

Architektura Zintegrowanego Ekosystemu Inżynieryjnego (CIEE) oparta na GERAM, ISO i OpenCMS

sty 25, 2026

Architektura Zintegrowanego Ekosystemu Inżynieryjnego (CIEE)

Koncepcja Graph-Centric: Pełna implementacja Open Source na Linuxie z hybrydową sztuczną inteligencją i orkiestracją agentową

0. Analiza porównawcza i kontekst rynkowy

Przed szczegółowym opisem architektury przedstawiamy analizę aktualnego stanu rynku systemów PLM/ERP oraz miejsce CIEE w tym ekosystemie.

0.1. Główne analogi

- 1. **Giganci komercyjni (Heavy PLM):** Siemens Teamcenter, Dassault Systèmes 3DEXPERIENCE (Enovia), PTC Windchill.
- 2. **Elastyczne platformy korporacyjne:** Aras Innovator (warunkowo otwarta architektura, ale komercyjny model biznesowy i powiązanie z Windows/SQL Server).
- 3. **Wąsko wyspecjalizowane rozwiązania Open Source:** OpenBOM (zorientowany na SaaS), Ganister (jądro grafowe, ale zamknięty kod źródłowy jądra).

0.2. Porównanie: Plusy i minusy

Kryterium	Systemy komercyjne (Teamcenter/Windchill)	Aras Innovator	CIEE (Nasze rozwiązanie)
Koszt (TCO)	Ekstremalnie wysoki (\$2-5k /użytkownika rocznie + wdrożenie).	Wysoki (subskrypcja za aktualizacje i wsparcie).	Minimalny (\$0 za licencje, koszty tylko za wdrożenie i DevOps).
Elastyczność danych	Niska (sztywne schematy relacyjne, trudne zmiany).	Średnia (Low-code, ale wewnątrz SQL).	Maksymalna (Jądro grafowe Neo4j, dynamiczny model danych).
Integracja AI	Zamknięte "czarne skrzynki" dostawcy.	Ograniczona do API.	Natywna (Lokalna Ollama + Chmurowe Gemini).

Zależność	Pełny Vendor Lock-in.	Zależność od aktualizacji dostawcy.	Pełna suwerenność (Open Source na Linuxie).
Integracja CAD	Idealna "z pudełka" dla własnych systemów CAD.	Dobra (poprzez płatne konektory).	Elastyczna (poprzez otwarte formaty i ETL Apache Hop).

0.3. Ekonomia i trendy 2025-2026

- **Koszt posiadania:** Przejście na CIEE obniża nakłady inwestycyjne (CAPEX) na oprogramowanie do zera. Koszty operacyjne (OPEX) są przekierowywane z opłat licencyjnych na rozwój własnych kompetencji inżynierskich i zespołu IT.
- **Trend "Data-Centric vs File-Centric":** Światowy przemysł odchodzi od zarządzania "plikami" na rzecz zarządzania "danymi". **CIEE znajduje się w czołówce tego trendu, wykorzystując Graf Wiedzy jako warstwę podstawową.**
- **Trend "Agentic AI":** W 2025 roku AI przestało być prostym chatbotem. CIEE wdraża *agentów*, którzy samodzielnie sprawdzają graf pod kątem błędów, co jest niedostępne w większości konserwatywnych systemów PLM.

1. Filozofia i podsumowanie architektury (The Big Picture)

Projekt **CIEE** stanowi zaawansowaną technologicznie realizację architektury przedsiębiorstwa opartą na metodologii **GERAM** (Generalized Enterprise Reference Architecture and Methodology). W przeciwieństwie do klasycznych podejść, ograniczonych przez rozproszone bazy danych, CIEE buduje „system nerwowy” przedsiębiorstwa wokół **Inżynierskiego Grafu Wiedzy**.

Kluczowa zmiana paradygmatu: Całkowicie rezygnujemy z „płaskich” i sztywnych tabel relacyjnych baz danych na rzecz **Grafu (Knowledge Graph)**. W rzeczywistym świecie inżynierskim obiekt to żywa sieć powiązań (wymagania -> funkcje -> komponenty -> zakupy). W CIEE relacja jest obiektem pierwszej klasy. Wykorzystanie wyłącznie komponentów Open Source na platformie Linux zapewnia suwerenność technologiczną, bezpieczeństwo danych i brak uzależnienia od dostawców komercyjnych (Vendor Lock-in).

2. Specyfikacja Open Source & Linux Compliance

Wszystkie serwerowe komponenty ekosystemu są **Linux-native** i posiadają otwarte licencje.

Rola w systemie	Komponent	Licencja	Stos technologiczny (Linux)	Opis
Graph Core (Mózg)	Neo4j Community	GPLv3	Java / Linux Binary	Pojedyncze źródło prawdy (SSOT). Przechowuje topologię projektu.
Orchestrator (Logika)	Node-RED	Apache 2.0	Node.js	Wizualne programowanie przepływów danych i zdarzeń.

Systems Eng. (MBSE)	Eclipse Capella	EPL	Java	Projektowanie architektury funkcjonalnej i logicznej.
Product Data (PIM)	Akeneo PIM CE	OSL 3.0	PHP, Symfony, MySQL	Zarządzanie specyfikacjami technicznymi i danymi mediów.
Project Management	OpenProject CE	GPLv3	Ruby on Rails, Postgre	Zarządzanie zadaniami, wykresami Ganta i tablicami Agile.
Portal / CMS	OpenCms	LGPL	Java, Tomcat	Portal front-end, baza wiedzy i metodologia (PRINCE2).
Resources / ERP	Apache OFBiz	Apache 2.0	Java, XML Engine	Jądro zasobowe: finanse, produkcja, magazyn, zakupy.
Local AI	Ollama	MIT	C++, Go	Lokalne uruchamianie LLM (Llama 3, Mistral) dla zadań prywatnych.
Cloud AI	Google Gemini	Proprietary SaaS	REST API	Potężny silnik analityczny do pracy z kontekstem ponad 1 mln tokenów.
Integration (ETL)	Apache Hop	Apache 2.0	Java	Orkiestracja danych z CAD (DEXPI, STEP, JSON).

3. Siedmiowarstwowa hierarchia ekosystemu CIEE (Model OSI dla inżynierii)

Architektura jest zbudowana w oparciu o ścisłą hierarchię warstw:

Warstwa 7: Poznawcza (Cognitive Layer)

- **Narzędzia:** Gemini 2.5 Flash + Ollama.
- **Funkcja:** AI Agentowe. Analiza grafu pod kątem błędów logicznych, sprawdzanie zgodności projektu ze standardami, wykrywanie kolizji.

Warstwa 6: Prezentacji (Presentation Layer)

- **Narzędzia:** OpenCms + Cytoscape.js + Three.js.
- **Funkcja:** „Pojedynczy widok” (Single Pane of Glass). Interaktywne pulpity nawigacyjne, wizualizacja grafu w przeglądarce i przeglądarki 3D produktów.

Warstwa 5: Sesji / Interoperacyjności (Interoperability)

- **Narzędzia:** Apache Hop, konektory do ePlan i SolidWorks.
- **Funkcja:** Jednolity protokół wymiany danych. Przekształcanie danych CAD w struktury grafowe.

Warstwa 4: Zarządzania Zasobami (Resource Management)

- **Narzędzia:** Apache OFBiz + Akeneo PIM.
- **Funkcja:** Zarządzanie fizyczną realizacją. Powiązanie „złotego standardu” danych (Akeneo) z transakcjami (OFBiz).

Warstwa 3: Topologii Sieci (Network Topology) — JĄDRO SYSTEMU

- **Narzędzia:** Neo4j.
- **Funkcja:** Implementacja standardów ISO 81346 i IEC 61355. Przechowywanie Cyfrowego Bliźniaka (Digital Twin) projektu.

Warstwy 1-2: Infrastruktury (Physical/Data Link)

- **Stos:** Ubuntu/Debian, Docker, Kubernetes.
- **Funkcja:** Zapewnienie wysokiej dostępności i skalowalności zasobów.

4. Integracja z CAD: ePlan i SolidWorks

Mimo że systemy CAD są historycznie powiązane z Windows, w architekturze CIEE są one integrowane jako **źródła danych pierwotnych** (Data Sources) dla jądra Linux.

4.1. ePlan (Elektrotechnika i P&ID)

ePlan jest źródłem struktury funkcjonalnej i logicznej projektu.

- **Jak się łączy:** Poprzez standard **DEXPI (XML)** lub zautomatyzowany eksport do JSON/XML.
- **Warstwa integracji:** Layer 5 (Interoperacyjność).
- **Wpływ na graf:** Dane z ePlan tworzą w Neo4j **aspekt funkcjonalny (-)** oraz **aspekt lokalizacji (+)** zgodnie z ISO 81346.
- **Automatyzacja:** Apache Hop monitoruje folder projektu ePlan i przy każdym zapisie aktualizuje powiązania między czujnikami, zaciskami i sterownikami w grafie Neo4j.

4.2. SolidWorks (Mechanika i Konstrukcja)

SolidWorks jest źródłem struktury fizycznej produktu (BOM).

- **Jak się łączy:** Poprzez eksport makr (SW API) lub zapis do neutralnych formatów (STEP, JSON BOM).
- **Warstwa integracji:** Layer 5 i Layer 4.

- **Wpływ na graf:** Dane tworzą **aspekt produktowy (/)**. Geometria (STEP) jest przekazywana do Akeneo PIM w celu wizualizacji na portalu OpenCms.
- **Cyfrowy wątek (Digital Thread):** Zmiana materiału lub masy części w SolidWorks natychmiast aktualizuje atrybuty odpowiedniego węzła w Neo4j i przelicza koszty w OFBiz.

5. Głęboka integracja standardów: ISO 81346 i MBSE

System implementuje standard **ISO 81346** poprzez podział odpowiedzialności:

1. **Aspekt funkcjonalny (-): Eclipse Capella & ePlan.** Koncepcja (Capella) jest uszczegóławiana przez schematy (ePlan) i przekształcana w sieć logiczną w Neo4j.
2. **Aspekt produktowy (/): SolidWorks, Akeneo PIM & OFBiz.** Konkretnie części z CAD są powiązane z kartami w PIM i stanami magazynowymi.
3. **Aspekt lokalizacji (+): Neo4j.** Hierarchia miejsc montażu (Hala -> Szafa -> Panel), zaimportowana z ePlan.

6. Hybrydowa strategia AI i orkiestracja agentowa

Wykorzystanie dwóch modeli AI pozwala na podział zadań:

- **Obieg lokalny (Ollama):** Sprawdza poprawność nazewnictwa węzłów w ePlan/SolidWorks zgodnie z wewnętrznymi standardami przedsiębiorstwa.
- **Obieg globalny (Google Gemini):** Analizuje złożenie 3D z SolidWorks i logikę ePlan pod kątem kolizji inżynierskich (np. niedopasowanie mocy silnika do wybranego w ePlan rozrusznika).
- **Agenci AI (Node-RED):**
 - *Agent Synchronizujący:* Pilnuje, aby skład produktu w SolidWorks nie odbiegał od schematu w ePlan. W przypadku wykrycia rozbieżności — tworzy zadanie w **OpenProject**.

7. Sieciowe zarządzanie projektami (OpenProject + Neo4j)

Zarządzanie projektami jest zintegrowane z topologią:

- Zadania w **OpenProject** są tworzone automatycznie podczas importu nowego złożenia z SolidWorks.
- **Ścieżka krytyczna:** Jest obliczana w Neo4j z uwzględnieniem gotowości węzłów inżynierskich.

8. Metoda "Document-to-Graph" i OpenCms

Dokumentacja jest generowana automatycznie:

1. **OpenCms** pobiera strukturę projektu z Neo4j.
2. Dołącza schematy z **ePlan** i rysunki 3D z **SolidWorks**.

3. AI (Gemini) przygotowuje opis techniczny, wyjaśniając, dlaczego wybrano konkretne rozwiązania.

9. Analiza SWOT architektury Open Source na Linuxie

Mocne strony (Strengths)

Zero License Cost: Brak kosztów oprogramowania serwerowego.

Hybrid AI: Połączenie prywatności i mocy obliczeniowej.

Digital Thread: Bezpośrednie połączenie od rysunku w ePlan do zakupu.

Słabe strony (Weaknesses)

CAD Dependency: Inżynierowie nadal potrzebują stacji Windows dla ePlan/SolidWorks.

Złożoność konfiguracji: Wymagane niestandardowe mapowanie danych z CAD do grafu.

Aktualizacje: Należy utrzymywać parsery przy wydaniu nowych wersji CAD.

10. Scenariusz wdrożenia (Deployment Roadmap)

1. **Infrastruktura:** Stos Docker na Linux (Ubuntu).
2. **Konektory CAD:** Konfiguracja Apache Hop do odbierania danych XML/JSON z ePlan i SolidWorks.
3. **Konfiguracja PIM:** Synchronizacja Akeneo z bibliotekami komponentów ePlan.
4. **Trening AI:** Konfiguracja lokalnej Ollamy do pracy z terminologią inżynierską przedsiębiorstwa.

11. Linki do technologii i komponentów

- **Neo4j** : Grafowa baza danych.
- **Akeneo PIM** : Zarządzanie informacją o produkcie.
- **Eclipse Capella** : Narzędzie MBSE.
- **OpenProject** : Zarządzanie projektami.
- **Apache OFBiz** : Jądro ERP.
- **Node-RED** : Orkiestrator przepływów.
- **Apache Hop** : ETL i integracja.
- **DEXPI** : Standard wymiany danych dla ePlan.

12. Podsumowanie

Integracja systemów **ePlan** i **SolidWorks** z ekosystemem Linux CIEE przekształca go w pełnoprawny kompleks PLM. Nie próbujemy przenosić systemów CAD na Linuxa — przenosimy **dane**, które one generują, do grafowego jądra wiedzy. Pozwala to na zarządzanie projektem na poziomie znaczeń i powiązań, a nie na poziomie plików i folderów.