APAC Community Call

September 18, 2025

Agenda

- OHDSI News
- Toward a Dynamic Translation Ecosystem for The Book of OHDSI by Michel Walravens
- 2025 APAC Symposium Updates

Congratulations, 2025 Titan Award nominees!

Agnes Kiragga • Akihiko Nishimura • Alexey Manoylenko • ALS TDI's Real World Evidence Team • Andrew Williams • Andrew Kanter • Aniek Markus • Anna Ostropolets • Anthony Sena • Asieh Golozar • ATLAS Development Team • Ben Martin • Bill O'Brien • Bingyu Zhang • Carlos Diaz • Chungsoo Kim • Christopher Knoll • Clair Blacketer • Craig Sachson • Critical Path Institute's Data Science and Data Engineering team • Cynthia Sung • Daniel Prieto-Alhambra • DARWIN-EU Team • Data4Life Team • Dave Kern • Davera Gabriel • Department of Biomedical Systems Informatics, Yonsei University College of Medicine • Deran Mckeen • Diane Corey • Egill Fridgeirsson • Eric Fey • Evanette Burrows • Eye Care and Vision Research WG • FHIR to OMOP WG • Freija Descamps • German Soto • Greg Klebanov • Hannah Lee • Harry Reyes Nieva • HealthPartners Institute • Henrik John • Ian Braun • Ilse Vermeulen • IQVIA OMOP DARWIN Team • IQVIA OMOP Productized Analytics Team • James Gilbert • Jamie Weaver • Jared Houghtaling • Jason Hsu • Jenna Reps • Jiwon Um • Joel Swerdel • John Gresh • Justin Bohn • Katia Verhamme • Lars Halvorsen • Liesbet Peeters • Lotte Geys • Maarten van Kessel • Marc Suchard • Marti Catala Sabate • Martijn Schuemie • Marty Alvarez • Maxim Moinat • Michael Matheny • Michel Walravens • Mike Pauley • Milou Brand • Mitchell Conover • Mukkesh Kumar • OHDSI Belgium Team • Patricia Mabry • Patrick Ryan • Pavan Sudhakar • Peter Hoffmann • Peter Rijnbeek • Polina Talapova • Renske Los • REWARD Team • Richard Boyce • Roger Carlson • Sam Patnoe • SciForce Team • Treatment Patterns Team • Vaccine Vocabulary Team • Will Roddy

TOWARD A DYNAMIC TRANSLATION ECOSYSTEM FOR THE BOOK OF OHDSI

From a need and request of the French speaking OHDSI community in Africa to a Global Language Agnostic Community Driven Translation Project

"It always seems impossible until it's done." — *Nelson Mandela*

"It's kind of fun to do the impossible." — *Walt Disney*

TRANSLATION OF THE OHDSI BOOK — BASIC PRINCIPLES

- Translation is the right of every language community: = equity issue
- Translation should be done within and by the OHDSI community of the translation language
- A translation work group is essential: managers and reviewers
- Maximise use of IT: machine translationis first step, organization and support for every action
- The best reviewers are bilingual OHDSI knowledgeable persons
- Traceable workflow: roles, planning, reviews, corrections, interactions, commits, and audit trails
- Lexical translation: no problem with AI based machine translation

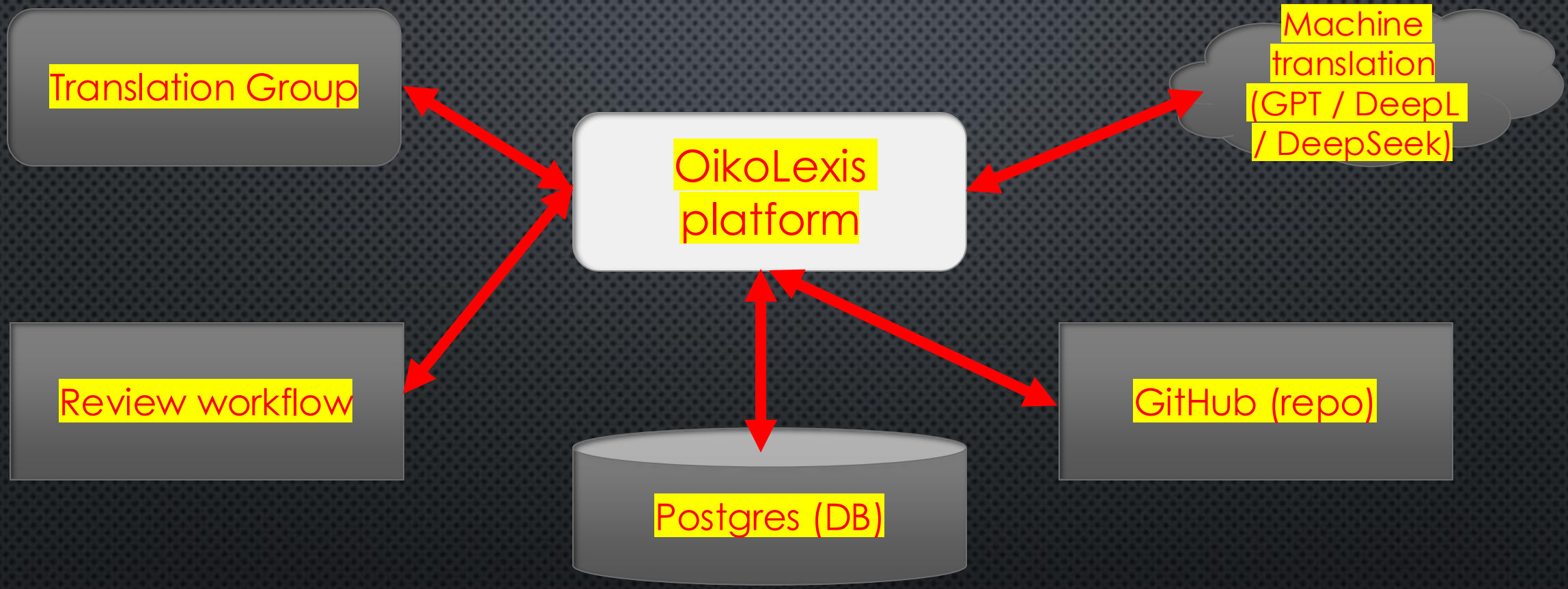
OIKOLEXIS TRANSLATION PLATFORM — BASIC PRINCIPLES

- Human-in-the-loop: MT first, expert review always
- Separation of concerns: GitHub for text; Postgres for metadata & logs
- Language-agnostic via IETF codes (source EN, many targets)
- Traceable workflow: roles, planning, commits, and audit trails
- Modular services: Flask API, React/Vite UI, JWT auth
- Engine-flexible MT: GPT, DeepL, DeepSeek (fallbacks & comparison)
- Deterministic chunking: chapter → chunk_pairs for structured review
- Cost/metrics logging for transparency and optimization

KEY FUNCTIONALITIES

- GitHub integration for versioning; Vite proxy for dev (/api → Flask)
- Subscription and Working group formation
- Role & language choice and management; JWT-based login
- Planning & task assignment (sequential review: Rev1 → Rev2 → Rev3)
- Metrics & pricing logs (tokens, chars, model, time, cost)
- UI-key scanner & translation validator (multi-engine compare)
- Import & normalize: combi-Rmd → chunk_pairs (per chapter, per language)
- Reviewer UI: /review/:chapter/:lang with ordered chunks
- Translate endpoints: /api/translate/estimate and /api/translate/execute
- Cost/metrics logging for transparency and optimization

OikoLexis, project organizing platform



All interactions flow via OikoLexis; bi-directional sync/integration.

WHY "OIKOLEXIS"?

- Name blends two Greek roots that match the platform's purpose.

- οἶκος (oikos) = "home/household" →oikonomia → ecology, economy
  - • A shared home for carefull management of multilingual texts, reviewers, and tools

- λέξις (lexis) = "word/speech" → lexicon
  - • Focus on words, chunks, and UI keys, managed with care

- **OikoLexis = "a home/ecosystem for words and "**
  - • Central hub connecting MT, reviewers, GitHub, and Postgres
  - • Emphasizes stewardship, traceability, and community governance

An ecosystem and home for multilingual terminology and text.

WHY IS OIKOLEXIS AN ECOSYSTEM?

CHARACTERISTICS OF AN ECOSYSTEM:

- Interdependence of elements
- Adaptability of elements
- Feedback loops
- Emergent behavior: COMPLEX PATTERNS, PROPERTIES, OR ACTIONS THAT ARISE FROM THE INTERACTIONS OF SIMPLER ELEMENTS WITHIN A SYSTEM—WITHOUT BEING EXPLICITLY PROGRAMMED OR DIRECTED. HUMAN FACTOR (REVIEWERS) IS WHAT MAKES IT AN ECOSYSTEM!

IS THE PROJECT SUSTAINABLE?

- Open standards: Markdown/Bookdown, IETF language codes
- Separation of concerns: GitHub for text; Postgres for metadata & logs
- Vendor-agnostic MT with transparent pricing & metrics
- Deterministic chunking enables reuse & low-cost incremental updates
- Community governance with audit trails across roles and phases
- Modular Flask/React services — maintainable and replaceable

Built to evolve without lock-in.

IS THE PROJECT SCALABLE?

- Chunk-based parallelism across chapters, languages, and reviewers
- Job orchestration (mt_jobs) for batching and horizontal worker scaling
- Index-friendly APIs & DB schema for high-throughput reads
- Many reviewers can work concurrently on different files (no conflicts)
- Decoupled frontend/backend; cloud-ready deployment
- Registries for languages/tools enable systematic growth

Scale by adding chapters, languages, and people.

WHY CAN WE SAY IT'S AI-BASED?

- **Multi-engine MT** (GPT, DeepL, DeepSeek) with human-in-the-loop review
- **Automated UI-key translation & comparison**; engine selection per language/task
- **Programmable orchestration with fallbacks** (choose engine, retry on error)
- **Metrics logging**: tokens, characters, latency, cost, model_used for governance
- **LLM-assisted checks for terminology & quality** (present/planned)
- **Reproducible configs & prompts versioned in GitHub**

AI augments experts; humans decide.

1) Multi-engine MT (GPT, DeepL, DeepSeek) + human-in-the-loop

- **AI role: Generates the first-pass translation for each chunk/UI key.**
- **Input → output:** source_text + target_lang (+ optional style/term constraints) → translated_text.
- **In the code:** calls like translate_text_with_stats(text, lang, tool_name) (single engine) or translate_text_multi(...) (multiple engines).
- **Human part:** reviewers read AI output in /review/:chapter/:lang, accept/edit, and finalize.

AI augments experts; humans decide.

2) AUTOMATED UI-KEY TRANSLATION & COMPARISON; ENGINE SELECTION PER LANGUAGE/TASK

- **AI role: Produces several candidate translations for the same short key (GPT, DeepL, DeepSeek).**
- System role (not AI): Compares candidates (e.g., identical after normalization? consistent with glossary?) and either:
- auto-accepts when candidates converge or match past decisions, or flags for a human decision when they diverge.
- **Optional AI assist: Ask an LLM to rank candidates**
- ("pick the best given these term rules/style"), but that's a design choice.
- In the code: translate_text_multi(...) returns a dict of candidates; your UI shows the optio

AI augments experts; humans decide.

3) Programmable orchestration with fallbacks.

AI role: Still the worker doing translations; the fallback logic itself is rule-based.

- Flow example: Prefer DeepL for fr/de → if error/timeout/low-confidence, call GPT → if still bad, DeepSeek → hand to reviewer.
- **Optional AI assist: Use an LLM to judge quality (see #5) and trigger fallback when score < threshold.**

AI augments experts; humans decide.

4) METRICS LOGGING (TOKENS, CHARS, LATENCY, COST, MODEL_USED)

- **AI ROLE: INDIRECT—THESE NUMBERS COME FROM THE AI CALLS, BUT THE LOGGING IS NOT AI.**
- WHY IT MATTERS: LETS YOU COMPARE ENGINES, COSTS, AND QUALITY OVER TIME AND GOVERN MODEL CHOICES.

AI augments experts; humans decide.

5) LLM-ASSISTED CHECKS (TERMINOLOGY, QUALITY)

- **AI ROLE: ACTS AS A CRITIC OR PROOFREADER ON TOP OF MT OUTPUT.**

- TYPICAL USES:TERMINOLOGY ENFORCEMENT ("ENSURE 'COHORT' → 'COHORTE' IN FR; DON'T TRANSLATE PROPER NAMES").
- STYLE/FORMAT CHECKS (SENTENCE CASE, PLACEHOLDERS {N}, CODE SPANS).
- SHORT JUSTIFICATION + SUGGESTED REWRITE.PATTERN (PROMPT SKETCH):

SYSTEM: YOU ARE A BILINGUAL REVIEWER FOR {LANG}. ENFORCE GLOSSARY: {JSON GLOSSARY}.
USER: SOURCE: "{SRC}"
CANDIDATE: "{HYP}"
ASSISTANT:
VERDICT: {GOOD | MINOR | BAD}
ISSUES: [...]
SUGGESTION: "..."

- OUTCOME: IF VERDICT=GOOD → ACCEPT; ELSE SHOW THE SUGGESTION TO THE HUMAN.

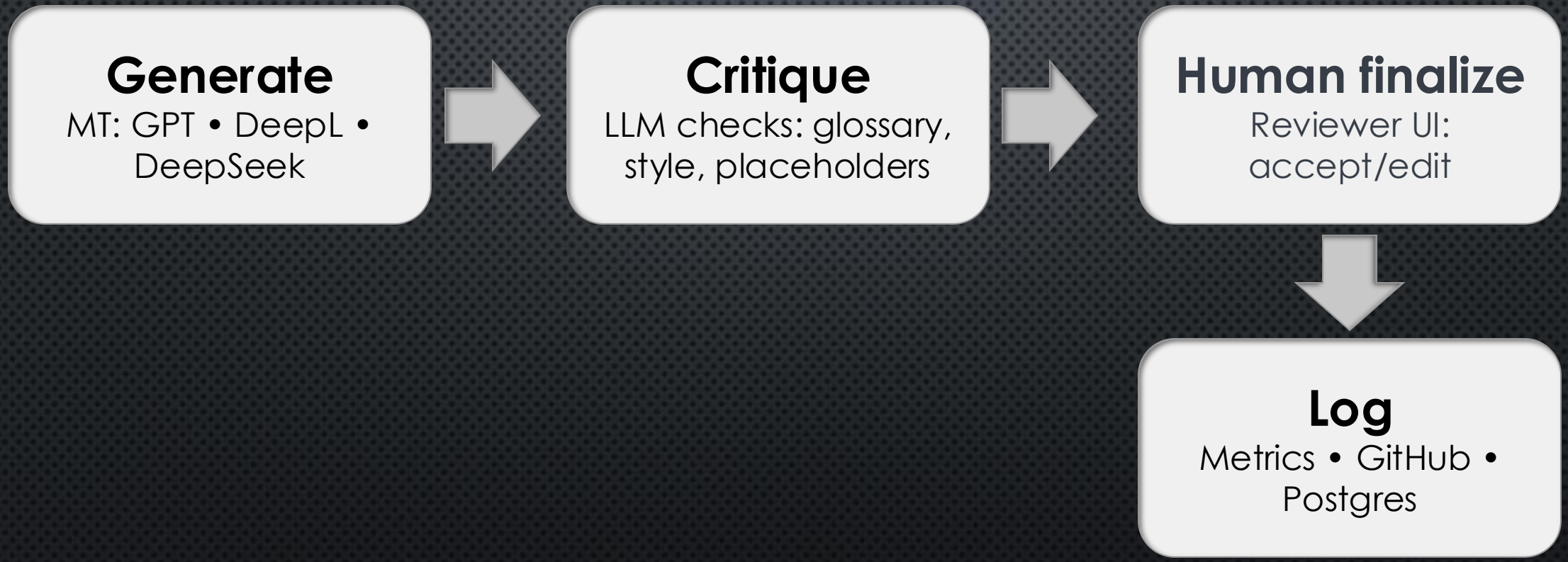
AI augments experts; humans decide.

6) Reproducible configs & prompts versioned in GitHub

- **AI role: None directly; this is governance for AI usage.**
- **Effect:** You can re-run the same model+prompt on the same text and get comparable results; diffs are reviewable.

AI augments experts; humans decide.

AI-in-the-loop — Generate → Critique → Human finalize → Log



Feedback from Log can refine prompts & engine choice in the next run.

Practical part of presentation

- The structure of the Book of OHDS
  - Layout
  - Images and code chunks
  - Content table
  - Index
  - References

PRACTICAL PART OF PRESENTATION

- **MARKDOWN**
- **BOOKDOWN**

Practical part of presentation

- The Book of OHDSI in GitHub
  - Chapters
  - Markdown and Bookdown
- The translation in GitHub
  - The translation repository, a fork of the GitHub source repository
  - Translated chapters and chunks
  - Chunks of source chapters
  - Combi files sourve-chunks (En) & Translated chunks (any transl)

AI augments experts; humans decide.

Practical part of presentation

- The SQL database (PostgreSQL)
  - Direct input tables; example: registration data
  - Functional data
    - Manually inserted; example: tasks, roles, languages
    - Process-created; example: machin translation and chunks
  - Results; example: reviews, loggings of workflow

Practical part of presentation

- The project-platform (OikoLexis)
  - Subscription and login
  - Chosing roles
  - Work group formation
  - Structure of review screen

→


OHDSI

2025 OHDSI APAC Symposium

December 6-7 • Shanghai Jiao Tong University, China

Collaborator Showcase

2025 OHDSI APAC Collaborator Showcase Brief Report Submission Form

Thank you for your interest in the 2025 OHDSI APAC Collaborator Showcase! We are delighted that you are considering joining our research community and presenting your work at this year's symposium. The 2025 OHDSI APAC Symposium will be held in person **December 6-7** at the Shanghai Jiao Tong University in Shanghai, China.

Please take a few minutes to fill out this submission form to help the OHDSI APAC Scientific Review Committee better understand your work. The deadline to submit your brief report is **September 21**. You will receive a confirmation email of your responses upon completion. If the committee has selected your work to be presented at this year's symposium, you will be notified via email by **October 17**.

Should you need to change your responses to any of the questions on this form, please click on the "Edit response" button in the confirmation email you received. Should you need to revise your brief report, please email apacsymposium@ohdsi.org. Your submission will be removed, and you will need to submit again with the revised PDF.

https://forms.gle/gj24ogEbH86xbjrF8

Aug 4 – Sep 21

Abstract Submission

Sep 22 – Oct 6
Oct 7-9 APAC WG workshop

Review by Scientific Committee

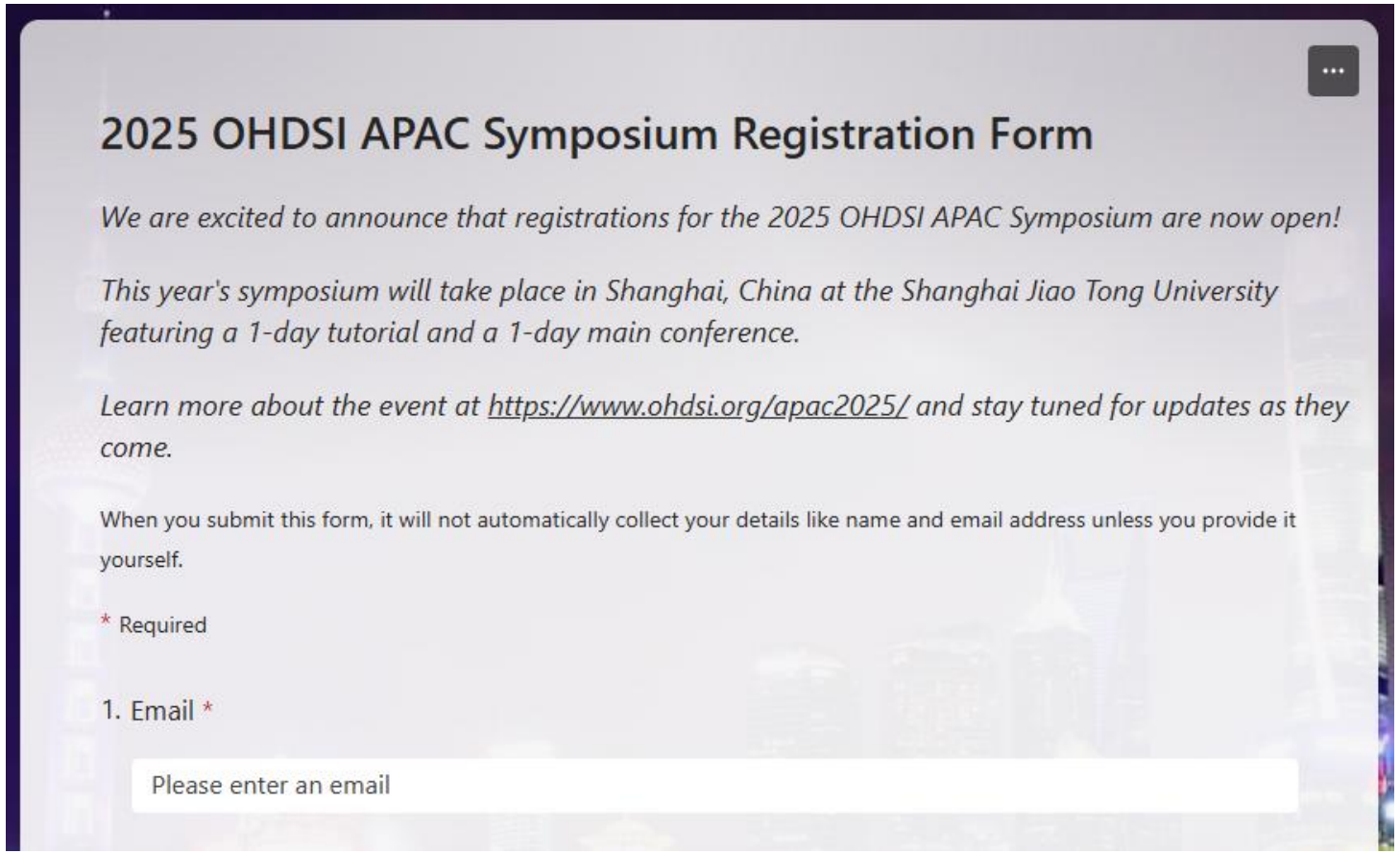
Oct 17

Notification of acceptance

Dec 6-7

Symposium

Registrations



https://forms.office.com/r/rF28Mjk8C0