An Architecture and Development Process based Enterprise Transformation Projects (ADPbETP)

ANTOINE TRAD
Transformation, IBISTM,
Courbevoie,
FRANCE

Abstract: - Enterprise Transformation Projects (ETP) are critical for a long-term enterprise's sustainability progress, but such projects are chaotic, complex, and can fail. Transformation complexities are related to various concurrent factors like the use of sets of uncoordinated Architecture and Development Processes' (ADP) related tools/environments/products, simplistic gap estimations, statuses' evaluations, needed Polymathic skills, and many others. Therefore, there is a need to implement an In-House Implemented (IHI) concept, methodology, and framework to support ADPs and ETPs. But such IHI solutions take a long time to be finalized and this article tries to propose a realistic solution that is based on Relational DataBases (RDB) fundaments, and ADPs. ADP-based IHI solutions and concepts can be built iteratively on the existing enterprise's legacy information system; without the need for unneeded major investments for financing external ADPs-oriented products. The proposed RBD-based ADP concept tries to show that it can support any type of ETP because the ADP is a generic concept that can be used for ETP activities like: Implementation, design, development, operations, and integration. ADPs offer the needed architecture methodologies, development concepts, information management, (re)structuring capacities, integrity checking, and mathematical constructs. The proposed ADPbETP concept adopts a Polymathic-holistic approach, which uses iterative design, change, and implementation phases. The ADPbETP uses the author's Applied Holistic Mathematical Model (AHMM) to interface and manage the ADP (AHMM4ADP). This research paper uses the author's transformation framework, terms, and related resources.

Key-Words: - ADP, RDB, ETP, Integrity Checks, Refinement, Enterprise Architecture, Development, Decision Making

Received: April 16, 2022. Revised: April 14, 2023. Accepted: May 15, 2023. Published: June 14, 2023.

1 Introduction

The enterprise's (simply the *Entity*) Holistic Transformation and Refinement Processes (EHRP) are used to convert the legacy Information and Communication System (ICS) to an integrated, coherent, and automated ADPbETP. An EHRP is a sequence of: Components' conversion, RDB-based integrity checking, extraction, and development operations that are done on various parts of the ICS.

Fig. 1. The relation between the Ehrp, RDB, and ADP.

An *Entity* is a set of organizational Units (simply the *Unit*), where each *Unit* has one or more *Unit* Platform(s) (UnP). An EHRP on the *Unit's* level refines and transforms its structure and its UnP(s); and the AHMM4ADP checks each iteration's integrity and robustness, by using existing RDB-based ADP constructs, as shown in Fig. 1, [1], [2], [3]. An ADPbETP can be applied to any APplication Domain (APD), *Units*, and all related ADP activities. *Entity's* ADP functions are EHRPed into Building Blocks (BB) and Architecture BBs (ABB) which can be reused to convert ICS' components, *Units*, and *UnPs*. *Units* are then (re)assembled and checked by the AHMM4ADP, to deliver a transformed ICS and a new form of an *Entity*. As already mentioned, ETPs are complex and they depend on EHRPs' successful terminations, [3]. To integrate the EHRP with the ICSs, *Units*, and *UnPs*, the first step is to: 1) Establish an IHI Methodology, Domain, and Technology Common Artefacts Standard (MDTCAS); 2) Select and install an Etalon RDB (ERDB) to abstract existing DBs and their mappings to existing ICS components, methodologies or resources; and 3) Design a generic ADP. The IHI

MDTCAS interfaces and manages EHRP's basic elements, which are: BBs, ABBs, Solution BBs (SBB), Composite BBs (CBB), Organizational BBs (OBB), and Micro-Artifacts (MA) (simply *Artefact).* The ERDB-based ADP is used to map the AHMM4ADP to ICS' sections and components like: 1) Networks and nodes; 2) Various types of DBs and data sources; 3) Applications, software components, and libraries; 3) Methodologies, like the: Unified Modelling Language (UML), Archimate language, Object Oriented Methodology (OOM), and other; 4) Interfaces, Gateways, Application Programming Interfaces (API), and other; 5) Processes, Scenarios, Transactions, and other; 6) Security, Governance, Audit, and other; 7) Actors, delimiters, or other; 8) Decision-Making System (DMS), Knowledge Management System (KMS), or other; 9) Control, Monitoring, Tracing, or other; 10) Applications and data services; and any other ICS' part. 11) *Artefacts* can be (re)used in standardized, external, or IHI *Unit's* Process/collaboration Models (UPM); 12) The integration of Development and Operations (DevOps) and agility concepts; 13) The usage of Project Management (PM) and Audit disciplines; and 14) APIzation of the ICS.

Fig. 2. ADPbETP's evaluation phases.

As shown in Fig. 2, ETP's success depends on *Entity's* structure, which is in general siloed and that makes ETPs complex and risky, because of various constraints and reasons, which are related to EHRPs' and ERDB's feasibilities. An ETP has various Viewpoints, like: "O" for organizational, "S" for Security, "F" for Financial, "I" for Integrity checking, "A" for ADP's Viewpoint… The ADPbETP is mainly a set of architecture and development transformation processes, and adopts primarily Viewpoint "A"; and Viewpoint "I" as a second objective. To prove this Research and Development Project's (RDP) and ADPbETPs feasibility, the author implements the RDP for ETP (RDP4ETP) and his Proof of Concept (PoC).

2 The RDP4ETP

2.1 A Polymathic Model

The ADPbETP needs the right strategy and risks mitigation mechanism to guarantee ADP operations' coherency. Coherency is guaranteed by the AHMM4ADP. A specific ETP requirement (or problem type), the AHMM4ADP-based DMS proposes the initial sets of Critical Success Factors (CSF), Critical Success Areas (CSA), and Key Performance Indicators (KPI) (simply *Factors*), to be applied by DMS' Heuristics Decision Tree (HDT). The Polymathic RDP4ETP maps the selected *Factors* to requirements and the EHRP generated sets of *Artefacts*, to support the ADP; all RDP4ETP steps are shown in Fig. 3. *Artefacts* support the ETP where the emphasis is on Viewpoint "A".

Fig. 3. The Polymathic RDP4ETP.

RDP4ETP's first action is to define the Research Question (RQ) and achieve an in-depth Literature Review Process (LRP) for the ADPbETP (LRP4ADP).

2.2 The RQ and LRP4ADP

The RDP4ETP's RQ is: "Is the ADPbETP capable to support *Entity's* transformation project and to deliver architecture and development processes' mechanisms?". Where this article's auxiliary RQ is: "How can the ADPbETP support the synchronization of various ETP domains by using the ERDB?". Knowing that the RDP4ETP uses Enterprise Architecture (EA), DevOps, ERDB concepts, AHMM4ADP, Transformation Research Architecture Development framework (*TRADf*), and DMS. *TRADf* is a sample that shows how an *Entity* implements an IHI framework. The ADPbETP implements an adapted version of EA, which is the

Transformation Development Method (TDM). LRP4ADP's analysis showed that are no similar approaches that use: the IHI transformation framework (like *TRADf*), EHRP generated *Artefacts*, AHMM4ADP, Polymathic RDP4ETP and there is a small number of relevant scholarly resources that are related only to basic ERDB-based ADP implementation concepts and especially topics related to EA. Therefore, the ADPbETP-related works, are pioneering, innovative, and cover an important ADPbETP gap. ETP-related gaps and high failure rates were confirmed by the LRP4ADP, [4]. There is an immense lack of a Polymathic-holistic approach to ADPbETPs operations which can be done on ICS parts and components. The LRP4ADP uses the following resources: 1) Articles and resources related to ADP, DevOps, ERDBs, EHRP, ICS reengineering, and ETPs; 2) TRADf-related previous RDP/LRP works; 3) ETP's feasibility; 4) Selected sets of CSAs/CSFs; and 5) RDP4ETP's use of the Empirical Engineering Research Model (EERM). The RDP4ETP found an important gap and the need to deliver ADPbETP solutions and recommendations. The gap is that there is nothing similar to the proposed approach; but there are some basic approaches that concern exclusively ADPs. The main difference is that this work adopts a Polymathic approach; unfortunately, such approaches are considered to be not serious, but in reality, it is the main problem for such complex transformations.

2.3 The EERM and RDP4ETP Phases

RDP4ETPS' phases are: 1) Phase 1 (represented in decision-Tables), represents empirical RDP4ETP's section; which checks this article's CSAs, which are: a) The RDP4ETP, which is synthesized in Table I; b) The ADPbETP's initial setup, which is synthesized in Table II; c) The EADP's integration, which is synthesized in Table III; d) ERDB's specific solutions, is summarized in Table IV; and f) This article's RDP4ETP outcome, which is synthesized in Table V. *TRADf* supports the ADPbETP to be finalized and the RDP4ETP to propose a list of managerial, architecture, and technical recommendations and solutions, and an adapted strategy; and 2) Phase 2, which solves a concrete ADPbETP problem, by the use of the of *TRADf* and HDT. RDP4ETP's usage of EERM is optimal because it applies a multi-level mixed-research by using the HDT; which is very different from conventional research models, and it includes, [5], [6]: 1) Heuristics-oriented processing and reasoning; 2) Quantitative Methods for ETP (QNT4ETP); 3) Qualitative Methods for ETP (QLT4ETP) mixed research concepts and methodologies, to deliver concrete patterns/concepts as an alternative concept for RDP4ETP mixed-methods; and 4) An HDT based Learning Process, which was mainly inspired by Action Research (AR) learning processes.

2.4 The AHMM4ADP

AHMM4ADP's main concept and elements are presented in a simple form to be easily understood. The ADPbETP uses the AHMM4UP that has the following structure:

- ICS Unbundling actions = supports EHRP operations, Implementation activities, and finalizing the UnPs.
- ETP parts = ∑ UnP (implemented by the ICS, *Artefacts*, and the infrastructure/networks).
- ADP4(Categories) = Transformation of ETP's parts + the objectives of the ETP.
- ADPbETP(Iteration) = Uses ETP's parts + ∑ RDB4(Categories).
- AHMM4ADP(APD) = ∑ ADPbETP(n).
- TDM(APD) = TDM + AHMM4ADP(APD).
- ETP = TDM(APD) + GapAnalysis(Iteration).

2.5 The RDP4ETP Factors' Evaluations

Table 1. This CSA's average is (rounded) 9

Critical Success Factors	KPIs	Weightings
CSF_RDP4ETP_Polymathic_Approach	Proven	From 1 to 10. **10 Selected**
CSF_RDP4ETP_Factors_Integration	Proven	From 1 to 10. **10 Selected**
CSF_RDP4ETP_EHRP_Integration	Complex	From 1 to 10. **08 Selected**
CSF_RDP4ETP_EERM	Feasible	From 1 to 10. **09 Selected**
CSF_RDP4ETP_AHMM4ADP_Usage	Feasible	From 1 to 10. **09 Selected**
CSF_RDP4ETP_ADP_Mechanisms	Feasible	From 1 to 10. **10 Selected**
CSF_RDP4ETP_IHI_Framework_TRADf	Possible	From 1 to 10. **09 Selected**
CSF_RDP4ETP_LTR4ETP	Proven	From 1 to 10. **10 Selected**

valuation

Based on the AHMM4ADP, LRP4ADP, and DMS, for this CSA's CSFs/KPI were weighted and the results are shown in Table I. This CSA's result of (rounded) 9.5, which is very high, is due to the fact that the RDB-based ADP simplifies the RDP4ETP and it is possible to be implemented. And the next step is the ADPbETP's initial setup.

3 ADPbETP's Initial Setup

ADPbETP's initial setup understands the following steps: 1) That EHRP's processes were successful and that generated *Artefacts* are ready to be used; 2) To define and implement an IHI MDTCAS; 3) Apply an EA and an Architecture Development Method (ADM) based TDM, [7]; 4) Setting up an ERDB to support transformation operations, which can be any type of RDB or a software application; and 5) To set-up an ADP based on the ERDB, MDTCAS, and DevOps.

3.1 EHRP Processes Successful Termination

ADPbETPs depend on the critical EHRPs-based Unbundling Process (UP). The *Entity's* UP, which is a set of EHRPs, disassembles its Legacy *Units*' structures, System's administration, Resources, Applications, UPMs, Working models, and components; into dynamic reusable *Artefacts* especially OBBs, [8]. EHRPs face difficulties because of the *Entity's* heterogenous human profiles/cultures/skills, ICS parts, exaggerated managers' financial ambitions, ADP/MDTCAS complexities, and ETP's limited time/budgets. Another major problem is that transformation and innovation technics have been monopolized for achieving only immediate tangible goals like business and financial aspects/profits, where the intangible complex technical aspects are simply ignored. Such approaches generate major ADP and ETP issues and high rates of failures that today are above seventy percent. It is important to define ADP's and ETP's levels of granularity and mapping concepts for each MDTCAS' elements' application. These facts enable the reuse of existing or newly generated *Artefacts*. After the successful UP (and its EHRPs' terminations, which is a major achievement), the ETP can move to the next phase, [8]. Refined *Artefacts* that are used in UPMs support *Entities* work by: 1) Visualizing operating and support activities; 2) Showing how employees report to higher Managers and how UPM based *Units* are transformed ; 3) Fixing goals that bring together employees with *Entity's* objectives; 4) Supporting interfaces (interactions) between *Units*; 5) Using the ERDB for integrity checking operations; 6) Integrating ADPs in the ETP; 7) MDTCAS' Implementation ; and 6) Restructuring *Units'* operations. MDTCAS usage by the ADP needs:

- To build an IHI framework to support: ETP, MDTCAS, and the ADM-based TDM, [7], [9].
- The ADPbETP breaks down *Entity's* monolithic silos. ETPs use the TDM and MDTCAS to model APD and ICS models and to define their scopes, [7]. The TDM synchronizes ETP's activities; and UP (and its EHRPs) are difficult to scope because they depend on the APD's and MDTCAS' incorporation capacities. The ADP supports ETPs for (re)organization operations, which enhance functional performances. An *Artefacts*-based UPM that can be used in APD models' development, which needs a Polymathic-holistic approach to transform Legacy *Units*, [9], [10].
- The ADPbETP main domain is EA and therefore the MDTCAS and generated *Artefacts* mainly (ABBs and SBBs) are critical. The ADPbETP synchronizes and manages the IHI framework and methodology that can map to any existing methodology, development concept, or technology. Where the MDTCAS manages EHRP-generated *Artefacts* and is transcendent MDTCAS-based ADP. That ensures that ETP's evolution is independent of all EA and DevOps (or other) domains and technological hyper-evolutions. MDTCAS' usage is an important factor for the success of ETPs and it unifies *Artefacts'* management.
- The AHMM4ADP supports Polymathic UPs and their embedded EHRPs to transform the *Entity's* legacy systems, by using the MDTCAS and TDM to integrate standard methodologies, like The Open Group's (TOG) Architecture Framework (TOGAF) and its ADM, [7]; where the TDM enhances and abstracts the ADM. The MDTCAS is a minimal and transcendent mix of existing methodologies, development technics, and practices like: 1) ERDB elements: Entity Relational Diagram (ERD), Entity Relational Modelling (ERM), Object Relational Modelling (ORM)… 2) Development methods like: OOM, UML, legacy methodologies (like the Structure Analysis and Structured Design-SA/SD)… 3) EA methodologies, like: TOGAF/ADM, Archimate, Decision Making Notation (DMN), and other; and 4) IHI Framework, like *TRADf* that integrates: MDTCAS/TDM.
- The MDTCAS/TDM supports the conversion initiatives in the following cases: 1) ERM, ERD, ORM models; 2) Mainframe legacy subsystems conversion to SA/SD models ; 3) Restructures to corresponding OOM/UML Entity-Classes mapping or ERD2CLS, which corresponds to ORM, which is today automated; 4) To transform existing OOM/UML models/diagrams based components into well-designed/mapped UML/Choreography models, using classes, sequences, communication models; 5) ERM/ERD/ORM, and UPMs/Business Processes (BP) and their Models (BPM) diagrams; 6) Implement a light-version of Spiraled/UML, TOGAF, and ADM based TDM development cycles to support the ADM; 7) Recycle processes

into *Artefacts* and especially ABBs and SBBs; and 8) Adopt basic DMN like elements, such as requirements diagrams and Tables' evaluations that are done by the DMS. For all mentioned methodologies, the OOM/ERD or the ERD2CLS is central for the MDTCAS and *Artefacts*, which is shown in Fig. 4.

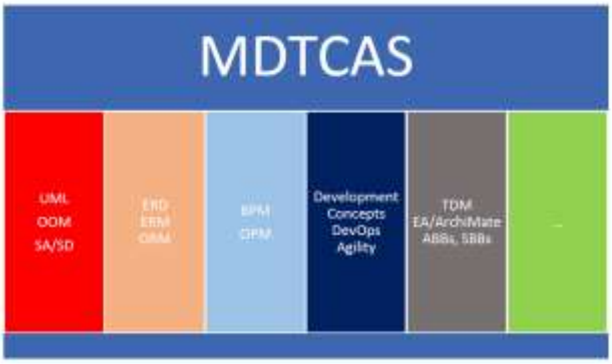


Fig. 4. The IHI MDTCAS

3.2 Adapting ERDB-based EA/ADM and TDM

The ERDB supports the integrity and evolution of ADPbETP operations, like the case of the generation of *Artefacts* like the ABBs, and it offers EA's capacities to Assess Readiness for ETP (AR4ETP). The AR4ETP estimated *Entity's* transformation readiness to achieve the ETP. ETP's transformation assessment is ensured by the TDM and *TRADf*. The results of *TRADf's* the mentioned assessments are added to ETP's capabilities to achieve complex assessments and risk mitigations. These ADP-associated risks are linked to ADPbETP's vision and the initial levels of risks. *Factors*-based risk mitigation controls are integrated by using the TDM and HDT, which enables ADPbETP's automation and implementation activities. The TDM applies phases' iterations, where ETP's actions are traced and logged in the RDB-based ADP. The ETP must be methodology and technology agnostic for the APD. TDM's integration in the ADPbETP offers the following capabilities: 1) Real-time ERDB-based integrity checks, mapping, and *Artefacts'* management, and especially ABBs and SBBs; 2) Improves ICS' capabilities; 3) Integrates The use of existing and standard frameworks/methodologies like ERM/UML (or ArchiMate); 4) The adoption of agile development and test concepts like DevOps; and 5) The use of tests and an integration-driven developments approach. The Any EA methodology and the TDM map to resulting ETP's applications' cartography. The transformed and classified applications ensure that: 1) Applied classification and transformation processes are done by EA technics and capacities, such as TOGAF's Application Communication Diagram (ACD), that abstracts and presents results in models and mappings, supports the interaction/communications intra-applications and linked libraries/modules; that results in an ETP pattern and related *Entity's* metamodel. Such a pattern and meta-model presents applications, libraries/components, and interfaces (between various ETP and external components); 2) All types of ETP interfaces are linked to data structures/classes, applications/modules and in turn linked to *Artefacts*; 3) ACDs show and depicts ETP and *Entity's* applications in the form of a cartography, or a logical model of a future architecture of an end-system. *Artifacts* and EA-based ETP are recommended; 4) *Entities* have in general hybrid ICSs and heterogenous components/applications, which need an ERDB-based enterprise-wide repository and new *Artefacts*-based ADP(s); 5) In the case of using *Artefacts (*and *ABB/SBBs)* based application/components, are restructured to support EA models, constructs, and patterns; 6) *Artefacts* based applications/components are linked to services, which use bus-connectors to integrate ABBs; 7) ACD based cartography enables dynamic TDM's integration; 8) The ERDB based ADP refined and delivers ETP's ACD based cartographies; and 9) As shown in Fig. 5, the applied EA based TDM concept is a layered one, where the focus is on: The interaction component layer that is on top, Process-based components in the middle, and The *Entity* components are in the bottom layer. The EA-based TDM integrates existing architecture and design standards, like TOGAF, as shown in Fig. 5. *TRADf* is enabled to integrate TOGAF (by using the IHI MDTCAS and ADP) and applying the optimal just-enough EA approach, where the following layers are used: 1) Business Architecture; 2) Data Architecture; 3) Application Architecture; and 4) Technology Architecture, [7], [9].

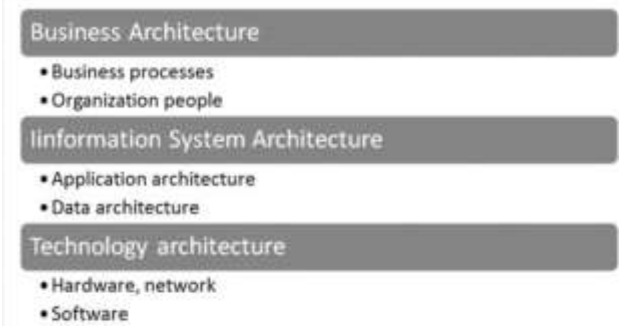


Fig. 5. The used EA concept is layered

3.3 The IHI Methodology-TRADf and the Role of the ERDB

The ADPbETP and RDP4RDP need an IHI Methodology and Framework, like *TRADf*, which an *Entity* can implement, and should avoid expensive products. ERDB's integrity rules are optimal for the ADPbETP, [11], [12], and the AHMM4ADP uses these rules. Many concepts can be used to unify the ERDB by using the Extract, Transform, and Load (ETL) or Enterprise Service Bus (ESB) which can be interfaced by any ADP or ICS, [13]. Where the ADP persistence repositories are used to store various types of models. Models use standard formats like the ArchiMate Model Exchange File Format (AMEFF), [14], [15].

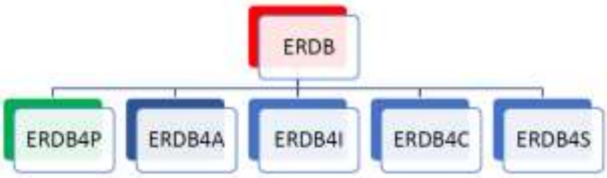


Fig. 6. The ERDB

As shown in Fig. 6, the ERDB is used to abstract and interface/map the following ICS, ADP, and standard methodologies categories: 1) ERDB for Platforms (ERDB 4P), which includes: Networks, DBs, and nodes; 2) ERDB for Applications (ERDB4A), which includes: Applications, Software (components and libraries), BPs, Transactions, ADP/ABB/SBB, Methodologies (like UML, Archimate, OOM, or other); 3) ERDB for Interfaces (ERDB4I): Interfaces, Gateways, API, Actors, Delimiters, and other; 4) ERDB for Control (ERDBC): Security, Governance, Audit, Monitoring, Tracing, or other; 5) ERDB for aGility (ERDB4G): DevOps, DevSecOps, Agility, or other; 6) ERDB for intelligent Systems (ERDB4S): like DMS, KMS, BPM based systems, or other; 7) ERDB for change Management (ERDB4M): Project management, Audit, or other; and 8) ERDB for APIzation (ERDB4Z) of the ICS and ADP.

3.4 EADPbETP's Initial Setup Factors' Evaluations

Table 2. This CSA's average is 8.70.

Critical Success Factors	AHMM4CBB enhances: KPIs	Weightings
CSF_Initial_Setup_EHRP_Termination	Complex	From 1 to 10. 08 Selected
CSF_Initial_Setup_MDTCAS_Implementation	Feasible	From 1 to 10. 09 Selected
CSF_Initial_Setup_TDM/ADM/EA	Feasible	From 1 to 10. 09 Selected
CSF_Initial_Setup_IHI_Framework	Feasible	From 1 to 10. 09 Selected
CSF_Initial_Setup_ERDB_Setup	Feasible	From 1 to 10. 09 Selected
CSF_Initial_Setup_ERDB-based-ADP_Setup	Complex	From 1 to 10. 08 Selected

valuation

Based on the AHMM4ADP, LRP4ADP, and DMS, for this CSA's CSFs/KPI were weighted and the results are shown in Table II. This CSA's result of 8.70, which is in a limit zone, is because EHPRs are very complex and RDP4ETP's initial setup is a difficult phase. And the next step is to establish an Etalon ADP's (EADP) integration with the ETP (EADPbETP).

4 EADP's Integration

4.1 Setting up the Basic ERDB for EADP

The basic ERDB capacities serve the EADPbETP by reflecting the status of the progress of the transformation of various ADP and ICS modules, patterns, *Artefacts*, and components or elements like the:

- ERDB4P manages the following ICS components: Networks, DBs, and platform nodes. The ERDB4P (or the integrated platform DB) is the basis of an ICS integrated Network Management System. The platform DB is distributed over the ICS and can be an RDB or any other type of DB. The persistent information related to ICS' components and networks, is needed for the proper operation of the ICS, [16], [17]. The ERDB4P supports and persists the EADP's EA modeling views like ArchiMate's Technology Platform View (Infrastructure View), [18], which is shown in Fig. 7.

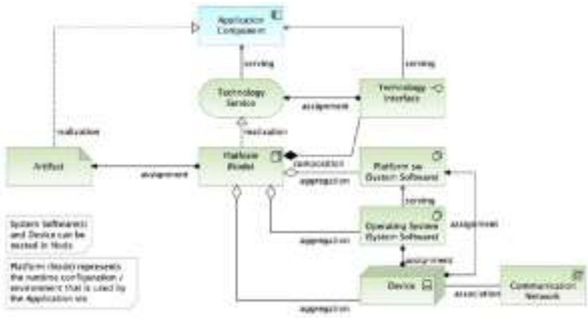


Fig. 7. The Technology Platform View - Design Pattern

- The ERDB4I serves the EADP by reflecting the status of the progress of the transformation of various ICS components or elements, like: Interfaces, Gateways, APIs, Actors, Delimiters, and others. The ERDB4I uses also the two already mentioned DB concepts by using API mechanisms. The ERDB4I supports and persists the EADP's EA modelling views like ArchiMate's Business Actor Map, [18].

- The ERDB4C serves the ETP by evaluating the status of the progress of the transformation of various ICS components or elements like: Security, Governance, Audit, Monitoring, Tracing, and others. The ADPbETP uses EA and TDM, which facilitate the integration of the Sherwood Applied Business Security Architecture (SABSA), [12]. That supports security, which depends on *Entities* and the selected CSFs, and the established sets of best practices that can influence the ERDB4C, [19]. The use of control frameworks, like SABSA, facilitates ERDB4C interfacing. The ERDB4C uses two already mentioned DB concepts. Concerning the ADP, the models can be mapped to MDTCAS equivalents and kept in a repository. The ERDB4I supports and persists the EADP's EA modelling views like ArchiMate's Risk and Security View Pattern, [18].
- The ERDB4S supports the EADP by reflecting the status of the progress of the transformation of various systems like: DMS, KMS, BPM-based systems (BPMS), and others. The BPMS manages tasks and processes related to the ICS; and it includes: 1) A process designer and implementer; 2) A process engine that manages BP tasks; 3) Data management tools; and 4) A reporting engine for monitoring BP activities. The BPMS supports BPMs' implementation by the ICS team(s), [20]. The ERDB4S uses two already mentioned DB concepts. Concerning the ADP, the models can be mapped to MDTCAS equivalents and kept in a repository. The ERDB4S supports and persists the EADP's EA modelling views like ArchiMate's Relation of the Value Stream and the Business Process, [18].

4.2 Integrating Advanced ERDB for EADP

The advanced ERDB capacities serve the ETP by reflecting the status of the progress of the transformation of various EADP modules, patterns, *Artefacts*, and components or elements like the:

- The ERDB4G supports the EADP by reflecting the status of the progress of the transformation of various systems like: DevOps, Agility, and others. DevOps activities, which emphasize the collaboration of ETP development and operations, infrastructure is supported by software engineering and BPs. The ERDB4G uses two already mentioned DB concepts. Concerning the EADP, the models can be mapped to MDTCAS equivalents and kept in a repository. The ERDB4G supports and persists the EADP's DevOps views like the EADP model maintenance deployment pipeline, [21].
- The ERDB4M supports the EADP by reflecting the stats of the progress of the transformation of various systems like: Project management, Audit, and others. The EADP can be used to model the management of ETP changes. The EADP of the migration process from a legacy construct (EA's current state) to a future situation (EA's target state) can have important consequences on the *Entity*. Where the EADP takes into account: 1) Implementing *Entity*-wide EA; 2) ICS remains operational through the ETP; and 3) The used ICS components can be unstable and heterogeneous. The ERDB4M uses two already mentioned DB concepts. Concerning the EADP, the models can be mapped to MDTCAS equivalents and kept in a repository. The ERDB4G supports and persists the EADP's Project Viewpoint, [18], [22].
- The ERDB4Z supports the EADP by reflecting the status of the progress of the transformation of various systems like: API platforms, Specific interfaces, and others. API platforms benefit ADPbETPs by offering centralizing control of API pools and ensuring that they are continuously secured and available. Concerning architecture and modeling resultant interface elements can be mapped to MDTCAS equivalents and kept in the ETP central repository. The EADP supports the API Viewpoint Modelling, which integrates ArchiMate and UML, [23], [24].
- The ERDB4A serves the ETP by reflecting the status and progress of the transformation of various ICS components or elements like: Applications, Libraries, BPs/Transactions, Methodologies, and others. The ERDB4A as shown in Fig. 8, uses two DB concepts, which are 1) Classical Read, Write, Update, and Delete (RWUD) operations, which are standard data access operations, and all ERDB categories' elements use the RWUD operations; and 2) Modelling and architecture activities, where the resultants diagrams, models and other are stored in case tools DBs. For these two DB concepts, the ETP can use the ETL or any other DB unification and integration concepts. Concerning architecture and modeling resultant elements can be mapped to MDTCAS equivalents and kept in a specific repository.

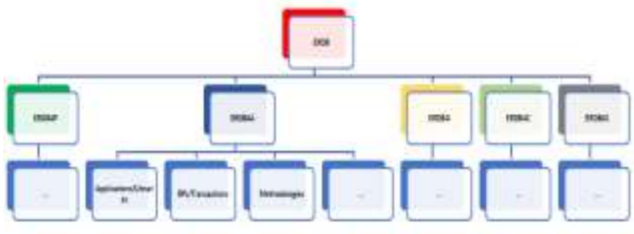


Fig. 8. The ERDB4A structure

The EADP Integration Factors' Evaluations

Table 2. This CSA's average is 9.0.

Critical Success Factors	AHMM4CBB: KPIs	Weightings
CSF_ERDB_based_EADP_Integration_Setup	Feasible	From 1 to 10: 09 Selected
CSF_ERDB_based_EADP_RDB_Mechanisms	Feasible	From 1 to 10: 09 Selected
CSF_ERDB_based_EADP_Basic_Capacities	Feasible	From 1 to 10: 09 Selected
CSF_ERDB_based_EADP_Advanced_Capacities	Complex	From 1 to 10: 08 Selected

valuation

Based on the AHMM4ADP, LRP4ADP, and DMS, for this CSA's CSFs/KPI were weighted and the results are shown in Table III. This CSA's result of 8.75, which is in high result, and is due to the fact that the EADP facilitates the integration and that is a feasible phase. And the next step is to analyse EADP's specific solutions.

5 EADP's Specific Solutions

5.1 Setting up Factors

Fig. 9. The EADPbETP Factors management.

As already mentioned, an *Entity's* (or ETP) CSA is a category (or set) of CSFs that are selected by the ETP implementation team, as shown in Fig. 9. A CSF is a set of KPI, where a KPI corresponds to an ETP requirement and/or feature. A KPI can be related to an EADP module (or application) concrete variable or ERD attribute. For a concrete EADPbETP requirement or problem, the team identifies the initial sets of Factors, to be used by the HDT-based DMS. The Factors map to the EHRP-generated sets of *Artefacts*, which include ABBs/SBBs. Hence the CSFs are important for the mapping between the requirements, knowledge constructs, *Artefacts*, patterns, ABBs, OBBs, Units, and DMS, [25]. Therefore, Factors support strategic EADPbETP goals; which need measurement technics, that are provided by *TRADf*, can be used to evaluate each CSA's performance, where CSFs can relate to: 1) EADP's status; 2) Mapping levels of EADP's *Artefacts*; 3) Gap analysis; 4) TDM phase's integrity; and 5) DMS' requests. KPIs can be integrated into *Artefacts*, so HDT's based evaluation processes can automatically estimate the values of Factors, [6]. Factors are tuned by the ETP team by using the EADPbETP and they are weighted by the DMS, in-order to offer possible solutions for a given EADPbETP problem.

5.2 Using Entity Logging Mechanism

An *Entity* can implement an enterprise-wide Log Server (LS) for EADP (LS4EADP), to support persistence, Gap analysis, monitoring, diagnosing, and troubleshooting activities.

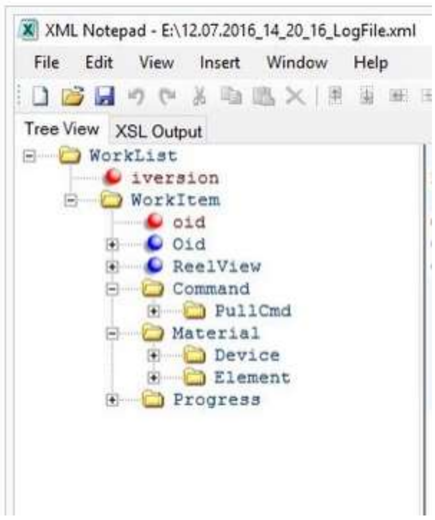


Fig. 10. XML log file, [17], [26]

Such activities are key activities for the *Entity's* ETP and TDM lifecycles, and logging is the core part of these logging activities. EADP module components deliver messages to the LS4EADP. The LS4EADP sends logs to various ICS' RDB destinations that connect to all nodes. An eXtensible Markup Language (XML) format is used for LS operations. As shown in Fig 10, the XML record includes: Timestamp, Nodes' settings, Status, Files/directories info, and Errors/warnings info. The LS4EADP can support the sending to various destinations and that needs the implementation of an IHI LS framework, [26], [27]. LS' operations can be improved by using atomic *Artefacts* (*aArtefacts*) based on atomic BBs (aBB).

5.3 aBBs' Integration

aBBs resources the TDM, wherein each phase they are refined and that gives the EADPbETP a unified view of the development cycles that consists of: 1) Unified pool of aBBs; 2) aBBs based EADP loose modules; 3) Unique identifier; 4) Optimal level of granularity; and 3) A scalable and secured ICS infrastructure. aBBs' integration needs an optimal architecture concept that needs: 1) EA business architecture pattern, 2) TDM-based MDTCAS; 3) Anti-locked-in tools; and 4). EADPbETP uses aBBs, and atomic Solution Blocks (aSB) to be used by optimized tooling and modelling environments, which are based on the Model View Control (MVC) pattern. There is a need for an atomic Architecture Vision (aAV) to support aBBs. aAV's is crucial importance for TDM's implementation phases and also for operations activities which comes after the finalization of the implementation phase. The aAV manages the implementation of autonomous aBB-based transactions, [28], [29]. aBBs are assembled in ABBs, aSBs, and SBBs.

5.4 ABBs and SBBs

The EADPbETP uses standards and for that goal, *Artefacts* are aligned by using: TDM, aBBs-based ABB concept, UML, or ArchiMate languages. The aBB concept is based on TOGAF's ABB and SBBs. There isn't a definition of services architecture, but there are common characteristics around the ABBs, SBBs, and EADP, which depend on the EHRP, [32]. The EHRP delivers the pool of aBBs, where an ABB is a set of aBBs. aBBs can be interfaced by using the API approach that is based on, [33]: 1) Modelling APIs with UML/Archimate; 2) A schema model is a contract between the EADPbETP and an actor; 3) A schema model is a contract describing what the API is and how it works; and 4) Uses an agile strategy. ABBs and SBBs are assembled to offer EA deliverables. The dimensions of EA and EADPbETP are scoped to ETP's boundaries, which integrate heterogeneous types ICS components, [7], [15]; and are supported by: 1) Applying an EA strategy; 2) Defining EADPbETP's and EA's interactions; 3) Applying *Artefacts* integration; and 5) Defining deliverables in the form of EA Patterns (EAP).

5.5 EAP's Usage and Integration

The usage of *Artefacts*, ABBs/SBBs, EAPs, and EADP can be confusing because an EAP can contain ABBs/SBBs and *Artefacts*. At the same time, an ABB can contain EAPs, and that depends on ETP's strategy that is proper to the *Entity* and that is why there is a need for an MDTCAS. The EAP enriches the MDTCAS Common Denominator Patterns (CDP), and it manages: 1) ERDB's concurrent access; 2) the Applications' user interface; and 3) EHRP activities. EAPs include the following patterns: 1) Domain Logic; 2) Data Source Architectural; 3) Object Relational Behavioural; 4) Object-Relational Structural; 5) Object-Relational Metadata Mapping; 6) Web Presentation; 7) Distribution; 8) Offline Concurrency; 9) Session State; and 10) Base Patterns. EAPs and CDPs are parts of the MDTCAS, which use OO relationship types. Relationships interconnect EAPs that use AMEFF or IHI Interchange Formats (IF), [34], [35].

5.6 IF-based EADP

The MDTCAS contains an IHI IF that maps to the common elements of the following methodologies and standards: OOM, UML, Archimate, and ERD/ERM. This implies that EA tools outputs are transformed to MDTCAS' IF and these files are persisted in EADP's repository, which supports Gap's Analysis and Evaluations (GAE).

5.7 GAE's Implications

The EADP enables GAEs on various ETP levels and various EA and ICS components. A GAE shows in each TDM's phase, whether an ETP made improvements or regressions. Concretely GAE's implication is: 1) In the case of EHRP, it shows how much *Artefacts* were generated and if the ETP has a sufficient level of integrity; by simply using ERDB technics and tables' differential technics; 2) Did the ERBD improve ETP's cohesion; and 3) Did the EADP improve EA's and DevOps integrations.

5.8 The ERDB's Specific Solutions Factors' Evaluations

Based on the AHMM4ADP, LRP4ADP, and DMS, for this CSA's CSFs/KPI were weighted and the results are shown in Table IV. This CSA's result of 8.7, is a sufficient result, and that is due to EADP's complexities to adapt to various solutions and this is a delicate phase. And the next step is to execute the PoC.

Table 3. This CSA's average is 8.70

Critical Success Factors	HMM enhances: KPIs	Weightings
CSF_EADP_Specific_Solution_Factors	Proven	From 1 to 10. 10 Selected
CSF_EADP_Specific_Solution_LS	Possible	From 1 to 10. 09 Selected
CSF_EADP_Specific_Solution_aBB	Complex	From 1 to 10. 08 Selected
CSF_EADP_Specific_Solution_ABB_SBB	Complex	From 1 to 10. 08 Selected
CSF_EADP_Specific_Solution_EAP	Complex	From 1 to 10. 08 Selected
CSF_EADP_Specific_Solution_IP	Possible	From 1 to 10. 09 Selected
CSF_EADP_Specific_Solution_GAE	Possible	From 1 to 10. 09 Selected

valuation

6 The PoC

6.1 Basic Preparations

As shown in Fig. 11, the first step is to prepare the PoC's environment by setting up the ADP's vision, MDTCAS/TDM and extracted *Artefacts* generated by the EHRPs. And afterward, start the phases of ERDB-based EADPbETP's integration. This PoC uses mainly Archimate environment (Archi) and its given project Archisurance which was modified as this PoC's Applied Case Study for TRADf (ACS4TRADf), [36].

Fig. 11. The PoC's basic preparation steps

Many of PoC's modules were already used in previous *TRADf*-related RDPs and PoCs, [9]. The EHRPs and ERDBs enable ADPbETP's integrity and feasibility checks, [1].

6.2 ERDB-based Integrity and Feasibility Check

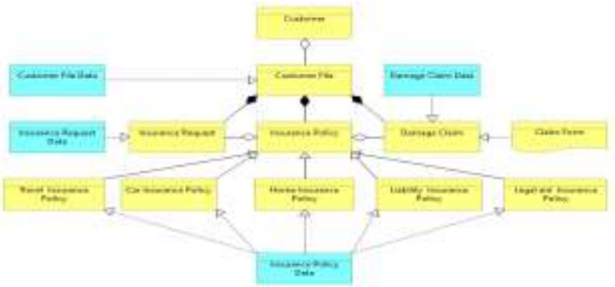


Fig. 12. PoC's Artefacts' based cATR ClsD, [36]

This PoC uses *TRADf's* mature modules (mainly the author's previous work that is related to the UP, which presents the extraction of *Artefacts*) and verified external solutions. *Artefacts* are assembled to build ABBs, OBBs, and complex Transactions (cATR) shown in Fig. 12. The cATR Class Diagram (CLsD) is presented in Fig. 12. The CLsD optimally maps to an ERD.

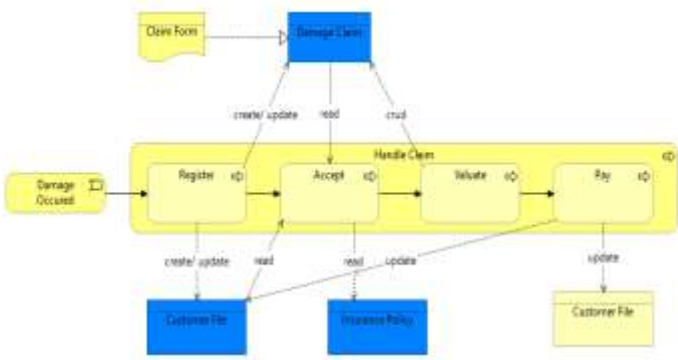


Fig. 13. cATR's activity diagram that respects the CLsD and ERD, [36].

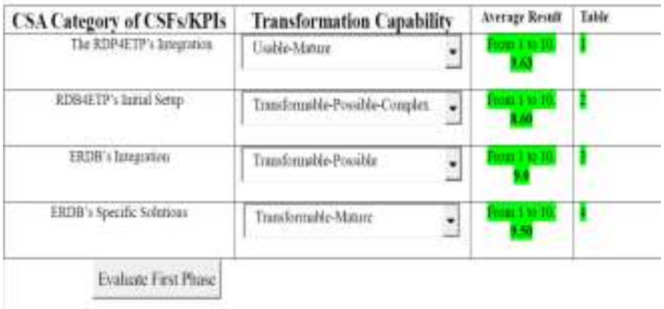
The *Artefacts*-based cATR is designed using a UML activity diagram (translated into a BP), which optimally matches the CLsD and ERD; that is a main MDTCAS constraint and it also proved that the granularity level can be used to refine the "1:1" mapping, [30]. A logical view of the cATR is presented in Fig. 13, and its consumption of *Artefacts*, in which all events are exchanged between various nodes, requires encryption which is defined in the TDM. The cATR uses a set of *Artefacts* which are assembled in an MDTCAS (that maps to UML and ArchiMate's elements). The ADM-based TDM's phases B and D are used to implement the needed MDTCAS-based cATRs.

6.3 ADPbETP' Design and Implementation

An essential constraint for the PoC is to use of existing standards in a reduced form, which corresponds to the MDTCAS and simplifies EADPbETP's integration. In this case, MDTCAS transcendent *Artefacts* and diagrams are used. The IHI MDTCAS includes *Artefacts*, ERD/RDB, *and resources* to be used to integrate basic architecture and design models. To identify the initial sets of Factors that are related to the RQ and ADPbETP's integration. The PoC uses HDT-based mixed qualitative and quantitative methods. The CSF's analytical process. The PoC in the beginning uses Phase 1 which is mainly based on the HDT decision tables, which use *TRADf's* weighting concept. Phase 1 is used to weigh the relative importance of CSAs and CSFs for the usage of MDTCAS, ERDB, and EADPbETP and that is done using a decision-tables, [22], [31].

6.4 POC's Phase 1

Table 4. This CSA's average is (rounded) 8.73.



CSA Category of CSFs/KPIs	Transformation Capability	Average Result	Table
The RDP4ETP's Integration	Usable-Mature	From 1 to 10. 8.63	1
RDB4ETP's Initial Setup	Transformable-Possible-Complex	From 1 to 10. 8.60	2
ERDB's Integration	Transformable-Possible	From 1 to 10. 9.8	3
ERDB's Specific Solutions	Transformable-Mature	From 1 to 10. 8.50	4

Evaluate First Phase

LRP4ADP's outcome proves the existence of a major ADPbETP knowledge gap and its (or Phase 1's) outcome supports RQ's credibility. The use of the LRP4ADP and *TRADf'*s archive or knowledge-base, of an important set of references, previous author's works, documents, and links.

6.5 Selecting ADPbETP's Node

Factors (CSA/CSFs) are linked to various HDT-based DMS scenarios. The PoC is based on the Factors (CSFs') binding to specific RDP4ETP resources, where the ADPbETP was prototyped using *TRADf's*. The HDT represents the relationships between this RDP4ETP's RQ/requirements, *Artefacts*, and selected Factors (CSAs/CSFs). PoC's interfaces were achieved using the Archi tool, Microsoft Visual Studio .NET environment, and *TRADf's*. The EADPbETP uses calls to resulting *Artefacts*, MDCATS, to execute HDT actions related to EHRP, EADPbETP, and ERDB requests. CSFs were selected and evaluated (using Weightings, HDT, and DMS) and the results are illustrated in Table V, which shows that the DMS is feasible mainly because of EADP maturity. In fact, it is essential for the DMS' risk concept. HDT's main constraint is that CSAs having an average result below 7.5, will be ignored. This fact leaves the EADPbETP's CSAs (marked in green) effective for RDP4ETP's conclusion(s); and drops the CSAs marked in red. Phase 1, shows that the DMS will probably succeed and that the ERDB-based EADPbETP can be implemented, but is complex. The PoC can proceed to Phase 2.

6.6 PoC's Phase 2

This phase includes the following steps:

- MDTCAS/TDM's Setup and CSFs' Selection: The Phase 2 setup includes: 1) Sub-phase A or the Architecture Vision phase's goals, establishes the EADP approach and goals; 2) Sub-phase B, or the Business Architecture phase establishes DMS' target TDM and related EHRP activities; 3) Sub-phase C shows and uses the Application Communication Diagram to describe EHRP and EADP activities; 4) Sub-phase D or the Target Technology Architecture shows the needed DMS' optimal ICS landscape; and 5) Sub-phases E and F, or the Implementation and Migration Planning, presents the transition EADP based architecture, which proposes intermediate situation(s) and evaluates DMS' statuses. The HDT-based DMS has mappings to *Entity's* resources and defines relationships between *Artefacts,* MDTCAS (models and elements), and requirements/problems.
- Problems Processing in a Concrete HDT Node: The DMS solves ADPbETP's problems, where CSFs link to specific ERDB or ADPbETP problem types and has a set of actions that are processed in a concrete HDT node. For this goal, the action *CSF_ADPbETP_Extraction_Procedure* was called and delivered Solution(s). Solving Problems involves the selection of actions and possible Solutions for multiple *Project* activities. The HDT is mixed quantitative/qualitative and has a dual-objective that uses the following steps: 1) In Phase 1, *TRADf's* interface implements HDT scripts to process the selected CSAs. And then relates PoC's resources to *CSF_ADPbETP_Extraction_Procedure*; 2) The DMS is configured to weigh and tuned to support the HDT; 3) Link the selected node to HDT to deliver the root node; and 4) The HDT starts with the *CSF_ADPbETP_Extraction_Procedure* and proposes Solution(s) in the form of EADP actions/improvements.
- Solution Nodes: HDT scripts support an AHMM4ADP instance that is processed in *TRADf's* background to deliver ADPbETP risks' mitigation value(s). The AHMM4ADP-based DMS uses *Artefacts* and the ERDB to deliver concrete actions.

7 Conclusion

Legacy ICS' UPs are very complex and can cause ETP failures and success rates. ETP's success rates can be improved by using *Artefacts*-based MDTCAS and ERDB. ADPbETP uses a just-enough approach and the PoC proved its application's complexities. The ADPbETP support

Units based *Entities* and the proposed ERDB is an optimal approach for unifying implementation, integrity checking, and feasibility activities. The ADPbETP supports transformation activities; and the LRP4ADP presented a knowledge gap, that is mainly due to the fact that are no similar research approaches and that there is a lack of a Polymathic-holistic approach. The RDP4ETP is part of a series of publications on ETPs, EHRP-based UP, ADM-based TDM, Polymathic models… The ADPbETP uses the HDT and CSFs/CSAs to support ERDB activities. PoC's Table V result of (rounded) 9.40 that used CSFs' binding to RDP4ETP resources, ERDB categories, RQ, and MDTCAS, shows that the ADPbETP is feasible due RDBs' maturity but the EHRP-based UP is risky. The set of ADPbETP's architecture, technical and managerial recommendations:

- EADPbETPs are important for ensuring long-term sustainability and operational excellence.
- This article presents the possibility to implement an IHI EHRP, ADPbETP, and MDTCAS which avoids the financial-only locked-in strategies and ensures ETPs' success.
- The ADPbETP concept adopts a Polymathic-holistic approach, which used iterative change and implementation phases.
- The ADPbETP proposes a realistic solution that is based on ERDB to transform *Entities*.
- The ERDB is used to abstract and interface/map the following ICS categories: ERDB4P, ERDB4A, ERDB4I, ERDB4C, and ERDB4S.
- ERDBs have already various mechanisms for persistence, integrity checks, and relating various ICS modules.
- The ERDB can use various technologies and concepts to unify an ICS-wide RDB concept.
- Each *Entity* constructs its own IHI ADPbETPs.
- The UP unbundles the legacy ICS into *Artefacts* to support the *Unit*'s *UnPs* and the *Entity*.
- *Entity's Artefacts'* stability and coherence are crucial for its evolution.
- EHRPs' bases UPs are ETP's most critical phase.
- *Unit*'s transformation needs an IHI Methodology, framework (like *TRADf*), Domain, and MDTCAS that manages *Artefacts* and *Models*.
- An ETP must implement a TDM and MDTCAS to support ERDB-based EADP's activities.
- The MDTCAS-based EADP fits in the TDM.
- TDM's integration in the ADPbETP enables the automation of all ETP's activities.
- *Artefacts* include ABBs and SBBs.
- ADPbETP interface *Entity's* TDM and delivers the pool of *Artefacts* based EADP categories.
- Avoid consulting firms and build internal ERDB-based EADP solutions.
- ADPbETP is *feasible* and will very probably succeed mainly due to ERDBs' and MDTCAS's maturities and *TRADf's* cross-functional capabilities.
- Viewpoints "M", "O", "S", "I", and "A" present a structured evolution roadmap, as shown in Fig. 14. And in this article the focus is primarily on Viewpoint "A". And on Viewpoint "I" is a second priority.
- APDs high demand for ETPs' and the hyper-evolution of ICS-related technologies, create serious problems because of the differences in their evolution rate. And MDTCAS can avoid such cases of desynchronization.
- All author's works are based on *TRADf*, AHMM, EHRP-based UPs, ADM-based TDM, and RDP; which are today mature and can be applied in various APDs.

CSAs evaluation results are very high, and that is due to the fact that the ERDB-based EADPbETP simplifies RDP4ETP and it is possible to be implemented.

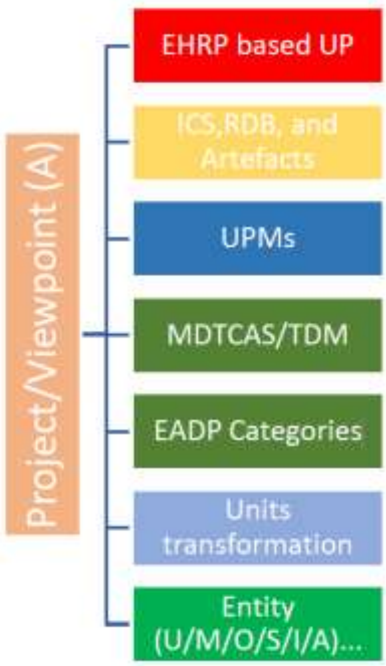


Fig. 14. RDP4ETP's similar Factors' flow, [31].

Acknowledgement:
This research article findings presented above were supported by *TRADf*.

References:

[1] A. Trad. A Relational DataBase based Enterprise Transformation Projects (RDBbETP). INASE Conference. Venise, Italy.
[2] A. Trad. Applied Holistic Mathematical Models for Dynamic Systems (AHMM4DS). *Journal: International Journal of Cyber-Physical Systems (IJCPS)*. Volume 3, Issue 1, Pages 1-24. IGI Global. USA. 2021.
[3] G. Stitt, F. Stitt, W. Vahid, and W. Najjar. A Code Refinement Methodology for Performance-Improved Synthesis from C. International Conference on Computer-Aided Design (ICCAD'06), November 5-9, 2006, San Jose, CA, USA. 2006.
[4] Capgemini. Business transformation: From crisis response to radical changes that will create tomorrow's business. A Capgemini Consulting survey. Capgemini. France. 2011.
[5] S. Easterbrook, J. Singer, M. Storey, and D. Damian. Guide to Advanced Empirical Software Engineering-Selecting Empirical Methods for Software Engineering Research. F. Shull et al. (eds.). Springer. 2008.
[6] C. Quinlan. Business Research Methods. Dublin City University. Cengage Learning. Ireland. 2015.
[7] The Open Group. Introduction to the Architecture Development Method (ADM). The Open Group. USA. 2011.
[8] A. Trad. Organizational and Digital Transformation Projects-A Mathematical Model for Building Blocks based Organizational Unbundling Process. IGI. USA. 2023.
[9] Bizzdesign. Digital Transformation. Bizzdesign. 2022. https://bizzdesign.com/blog-category/digital-transformation/
[10] Chaione. Digital Transformation-The 4 Types of Digital Transformation. 2022. Chaione.com.
[11] Codd E.F., Codd S.B., and Salley C.T. "Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate". Codd & Date, Inc 1993. http://www.fpm.com/refer/codd.html.
[12] IBM. An overview of cloud security. IBM. 2022. https://www.ibm.com/topics/cloud-security
[13] IBM Cloud. 2021. *What is ETL?* IBM. Avaible at: https://www.ibm.com/topics/etl#:~:text=the%20next%20step-,What%20is%20ETL%3F,warehouse%20or%20other%20target%20system.
[14] Ph. Beauvoir. Multi-User Repository Development and Archi Roadmap Updated. ARCHI. 2017. https://www.archimatetool.com/blog/2017/06/09/multi-user-repository-development-and-archi-roadmap-updated/
[15] The Open Group. The Open Group ArchiMate Model Exchange File Format. The Open Group. 2023. https://www.opengroup.org/open-group-archimate-model-exchange-file-format
[16] B. Schwab, L. Wasson, J. Sholberg, and S. Kwong. (1990). Data Base Management for an Integrated Network Management System: Requirements/Test/Evaluation. In: Kershenbaum, A., Malek, M., Wall, M. (eds) Network Management and Control. Springer, Boston, MA. https://doi.org/10.1007/978-1-4613-1471-4
[17] A. Cruz. Development of a Network Management-System for the EU DICONET Project. ETSETB. 2010. https://upcommons.upc.edu/bitstream/handle/2099.1/9449/developmentofanetworkmanagementsystemfortheeudiconetproject.pdf?sequence=1
[18] E. Hosiaisluoma. ArchiMate Cookbook-Patterns & Examples. 2022.
[19] J. Kasarkod. Integration of SABSA Security Architecture Approaches with TOGAF ADM. InfoQ. 2011. https://www.infoq.com/news/2011/11/togaf-sabsa-integration/
[20] Integrify. BPM Platform: Manage Business Processes with Automation-BPM Platforms can speed the development of process-based applications. Integrify. 2023. https://www.integrify.com/bpm-platform/#:~:text=A%20BPM%20platform%20provides%20a,tasks%20that%20constitute%20a%20process
[21] A. Sabau, S. Hacks, and A. Steffens. Implementation of a continuous delivery pipeline for enterprise architecturemodel evolution. Software and Systems Modeling.

SPECIAL SECTION PAPER. 2020. https://doi.org/10.1007/s10270-020-00828-z

[22] Visual Paradigm. Project Viewpoint. Visual Paradigm. 2023. https://www.visual-paradigm.com/support/documents/vpuserguide/4455/4456/86497_projectviewp.html

[23] CHARITO. Designing API products. eaminds. 2018. https://eaminds.com/2018/03/09/designing-api-products/

[24] IBM. API management platform. IDM. 2022. https://www.ibm.com/topics/api-management#:~:text=An%20API%20management%20platform%20is,high%20performance%20and%20security%20standards

[25] S. Peterson. Why it Worked: Critical Success Factors of a Financial Reform Project in Africa. Faculty Research Working Paper Series. Harvard Kennedy School. 2011.

[26] Wikipedia. XML log. Wikipedia, the free encyclopedia. 2023. https://en.wikipedia.org/wiki/XML_log

[27] JT. Logback Introduction: An Enterprise Logging Framework. 2016. Springframework. https://springframework.guru/logback-enterprise-logging-framework-2/

[28] A. Trad. A Transformation Framework Proposal for Managers in Business Innovation and Business Transformation Projects-Intelligent atomic building block architecture. Journal: Procedia Computer Science. Volume 64; Pages: 214-223. Elsevier. 2015. Sciencedirect.com

[29] A Transformation Framework Proposal for Managers in Business Innovation and Business Transformation Projects-An Information System's Atomic Architecture Vision. Journal: Procedia Computer Science. Volume: 64, Pages: 204-213. Elsevier. 2015. Sciencedirect.com

[30] S. Yalezo, and M. Thinyane. Architecting and Constructing an Service Oriented Architecture Bridge for an Model View Control Platform. IEEE Computer Society Washington, DC, USA. 2013.

[31] T. Quang Phu, and H. Thi Yen Thao, H. Enterprise Risk Management Implementation: The Critical Success Factors For Vietnamese Construction Companies. Journal of Multidisciplinary Engineering Science Studies (JMESS). ISSN: 2458-925X. Vol. 3 Issue 2, February - 2017. http://www.jmess.org/wp-content/uploads/2017/02/JMESSP13420283.pdf

[32] B. Bebensee, and S. Hacks. Applying Dynamic Bayesian Networks for Automated Modeling in ArchiMate-A Realization Study. 2019 IEEE 23rd International Enterprise Distributed Object Computing Workshop (EDOCW). IEEE. 2019. DOI 10.1109/EDOCW.2019.0001.

[33] S. Patni. Pro RESTful APIs Design, Build and Integrate with REST, JSON, XML and JAX-RS. Apress. Santa Clara, California. USA. 2017.

[34] Business Transformation Projects Based on a Holistic Enterprise Architecture Pattern (HEAP): The Basic Construction. Book: Handbook of Research on Digitalization Solutions for Social and Economic Needs. Pages: 45-82. IGI Global. USA. 2023.

[35] A. Crosswell. Bricks and the TOGAF TRM. National Institute of Health. 2014.

[36] Archi. Archi The Open Source Modelling Toolkit. ARCHI. https://www.archimatetool.com/

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The author contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflict of interest to declare.