ECON H191 — Mega Context & Handoff (Master Report)

Last updated: 2025-11-11

This document is a **full-context handoff** for new ChatGPT sessions so they can become immediately effective on Elye Kehat's honors thesis (Econ H191). It explains the project's purpose, identification, data assets, build pipeline, estimation code, outputs, guardrails, and next actions. Use this as the canonical reference before doing any work.

---

1) Executive Summary

Goal. Measure how Israel's **2021–2023 port reforms** (entry of new container terminals and privatizations) affected **container-terminal labor productivity (LP)**, and later assess whether **capital deepening (K/L)** mediates those effects.

Outcome. $\ln(\mathrm{LP})$ at the port/terminal level; LP is stored **in levels** on disk and **log-transformed only inside estimation**.

Identification. Not-yet-treated **event-study/DiD** with **terminal fixed effects** and **calendar-quarter fixed effects**; donut omission of the event quarter $k = -1$. Optional **port-specific linear trends** and **shock windows** (COVID 2020–21; late-2023/24 security disruptions) for robustness.

Main comparisons. Within each port: **entrant** (Haifa: SIPG; Ashdod: HCT) vs **legacy** terminal. Also a **pooled entrant vs pooled legacy** view to improve precision.

Inference. Clustering by **port** (two clusters) with **wild-cluster p-values** for headline scalars and joint **pre-trend F-tests** for leads.

Current bottleneck. Waiting for **direct monthly labor-hours (L)** from port authorities. A carefully engineered **L proxy** (terminal×month) is used for scaffolding and can be swapped one-for-one once direct L arrives.

Status. The full LP pipeline is built; a **canonical quarterized panel** is produced; the **event-study runchain** generates CSVs, plots, and LaTeX tables; inline manuscript snippets are ready to paste.

---

2) Repository Layout (Mental Map)

Root: `THESIS/`

- `Data/` — sources and constructed artifacts by domain.

- `Output/` — tons and TEU backbones.
- `LP/` — stacked and mixed-frequency LP (`LP_Panel.tsv`).
- `L_proxy/` — terminal×month labor-hours proxy and QA.
- `K/` — capital tracks and mediator (`Mediator_K_over_L.tsv`).
- `Time_controls/` — togglable windows (COVID, 2023/24 shocks).
- `Design/` — code, configs, analysis-ready outputs, visuals.
- `Code/Econometrics/` — scripts `01_...` through `09_...` and `model1_params.yaml`.
- `Output Data/` — canonical analysis outputs (CSV/TSV/JSON/PNG/TeX).
- `visuals/` — descriptive and ES plots.

Conventions. - LP is stored **in levels** on disk; take logs only inside estimation. - Pre-reform port-level rows have `terminal = ""`; post-entry, terminal splits into **Legacy** and **Entrant** (Haifa: SIPG; Ashdod: HCT). - Maintain **TEU additivity** and **weight normalization** upstream; do not recompute LP internals in the design layer.

---

3) Upstream Data Assets (Contracts)

3.1 Tons (monthly)

`Data/Output/monthly_output_by_1000_tons_ports_and_terminals.tsv` - Port and (when available) terminal-level monthly tonnage. If terminal rows exist for a (port, month), **sum terminals**; else use the port row (tracked via `tons_source ∈ {sum_terminals, port_row}`).

3.2 TEU (mixed frequency)

`Data/Output/teu_monthly_plus_quarterly_by_port.tsv` - **Monthly by port** + **Quarterly by terminal**. Canonical terminal labels: {Haifa-Legacy, Haifa-SIPG, Ashdod-Legacy, Ashdod-HCT}. Enforce additivity: port-quarter TEU equals the sum of terminal-quarter TEU.

3.3 Labor Proxy (terminal×month)

`Data/L_proxy/L_Proxy.tsv` - Identity-preserving construction with Π (terminal-year TEU/hour). Pre-opening months are **structural zeros**. Designed to be swapped for **direct L** without downstream code changes.

3.4 Stacked LP (mixed frequency)

`Data/LP/LP_Panel.tsv` - **Single source of truth** for LP entering the design layer. Unifies six LP series (monthly-port and quarterly-terminal) with a common schema: `series_id, freq, port, terminal, year, month, quarter, month_index, quarter_index, TEU, tons, L_hours, w, Pi, LP, LP_id, tons_source`.

3.5 Mediator (capital deepening)

`Data/K/Mediator_K_over_L.tsv` - Tidy panel for $\ln(K/L)$, derived from Tracks A–D capital series and aligned to port×time (terminal granularity optional). Used for the **mediation pass** after baseline results are locked.

---

4) LP Construction (S1–S5) — What the Pipeline Does

Key definitions. - w: normalized weight from tons/TEU, re-based so **mean(w)=1** within (port, year). - Π: terminal-year productivity KPI (TEU per labor-hour) from the L proxy table. - **LP formula:** $\mathrm{LP} = w \times \Pi$ with guardrails for monthly-port vs quarterly-terminal series.

S1 — Tons. Apply precedence and produce a clean monthly tons panel with provenance.

S2 — TEU. Canonicalize labels, attach frequency flags, and compute port-quarter TEU as the sum of terminal-quarter TEU.

S3 — L proxy harmonization. Normalize the terminal×month labor table and produce `terminal-year Π`.

S4 — Compute LP. Build monthly-port and quarterly-terminal LP series with acceptance checks: mean(w)=1, uniqueness by grain, additivity, structural zeros pre-opening.

S5 — Stack. Combine series into `Data/LP/LP_Panel.tsv` with a unified schema and QA artifacts.

Important: The **design layer never recomputes LP**; it consumes `LP_Panel.tsv` and performs only frequency harmonization, logging, and event-time mapping.

---

5) Model 1 (Non-Mediation) — Event-Study/DiD Design

Estimand. Event-time coefficients β_k relative to entry/privatization; headline scalar = **Average Post** = mean of β_k for $k \in [1, 4]$.

Fixed effects. Terminal FE and calendar-quarter FE in all columns.

Donut. Omit $k = -1$ from the regression (baseline bin).

NYT controls. Enforce **not-yet-treated** comparisons for post bins.

Toggles. Optional **port-specific linear trends**; optional **shock windows** (COVID 2020–21; late-2023/24).

Inference. Cluster by **port**; report **wild-cluster p-values** for scalars and **leads joint F-test p** in dynamics.

---

6) Design/Code: Python Run-Chain (01 → 09)

All scripts live in `Design/Code/Econometrics/`. The single source of timing truth is `model1_params.yaml` (events, bins, windows, paths).

1. `01_panel1_build.py` → `Design/Output Data/01_panel_port_quarter_full.csv` (+ JSON meta)
2. Assemble **port×quarter LP** from `LP_Panel.tsv` with source precedence (direct port-quarter > geom mean of terminal-quarter > pre-panel fallback).
3. `02_model1_combine_lp_quarter.py` → `Design/Output Data/02_LP_Panel_quarterized.tsv`
4. Produce the **canonical quarterized LP**: `port, terminal, year, quarter, lp`. Drop all monthly rows.
5. `03_lp_enrich_stepwise.py` → `…step3_qtr_Y_tindex.tsv`
6. Add `qtr=YYYYQ#`, `Y=ln(lp)`, and a global `t_index`. Output stepwise TSVs with cleaning logs.
7. `04_build_panel_terminal_sharedpre.py` → `04_panel_terminal_sharedpre_log.csv` (+ meta)
8. Duplicate **shared pre** across entrant & legacy; normalize terminal labels; attach **event clocks** from YAML; compute **NYT eligibility** flags.
9. `05_prep_model1_terminal.py` → `05_panel_terminal_sharedpre_model1.csv` (+ meta)
10. Bin event time with $k = -1$ omitted; **gate NYT** for post bins; write the model frame.
11. `06_run_es.py` → per-terminal ES outputs
12. Run dynamic ES with terminal + quarter FE, **cluster on port**. Write: `06_es_coeffs_{HCT,Legacy,SIPG}.csv`, plots, and `06__meta_model1_es.json`. Stata fallback script is auto-generated if desired.
13. `07_pooled_main_and_figs.py` → pooled entrant vs legacy
14. Re-estimate ES by **roles** (entrant, legacy), compute **Average Post** ($k = 1..4$), and generate **two-cluster wild-cluster p-values** for the scalar. Outputs: `07_es_coeffs_{entrant,legacy}.csv`, plots, `07_table_main.csv`, `07__meta.json`.
15. `08_make_tables.py` → paper tables

16. Use Step-07 CSVs to build LaTeX: `08_table1_main.tex` (Average Post) and `08_table2_dynamic_pooled.tex` (dynamic paths, Panels A/B). Also write CSV mirrors and `08__meta_tables.json`.

17. `09_make_inline_results_tex.py` → manuscript insert

18. Inline the 08 LaTeX tables into a single `09_results_tables_inline.tex` section for quick pasting into the paper.

Descriptives: `Plot_LP_Series.py` generates `visuals/{haifa_lp, ashdod_lp, all_lp}.png` from `LP_Panel.tsv` (levels, not logs).

---

7) Produced Outputs (As of This Snapshot)

- **Quarterization & enrichment:**
- `01_panel_port_quarter_full.csv` (+ meta JSON)
- `02_LP_Panel_quarterized.tsv`
- `03_LP_Panel_quarterized.step3_qtr_Y_tindex.tsv`
- **Terminal shared-pre & model frame:**
- `04_panel_terminal_sharedpre_log.csv` (+ meta JSON)
- `05_panel_terminal_sharedpre_model1.csv` (+ meta JSON)
- **Per-terminal ES (Step 06):**
- `06_es_coeffs_{HCT,Legacy,SIPG}.csv`, `06_es_plot_{HCT,Legacy,SIPG}.png`, `06__meta_model1_es.json`
- **Pooled ES (Step 07):**
- `07_es_coeffs_{entrant,legacy}.csv`, `07_fig_es_{entrant,legacy}.png`, `07_table_main.csv`, `07__meta.json`
- **Tables & manuscript inserts (Steps 08–09):**
- `08_table1_main.{csv,tex}`, `08_table2_dynamic_pooled_{panelA,panelB}.csv`, `08_table2_dynamic_pooled.tex`, `08__meta_tables.json`
- `09_results_tables_inline.tex`
- **Descriptive visuals:**
- `visuals/haifa_lp.png`, `visuals/ashdod_lp.png`, `visuals/all_lp.png`

---

8) Manuscript Table Skeletons (Significance for the Paper)

- **Table 1 — Average Post-Reform Effect on ln(LP) by Port × Terminal.** Headline policy result. Columns: Baseline / +Trends / +Shocks for **entrants** and **legacy** terminals within each port. Rows include Average Post ($k \in [1, 4]$), pre-trend p (leads F), FE flags, N, and clusters. SEs clustered by port; wild-cluster p noted.

- **Table 2 — Dynamic Event-Time (Haifa).** Event-time coefficients for SIPG (entrant) and Legacy across Baseline/+Trends/+Shocks. Demonstrates **no pre-trends** and the **shape** of effects around entry.

- **Table 3 — Dynamic Event-Time (Ashdod).** Mirror of Table 2 for HCT and Legacy.

- **Table 4 — Robustness (Average Post).** For Haifa, Ashdod, and pooled entrant vs legacy: Baseline / +Trends / +Shocks / **Balanced sample** / **Exclude 2020–21** / **Exclude 2023–24**. Confirms results aren't driven by composition or macro shock windows.

---

9) Guardrails, QA, and Known Pitfalls (Already Addressed)

- **On-disk discipline.** LP always in **levels**; logs only inside estimation.
- **Additivity & normalization.** TEU additivity enforced; `w` normalized to mean 1 by (port, year); identity checks for monthly vs annual constructs.
- **Timing truth.** All clocks, bins, and windows sourced from `model1_params.yaml` (no hard-coding in scripts).
- **NYT enforcement.** Already-treated never act as controls for post bins.
- **Small-N inference.** Cluster by port and report **wild-cluster p-values** for the headline scalar; include **leads F-test p** in dynamic tables/figures.
- **Shocks toggles.** COVID and late-2023/24 windows can be included or excluded by spec.
- **LaTeX hygiene.** Use `booktabs` + `threeparttable`; avoid literal Unicode β (use macros), keep preamble clean, and avoid duplicate `\begin{document}`.

---

10) Bottleneck & Swap Plan

- **Bottleneck.** Direct **monthly labor-hours** (terminal×month) from port authorities.
- **Current workaround. L proxy** preserves annual totals and pre-opening zeros and provides terminal-year Π; it is **serviceable for scaffolding**.
- **Swap plan.** Replace `Data/L_proxy/L_Proxy.tsv` with direct L (same schema), rerun S1–S5 → 01–09. All outputs regenerate with no code changes to identification logic.

---

11) What to Do Next (Standard Runbook)

1. **Confirm clocks & bins** in `model1_params.yaml` (competition entry dates; privatization milestones; post window $k = 1..4$; shock windows).
2. **Rebuild design inputs** if `LP_Panel.tsv` changed:
3. Run `01_…` → `03_…` to rebuild `02_LP_Panel_quarterized.tsv` and `…step3_qtr_Y_tindex.tsv`.
4. **Attach clocks & NYT**:
5. Run `04_…` and `05_…` (shared pre, binning, gating).
6. **Estimate ES**:
7. Run `06_…` (per-terminal) and `07_…` (pooled entrant vs legacy; compute Average Post with wild-cluster p).
8. **Build tables & manuscript insert**:

9. Run `08_...` (LaTeX tables) and `09_...` (inline section). Paste `09_results_tables_inline.tex` into the paper.
10. **Optional: Descriptives**:
11. Run `Plot_LP_Series.py` to refresh level-LP figures.

If direct L arrives: swap file, re-run steps 1–6 and update manuscript text noting the data improvement.

---

12) Stretch Goals (After Baseline Locks)

- **Mediation (K/L).** Merge `Mediator_K_over_L.tsv` and run two-stage or interaction-based mediation to quantify the capital-deepening channel; report first-stage strength and mediated share.
- **Alternative clocks.** First ship vs full commercial ops vs privatization; sensitivity to alternative timing.
- **Placebos & falsification.** Off-window event times; pseudo-entries; national aggregates to assess spillovers.
- **Heterogeneity.** Congestion, ship mix, or terminal technology features.
- **Synthetic control (port-level).** High-level complement to terminal ES.

---

13) Quick Start for a New Chat (Checklist)

- [ ] Read this document end-to-end.
- [ ] Confirm understanding of: goal, design, data contracts, run-chain, outputs, and bottleneck.
- [ ] Open `model1_params.yaml` and verify event dates and bins.
- [ ] If inputs changed, run 01→05; otherwise proceed to 06→09.
- [ ] Populate LaTeX tables and paste `09_results_tables_inline.tex` into the manuscript.
- [ ] If asked, stage mediation with `Mediator_K_over_L.tsv`.

---

14) Glossary

- **LP (Labor Productivity):** Constructed as $w \times \Pi$, combining a normalized tons/TEU mix weight with terminal-year TEU/hour Π.
- **NYT (Not-Yet-Treated):** Comparison design where controls in post periods exclude already-treated units.
- **Leads F-test:** Joint significance test of pre-event lead coefficients to assess pre-trends.
- **Wild-cluster p:** p-values robust to few clusters (here: two ports) via Rademacher resampling.

This document is the authoritative context handoff. New sessions should confirm understanding, then proceed with the runbook in Section 11.