

Agentic AI: Theory → Architecture → Practice

เอกสารนี้รวม 3 ส่วนในชุดเดียว เพื่อใช้ได้ทั้ง **อธิบายเชิงทฤษฎี, ออกแบบระบบจริง, และ สอนทีม/ลูกค้าองค์กร**

PART 1: Concept Map — Theory → Architecture → Agent Design

1. รากทฤษฎี (Foundational Theories)

Systems Engineering & Control Theory - Deterministic vs Non-Deterministic Systems - Hierarchical Control (Layer ล่างเร็ว / Layer บนฉลาด)

Decision Theory & Bounded Rationality (Herbert Simon) - ใช้ทรัพยากรเท่าที่จำเป็น - งานง่าย ไม่ควรใช้สมองใหญ่

Expert Systems & Knowledge Engineering - Rule-based (IF-THEN, Decision Tree) - Compliance / SOP / Policy Logic

Industrial Engineering & BPM - Repetitive Work - Standardized Process

Statistics / Machine Learning / NLP - Classification - Summarization - Semantic Parsing (NL → SQL)

2. การแปลงทฤษฎี → Architectural Concept

แนวคิดเชิงทฤษฎี	หลักออกแบบระบบ
Task Decomposition	แตกงานเป็น task ย่อย
Deterministic Logic	ใช้ SLM / Rule Engine
Bounded Rationality	SLM-First, LLM-Escalation
Modularization	Specialist Agent
Hierarchical Control	Router + Confidence Gate

3. การแปลง Architecture → Agent Design

Task Tagging - Repetitive - Deterministic - Rule-based

Agent Roles - SLM Summarizer - SLM Classifier - SLM SQL / Tool Agent - LLM Reasoning Agent (เฉพาะ Escalation)

PART 2: Technical Whitepaper (Enterprise-Ready)

1. Problem Statement

องค์กรส่วนใหญ่ล้มเหลวในการใช้ Agentic AI เพราะ: - ใช้ LLM ทำทุกอย่าง - ต้นทุนสูง - คุมพฤติกรรมไม่ได้ - Audit ไม่ได้

Root Cause: ไม่มี Decision Architecture

2. Principle: Task-Driven Model Selection

แทนที่จะถามว่า "ใช้ model อะไร" ให้ถามว่า

งานนี้เป็นงานประเภทไหน?

3. Task Classification Framework

Tag	ความหมาย	ตัวอย่าง
Repetitive	งานซ้ำ	Daily Report, QC Check
Deterministic	Logic ชัด	Threshold, Formula
Rule-based	อิง SOP	GMP, ISO

งานที่เข้า 2-3 ข้อนี้ → **SLM Candidate**

4. SLM Specialist Design

4.1 Summarizer Agent

- NLP Task: Text Summarization
- Output: Structured Summary
- KPI: Accuracy / Time Saved

4.2 Classifier Agent

- ML Task: Classification
- Output: Label + Confidence
- KPI: Precision / Recall

4.3 SQL / Tool Agent

- NLP + Program Synthesis
- Output: SQL / API Call

- KPI: Query Accuracy / Latency

5. SLM-First Architecture

Flow: 1. Task Ingestion 2. Task Tagging 3. SLM Routing 4. Confidence Validation 5. Escalate to LLM (if needed)

ผลลัพธ์: - Cost ลด 5-20x - Latency ต่ำ - Governance ง่าย

6. Example: Ovantin (Manufacturing)

Task	SLM	LLM
Batch Report Summary	✓	✗
QC Spec Check	✓	✗
Deviation Classification	✓	✗
Root Cause Analysis	✗	✓

PART 3: Training Material (Workshop-Ready)

Module 1: เปลี่ยน Mindset (30 นาที)

- AI ≠ Chatbot
- Agentic AI = Decision System
- Model เป็น Resource ไม่ใช่ Hero

Module 2: Task Tagging Workshop (45 นาที)

Exercise 1. เขียนงานจริง 10 งาน 2. Tag: R / D / Rule 3. เลือก SLM หรือ LLM

Output: - Agent Candidate List

Module 3: Agent Mapping (45 นาที)

Template - Task Name - Input - Decision Logic - Output - KPI - Model Type

Module 4: Governance & Cost Control (30 นาที)

- Confidence Threshold
 - Escalation Rule
 - Cost per Task
 - Audit Log
-

Final Outcome (สำหรับลูกค้า)

- เห็นภาพชัดว่า AI ทำงานอะไร
 - คุมต้นทุนได้
 - ขยายระบบได้จริง
 - ไม่ผูกกับ vendor เดียว
-

Executive Summary (1 หน้า)

Agentic AI ที่ใช้งานได้จริงในองค์กร ไม่ได้เริ่มจากโมเดลที่ฉลาดที่สุด แต่เริ่มจากการออกแบบการตัดสินใจที่เหมาะสมที่สุด

SLM Specialist + LLM Escalation คือโครงสร้างที่ยั่งยืนที่สุดสำหรับ Enterprise AI