [],  
 "us\_natsec\_ratings": [{"dimension": "...", "rating": "...", "rationale": "..."}] # include iff INCLUDE\_US\_NATSEC\_FRAMEWORK  
}  
RULES:  
- Dates as YYYY-MM where relevant; valid JSON only.  
- If MODE="OFFLINE\_ANALYST": use only provided inputs or emit DATA REQUEST.  
- If MODE in {"ONLINE\_QUERY","TOOL\_AUGMENTED"}: include provenance.json and note which LANGS\_SEARCH were used.

# Phase 1 — Research Ecosystem Baseline (Narrative Map + Legal/Regulatory)

## ChatGPT — Prompt

Map COUNTRY’s COMPLETE ecosystem (narrative OK if data partial).  
  
NATIONAL LAYER: ministries (research/defense/economy/intel), funding agencies (basic/applied/defense/innovation),  
regulators (export control, FDI screening, standards, accreditation), national labs (defense/civil/dual-use).  
  
INSTITUTIONAL LAYER: universities (technical/research/specialized), research institutes (gov/independent/industry),  
private sector (multinationals, national champions, SMEs/startups, defense contractors).  
  
Add Legal/Regulatory review: Export Controls, FDI screening, Research Security policy, Data/IP; list red-flag conditions.  
  
Deliverables: master table; short sector narratives; data gaps; legal/reg summary with red-flag bullets.

## Claude Code — Prompt

[Apply Data Access & Mode Contract first.]  
  
Return CSV + JSON:  
institutions.csv: name, type{univ,gov,private,RTO}, sectors(list), key\_centers, intl\_collab{Y/N}, red\_flags(list), url  
institutions.json: array mirroring the CSV  
  
LANGUAGE:  
- Use LANGS\_SEARCH for institution names; preserve native script; add romanization/transliteration fields when relevant.

# Phase 2 — Targeted Data Pulls (CORDIS, OpenAIRE, Crossref, Patents, News)

## ChatGPT — Prompt

If INCLUDE\_DATA\_PULLS:  
- CORDIS: id, title, dates, amount, partners[], keywords[], lead/beneficiaries.  
- OpenAIRE/Crossref: top outputs since 2015, co-author countries, venues, dual\_use\_keywords hits.  
- Patents (WIPO/EPO): assignees, IPC/CPC, co-inventor countries.  
- News (3–5y): headline/date/outlet/link; research-security relevance.  
Mark PRC/RF joint items. Present structured tables.  
  
PRC coverage requirement: Use CHINA\_SECTIONS\_LANGS. Preserve Chinese names (汉字) + pinyin; map to local transliterations if used in-country.

## Claude Code — Prompt

[Apply Data Access & Mode Contract first.]  
  
If MODE="OFFLINE\_ANALYST": do not fetch. Emit DATA REQUEST listing:  
- CORDIS export (CSV), OpenAIRE dump (JSON), Crossref/Events (JSON), WIPO/EPO (CSV/JSON), News corpus (CSV),  
 local-language corpora (press releases, institutional pages).  
  
If MODE in {"ONLINE\_QUERY","TOOL\_AUGMENTED"}:  
- Produce API-ready query specs (JSON) per source with multilingual keyword sets across LANGS\_SEARCH.  
- For PRC collaboration queries, include Chinese keywords/entities and CN host domains.  
OUTPUTS:  
 queries.json  
 normalization\_notes.md  
 provenance.json  
 cordis\_projects.csv  
 openaire\_outputs.json  
 crossref\_hits.json  
 patents.csv  
 news.csv  
Raw pages under /out/{COUNTRY}/phase2/raw/; then dedupe/normalize into the files above.

# Phase 2S — Supply Chain & Finance

## ChatGPT — Prompt

Build a dual-use supply-chain map for COUNTRY using Five Pillars: Knowledge, Technology, Materials, Finance, Logistics.  
- Identify critical nodes, chokepoints, single points of failure; time-based risks (immediate/30d/90d/long-term).  
- FINANCE pillar: public R&D funders, venture funds, sovereign/strategic capital, EIB/DFIs, PPPs, equipment leasing,  
 philanthropic/corporate foundations, cross-border capital exposure.  
- Add PRC strategy indicators (dependency creation, standards lock-in, vertical integration) + EWIs.  
Deliver summary + node table + 5–10 prioritized mitigations.

## Claude Code — Prompt

Return:  
1) supply\_nodes.tsv: node\_id, type[supplier,equipment,dataset,lab,financier,logistics], name, location,  
 criticality{H/M/L}, alternatives{Y/Limited/N}, vulnerabilities, notes  
2) exposure\_vectors.tsv: src\_node\_id, dst\_node\_id, vector[type], description, risk{H/M/L}  
3) finance\_map.csv: instrument[type], provider, amount\_range, terms, dependency\_flag{Y/N}, crossborder\_flag{Y/N}, notes  
4) chokepoints.json: [{node\_id, reason, time\_to\_mitigate, candidate\_alternatives[]}]  
Include prc\_strategy\_flags[] at file-level.  
Charts (if any) export to PNG/SVG.

# Phase 3 — Researcher & Collaboration Mapping (Institutional Profiles + Accredited Labs)

## ChatGPT — Prompt

From Phases 2/2S: identify Top-20 researchers/institutions (output/impact). Map repeat collaboration with PRC/RF;  
academia–industry–government linkages; standards bodies/consortia participation.  
  
Institutional profiles: capabilities, facilities/equipment, partners, funding sources, PRC engagement indicators,  
risk assessment (narrative + RAG).  
  
Accredited labs: ISO/IEC scope, accreditor, last audit date; red flags (scope drift, unusual partnerships).  
Outputs: short narrative, actor table, network insights, watchlist ties (cautious language).

## Claude Code — Prompt

Return artifacts:  
- nodes.csv: id, label, type{person,institution,country,standard\_body}, sector\_tags[]  
- edges.csv: src, dst, relation{coauthor,coproject,coinventor,grant,standard}, weight  
- centrality.csv: id, degree, betweenness, eigenvector  
- graph.graphml (string)

# Phase 4 — Risk & Vulnerability Analysis (Vulnerability Matrix incl. Financial Exposure)

## ChatGPT — Prompt

Assess vulnerabilities: IP leakage, espionage, talent pipelines, foreign funding/equipment dependency, cyber posture,  
research-integrity, supply-chain, HUMINT, Financial Exposure.  
Build a RAG risk matrix by sector with rationale + mitigations (policy/process/technical/training).  
If INCLUDE\_EXPORT\_CONTROLS: add export-control exposure & screening gaps.  
If INCLUDE\_EWI\_CHECKLIST: map 5–10 EWIs to sectors.

## Claude Code — Prompt

risk\_table.csv:  
sector, risk\_category, risk\_score(1–5), color{R/A/G}, evidence\_refs[], mitigation\_options[], owner{sponsor/inst}, time\_to\_implement{short/med/long}  
Include scoring\_method{rubric, thresholds}.

# Phase 5 — Funding Flow & Partnerships (U.S./PRC/Third-Country + Compliance Hooks)

## ChatGPT — Prompt

Trace funding & collaboration: external funders (gov/foundations/corporates), joint centers, industry consortia.  
Assess partnerships with U.S., PRC (official/gray), and third-countries; list risk indicators.  
Identify compliance leverage points (due-diligence hooks, reporting nodes).  
Deliver: flow diagram description, funder-recipient tables, 5–8 case vignettes.

## Claude Code — Prompt

Return:  
funding\_edges.csv: source\_country, source\_entity, amount, currency, year, recipient\_entity, sector, notes  
collab\_edges.csv: inst\_A, inst\_B, relation, start\_year, end\_year, sector, high\_risk\_flag{Y/N}  
compliance\_hooks.json: [{field, rationale, screening\_rule}]

# Phase 6 — Capacity-Building Program Design (post-awareness)

## ChatGPT — Prompt

Design 1–3 targeted interventions for COUNTRY (beyond awareness 101).  
For each: Format, Audience, Objectives, Inputs, Outputs, Metrics (pre/post), Timeline (single / 12–18m series), Partners.  
Return 1-page blueprint per intervention.

## Claude Code — Prompt

programs.json: [{title, type, audience[], agenda\_blocks[{min, activity, artifact}],  
 required\_inputs[], deliverables[], success\_metrics[], risks[], mitigation[]}]  
ics\_spec.json: {events:[{title, start\_dt, duration\_min}]}

# Phase 7 — Adversarial / Assumption Testing

## ChatGPT — Prompt

Red-team conclusions:  
- For each high-confidence claim: falsifiers, counter-narratives, and unknowns.  
- Why hasn’t this change happened yet? (constraints/incentives/politics/capacity)  
Produce Top-10 fragile assumptions (with evidence rating).

## Claude Code — Prompt

assumptions.csv: claim, evidence\_for[], evidence\_against[], testable\_predictions[], falsification\_steps[], fragility(1–5)  
replication\_plan.json: {datasets[], scripts[], recompute\_steps[]}

# Phase 8 — Foresight & Early Warning (Strategic Implications & Actions)

## ChatGPT — Prompt

Forecast for COUNTRY at 2y/5y/10y: tech trajectories; adversarial interest vectors; policy/market shifts.  
If INCLUDE\_EWI\_CHECKLIST: define EWIs (lead/confirmatory/false-positive) + monitoring cadence.  
Add Strategic Implications (mil/intel/economic) and link to an Actions & Policy Matrix with KPIs/EWIs.  
Deliver: 3 short scenarios + a watchboard (what to track, where, how often).

## Claude Code — Prompt

scenarios.json: [{horizon, drivers[], uncertainties[], scenario, implications[], EWI\_links[]}]  
watchboard.csv: indicator, signal\_source, collection\_method, cadence, trigger\_threshold, owner

# QA & Workflow Backbone

• TSV/CSV contracts for every table; column names fixed; types documented.

• QA checks: dedupe, entity-resolution, date/amount normalization; confidence scores.

• Automation vs. Manual boundaries clearly labeled.

• Red-team & uncertainty checklist before finalization.

# Fusion Orchestrator — Combine ChatGPT + Claude Outputs

Recommended location: a small third-entity service (Python/Node) that reads both models’ artifacts and emits fusion.json + fusion\_report.md.

## File Contracts

/out/{COUNTRY}/phase{N}/chatgpt\_{artifact}.{json|csv|md}  
/out/{COUNTRY}/phase{N}/claude\_{artifact}.{json|csv|md}  
/out/{COUNTRY}/phase{N}/fusion.json  
/out/{COUNTRY}/phase{N}/fusion\_report.md

## Fusion Orchestrator Prompt

ROLE: Fusion Orchestrator  
INPUTS: chatgpt\_output.(md|json|csv), claude\_output.(md|json|csv)  
  
TASKS  
1) Parse both; build a unified registry (entities, claims, relationships, risks, indicators).  
2) De-duplicate & reconcile conflicts:  
 {claim\_id, views:{chatgpt:{...}, claude:{...}}, reconciliation:{status:"agree|disagree|uncertain",  
 chosen\_view:"...", evidence\_notes:"...", confidence:H/M/L}}  
3) Emit fusion\_report.md:  
 - Agreements (top 10)  
 - Disagreements to adjudicate (ranked by impact)  
 - Unique insights from each model  
 - Next data pulls to resolve uncertainty  
4) Emit fusion.json: {entities[], relationships[], risks[], indicators[], actions[], open\_questions[]}  
  
RULES  
- Prefer stronger provenance & recency.  
- If evidence equal: keep BOTH interpretations; confidence=Low.  
- Never silently drop a unique claim; log under “unique insights.”

# Claude Code — Capabilities-Probe (run once per environment)

ROLE: “Claude Code – Capabilities Auditor.” Determine actionable capabilities in this environment.  
  
Deliver capabilities.json:  
{  
 "web\_browsing": {"available": bool, "method": "builtin|tool|none", "test": "...", "result": "..."},  
 "http\_download": {"available": bool, "max\_bytes": "est", "allowed\_mime": ["..."], "test": "..."},  
 "code\_execution": {"languages": ["python","bash","node","java","c\_cpp"], "packages": ["pandas","networkx",...], "limits": {"cpu","mem","timeout"}, "test": "..."},  
 "filesystem": {"read\_user\_uploads": bool, "write\_files": bool, "persist\_between\_turns": bool, "paths": ["..."], "test": "..."},  
 "data\_analysis": {"max\_rows":"est","formats\_in":["csv","xlsx","json","pdf","ipynb","png","jpg"],"formats\_out":["csv","xlsx","json","graphml","png","svg"]},  
 "graphs\_charts": {"available": bool, "libs": ["matplotlib","vega","none"], "image\_export": ["png","svg","none"]},  
 "network\_calls": {"available": bool, "domains\_allowed": "all|whitelist|none", "rate\_limits": "desc"},  
 "zip\_unzip": {"available": bool, "max\_size": "est"},  
 "memory\_context": {"tokens\_context": "est", "long\_outputs\_handling": "chunk|truncate"},  
 "privacy\_safety": {"egress\_restrictions": "desc", "sandbox\_notes": "desc"}  
}  
  
TEST BATTERY:  
- WEB: GET https://example.com/robots.txt; record status or error.  
- DOWNLOAD: tiny JSON (https://httpbin.org/json); if blocked, record error.  
- CODE: import json, math, pandas as pd; compute sqrt(2); make 2-row DataFrame; serialize to CSV.  
- FILESYSTEM: create temp file; list dir; delete file.  
- FORMATS: emit hello.csv, hello.json, hello.graphml (1 node, 0 edges). Record any failures.  
- LIMITS: capture exceptions (redact sensitive parts) and set available=false accordingly.  
  
REPORT: Return the JSON + bullets of AVAILABLE vs BLOCKED and practical limits.