Data Warehouse-Based Analytical System in Private Higher Education Institution

Mario Fabijanić*, Domagoj Ružak*, Ana Novosel*,** and Tomislav Hlupić*,**
* Algebra University College, Zagreb, Croatia
** Poslovna Inteligencija, Zagreb, Croatia
tomislav.hlupic@racunarstvo.hr

***Abstract* – Data warehouses have been present in the business environment for several decades, providing businesses with deeper insight into their business data through which they gain competitive advantage and decision-making support. However, the integrated characteristics of data warehouses can be a key factor in improving process automatization through the integration of various heterogeneous sources, speeding up certain processes, and providing process integration in the organization without impacting the existing infrastructure. In this paper, an analytical system based on a data warehouse is presented which integrates data sources in a high education institution, such as internal operational applications and structured and semi-structured files, and incorporates new data sources such as websites and their interfaces. By introducing the integration of the data sources, processes that share the same data scattered across multiple files or need multiple manual entries can use a single source of truth, while also having the possibility of automatic integration of open-access data. In the paper, the necessary processes are identified, and their respective data marts are proposed, while also providing proof-of-concept for several chosen processes.**

Keywords – Data warehouse; data mart; education institution; teaching staff; evaluation system; evaluation

I. INTRODUCTION

Taking care of the quality of the teaching staff and the consequent quality of the education they provide is one of the most important tasks of every higher education institution. This is especially pronounced in private institutions, which must continuously confirm and improve their representation in the market and society. To make this possible, higher education institutions collect relevant data, process them, and categorize teaching staff, on the one hand, to reward the most successful, on the other hand, to empower the less successful.

Data warehouses are a well-known part of higher education information systems. They are introduced at some point during the development of those information systems to provide an efficient analysis of the provided data through an intuitive service [1]. Throughout time, data warehouses grow through time, face typical data warehousing issues (like data quality), and require adjusting the ETL loads based on the analysis of usage statistics [2] [3]. Also, similar systems were introduced on other universities [4], where some have been focused on calculating performance through multiple academic years [5].

At Algebra University College, an evaluation system has been established. It implements the evaluation criteria for the lecturers and their assistants, which are transparently scored and displayed in real-time, at each semester and academic year, for each member of the teaching staff. The evaluation system generates four ranking lists of courses: a list of lecturers, a list of assistants in undergraduate study courses, a list of lecturers, and a list of assistants in graduate study courses. The number of points for the lecturer on a certain course and the number of points for the assistant on a certain course will be within each list. For example, a lecturer who teaches three undergraduate courses will be shown on three lists and will potentially have a different number of points earned for each course.

All lists are divided into five categories. The number of points is monitored continuously, and the final calculation will be made at the end of the academic year for each course since certain elements of scoring depend on the annual work, and some of them depend on the work on the course itself within a particular semester.

The evaluation criteria are divided into categories:

- The result of the first survey in the semester
- The result of the survey at the end of the semester
- Exam quality
- Innovative approaches and good practices in teaching
- Absence of lateness to classes
- Absence of delay in entering grades
- Absence of points input delay
- Absence of delay in setting up courses at the beginning of the semester
- Participation in the preparation of students for competitions / extracurricular activities, or mentoring of final and graduate theses and practice
- Attending training for lecturers and their assistants (participation + results)
- Participation in international exchange, teaching in English, or professional/scientific production in English, related to our institution
- Publication of papers and their registration in CROSBI (*Croatian Scientific Bibliography* [7]).

One aspect of the above categories is to cover the lecturer and lecturer assistant's everyday work quality

during the semester. At Algebra University College, an internal operating system is used to handle the whole teaching process, e.g., the timetable for teaching, course materials, student, and lecturer's (assistant's) attendance lists, exam terms, exam applications, and student survey results. Besides the information needed to support those activities, there are also time stamps for each activity which help to evaluate some of the mentioned criteria.

Another aspect is related to the thrive to strengthen students and teaching staff, e.g., helping students to participate in extra curriculum activities and competitions, or lecturers attending additional training targeted to their needs. The publication of scientific papers also belongs to this aspect, and all those activities must participate in the evaluation process.

The exact formulations for each of the above evaluation criteria nor their weight coefficients are not the subject of this paper, and therefore they will not be explained further. Rather, this paper is dedicated to the warehouse architecture system which may support the whole process.

From the above, the teaching staff evaluation system is complex to implement because the data required for the evaluation itself are found in different IT systems, databases, websites, and individual documents. Manual data collection is not time-optimal, does not always provide information, and is subject to human error. Such mistakes may significantly reduce confidence in the entire evaluation system and can increase dissatisfaction among all stakeholders.

This paper proposes a solution to the described problem in the form of a data warehouse that collects data from different sources, continuously, and transparently, to make the whole process consistent and fair to evaluated lecturers and their assistants. The proposed solution consists of an architecture that is not only limited to the currently valid evaluation criteria but can subsequently be supplemented with new data marts to support some new criteria that may be added in the future. As an example of the usefulness of the proposed system, this paper presents the data mart developed to display data about the scientific production of an individual member of the teaching staff. In addition to the evaluation process, the proposed data mart is an example of usefulness in other needs also, such as discovering the same or similar scientists' research areas, who may not be aware of it.

II. METHODOLOGY

The proposed data warehouse serves as a central point for gathering various data from multiple sources. Examples of these sources are internal operational software with transactional databases, structured and semi-structured data scattered in multiple files that typically have the same data stored for different purposes, and outside sources such as websites with their APIs. Through introducing a data warehouse, these data are cleansed, integrated, and stored in a way that previously independently stored data can potentially provide more insight into existing data.

A. High-level architecture

In Figure 1, the high-level analytical system architecture is shown which will is described in the next paragraphs. The architecture can be divided into three main layers: the staging area, with a staging database and landing zone, the data warehouse, and the business intelligence layer. Data, arriving from the data sources described later in this chapter, are either loaded directly into the staging database or they are stored in the landing zone from where the integration processes fetch the files and load them to appropriate staging database areas. All entities in the staging database reflect the structure of loaded tables (when the sources are other databases) and files that have fixed, known structures. External semi-structured data sources require either flattening the data if applicable (such as JSON and XML formatted data) or preprocessing the data when the flattening process would be intensive performance-wise and would lead to a lengthening of the whole integration process.

The data warehouse is modeled using Kimball's star-schema, bottom-up approach by modeling one data mart at a time. Through modeling, conformed dimensions are easily identified, although several of them fall into this category, such as teacher dimension, student dimension, course dimension, etc. Each data mart also introduces new attributes for existing dimensions which are determined by their final use by a certain business process. Dimensional tables, when describing hierarchical data, are flattened to the highest granularity level to represent the hierarchy in the dimension, rather than in the fact table. Initially, the dimensions are of SCD1 and SCD2 type (when applicable), while they might later be redesigned to match the hybrid slowly changing dimension structure if the business processes will need such an approach.

The business intelligence layer is intended for adjusting the data mostly as a support for building models in various business intelligence applications but also can provide data for various ad-hoc analyses or even advanced analytics usage, such as data mining and machine learning. This architecture area is easily adjustable, representing the data in a user-friendly and application-ready structure, and should handle all necessary adjustments so the data models in different applications would be aligned.

B. Categorization of data sources and integration specifics

Most of the operational data come from a single source, the internal operational software, holding different data about student performance and assignments, logs containing the data of teachers' and assistants' actions, and many other data that can, for example, affect the final teacher's evaluation score. These data are stored in a relational database that can be accessed through direct queries or views adjusted for specific purposes.

Other operational data are typically stored in Excel or CSV files created from manual entries. Some of these data are a result of entries in the online forms, while the others are wrangled, adjusted data that is exported from the previously mentioned operational software.

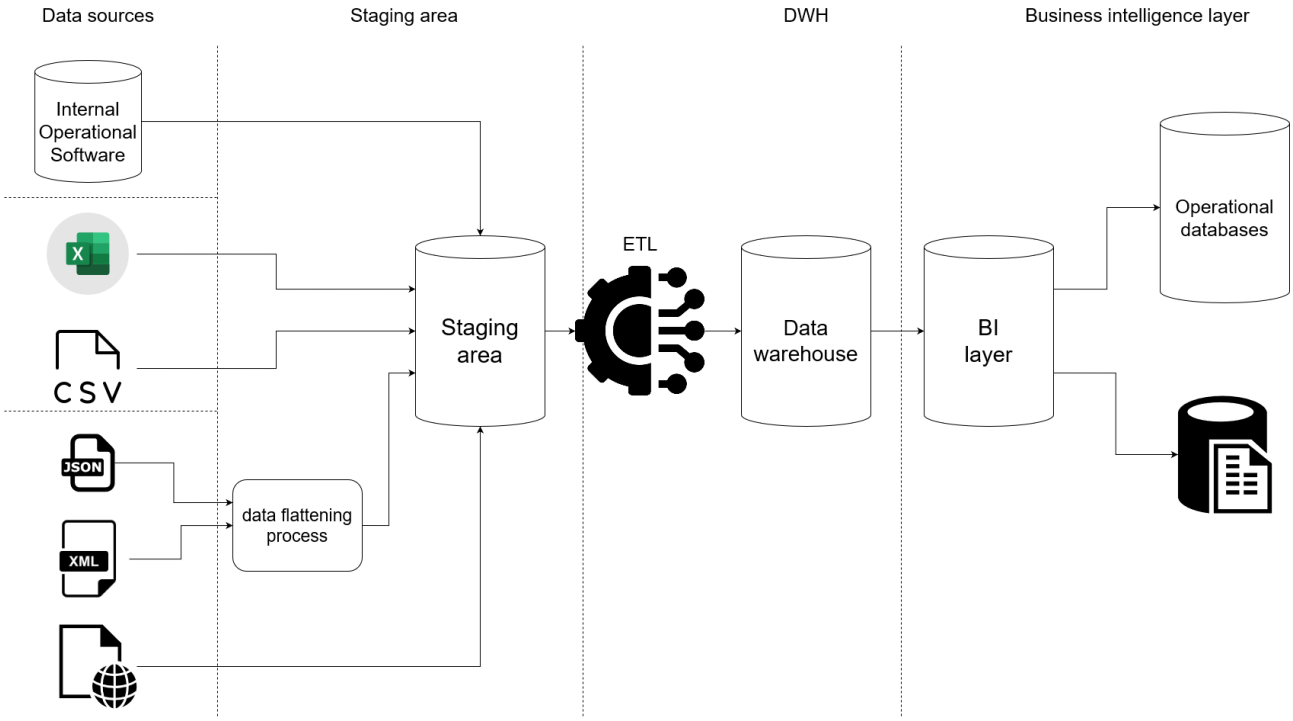

Figure 1. High-level analytical system architecture

Finally, data that is currently largely unused, but can supplement existing data or even become newly used data comes from outside the organization. These data are provided by external websites, each of them being a source for a certain process. An example of the external data from the website and its integration is described in the next chapter. All three data source categories are described in the next paragraphs.

The first data source category includes the data from operational software. These data are integrated on a daily level, which makes the data warehouse a copy of the transactional data, following the definition given by Kimball [6]. That way the basic performance of teachers, such as entering students' scores and grades or course preparations are available for analysis on the next batch load, and the provided data can be shown on dashboards if needed. Moreover, the stored data can be used for further calculations or ad-hoc analysis without impacting the data in the operational software.

The second data source category is various Excel and CSV files whose structure is previously defined. Data stored in the files belonging to this category serve many different purposes. As internal operational software adjustment to new features is lengthy, manual entries serve as data sources for various analyses and reports. Some of these files hold the data originating from the operational software itself but are adjusted or manually wrangled to conform to structures used later for operational and/or analytical purposes. Typically, adjusting the data requires manual labor which generates additional overhead in the processes, often becoming a bottleneck.

Moreover, all the data originating from online forms also fall in this category, as the forms, once created, have a known structure. Even though these data, strictly speaking, are indeed created outside the organization, they belong to the operational data since their structure is defined by the employees inside the organization.

The files containing these data can be integrated both manually and on schedule – both processing approaches include placing files in the landing zone, their ingestion into the staging database, running a defined data quality check, and storing them in the data warehouse.

Fully external data are the third data source category. This category includes all relevant web pages, APIs with semi-structured data, files created outside of the organization (whose structure and content are governed by external users), and generally all files containing data that can augment or supplement existing data from within the organization.

Data sources in this category are usually integrated automatically as the emergence of new external data is unpredictable. The velocity of integration depends on the importance and volume of the data belonging to a certain data source. For example, some data sources create new data daily, but those data are needed periodically (monthly, quarterly…) or only after a certain date, so the integration process schedule is adjusted to the final usage, creating the on-time data in the data warehouse. All automated integrations must also support ad-hoc loads as reloads may be required in certain cases where refreshing the data could be an additional burden.

C. Proposed data mart list

Each data mart in the proposed data warehouse model typically holds the data for a separate business process. Most of the data marts have one or more conformed dimensions, as the dimensionality of this data warehouse

lies in the several inter-joined academic entities such as students, teachers, and courses.

In Table I, a list of proposed data marts with a short description and examples of dimensions are provided.

D. Scientific production data mart

The scientific production data mart is chosen as an example since this data mart serves as a source of data for multiple teaching staff evaluation processes and reports. Currently, three different business processes consume data from this database: the annual scientific publication of an individual lecturer or lecturer assistant, the annual assessment of work, and promotion to a higher teaching or scientific position. In the model, shown in Figure 2, this fact table is named *f_publications*. The main source for the scientific publications data is CROSBI [7], which allows the automatic fetching of all publications belonging to certain authors.

The main prerequisite for the integration is storing the CROSBI identifier in the teacher dimension (marked as *d_teacher* in the data warehouse model), from which a file in BibTeX format with publications can be exported. Other prerequisites include the publication's identifier, depending on its category (DOI for the published papers, ISBN for the published books, etc.) that serves as a unique identifier in the fact table. In this case, a single published publication is seen as a business event with its measures.

The second part is building a bridged table that holds the connection between the publication and its authors. This certain data warehouse design pattern allows a fact table to be related to a dimension in a many-to-many relationship without a time variance since the time variance of the publication is already held in its fact table. Moreover, this supports the occurrences when the author of the existing publication that was published before his tenure at the university entered the dimension – in the next data load, the connection between the publication and the author will be established immediately.

Besides with identifier (DOI/ISBN), each publication is represented by measures, such as the number of authors (calculated during integration), the journal's quartile (derived from the dimension, when applicable), different flags on whether the publication is associated with Algebra University College and whether it was written by an employee or external associate, and dimensional data such as date of publication, the academic year of publication (calculated based on dimensional data), recension type, publication type, etc.

The load of the data in the data mart is done in several steps, as shown in Figure 3. Loading of the data starts after the data fetching from the CROSBI portal when the data about publications from all lecturers and lecturer assistants are gathered using their CROSBI identifiers.

During the preprocessing stage, the DOI and ISBN identifiers are extracted from the data because they can be used to enrich the existing publication description with the data from other sources, such as Scopus journal data, or publication type. Initially, all the data are retrieved, and it is expected that a system that identifies only new entries will be introduced during the next phases of development.

TABLE I. LIST OF PROPOSED DATA MARTS

Data mart	Description	Dimensions
Operational finances data mart	Holds the data for contract payments based on the delivered workload. Data are derived from other data marts	Teacher (d_teacher)
Teacher workload data mart	Holds the data on the delivered workload (lectures, labs, thesis committee…)	Teacher (d_teacher) Course (d_course)
Exams data mart	Holds the data on conducted exams	Teacher (d_teacher) Student (d_student) Course (d_course)
Exam quality data mart	Holds the data on the quality of conducted exam	Teacher (d_teacher) Course (d_course)
Course operational data mart	Holds the course data operational metrics (such as timestamps of the course preparations)	Teacher (d_teacher) Course (d_course)
Thesis data mart	Holds the data of thesis proposal, defense, and committee workload	Teacher (d_teacher) Student (d_student)
Survey data mart	Holds the data of surveys conducted for each course	Teacher (d_teacher) Course (d_course)
Evaluation system reporting data mart	Holds the calculated measures based on the data from other data marts	Teacher (d_teacher) Course (d_course)

The first step in loading the fact table includes fetching the basic publication data from the staging database. The basic data are retrieved from CROSBI data staged in the database, in which BibTeX entries are modified and stored in a single database record.

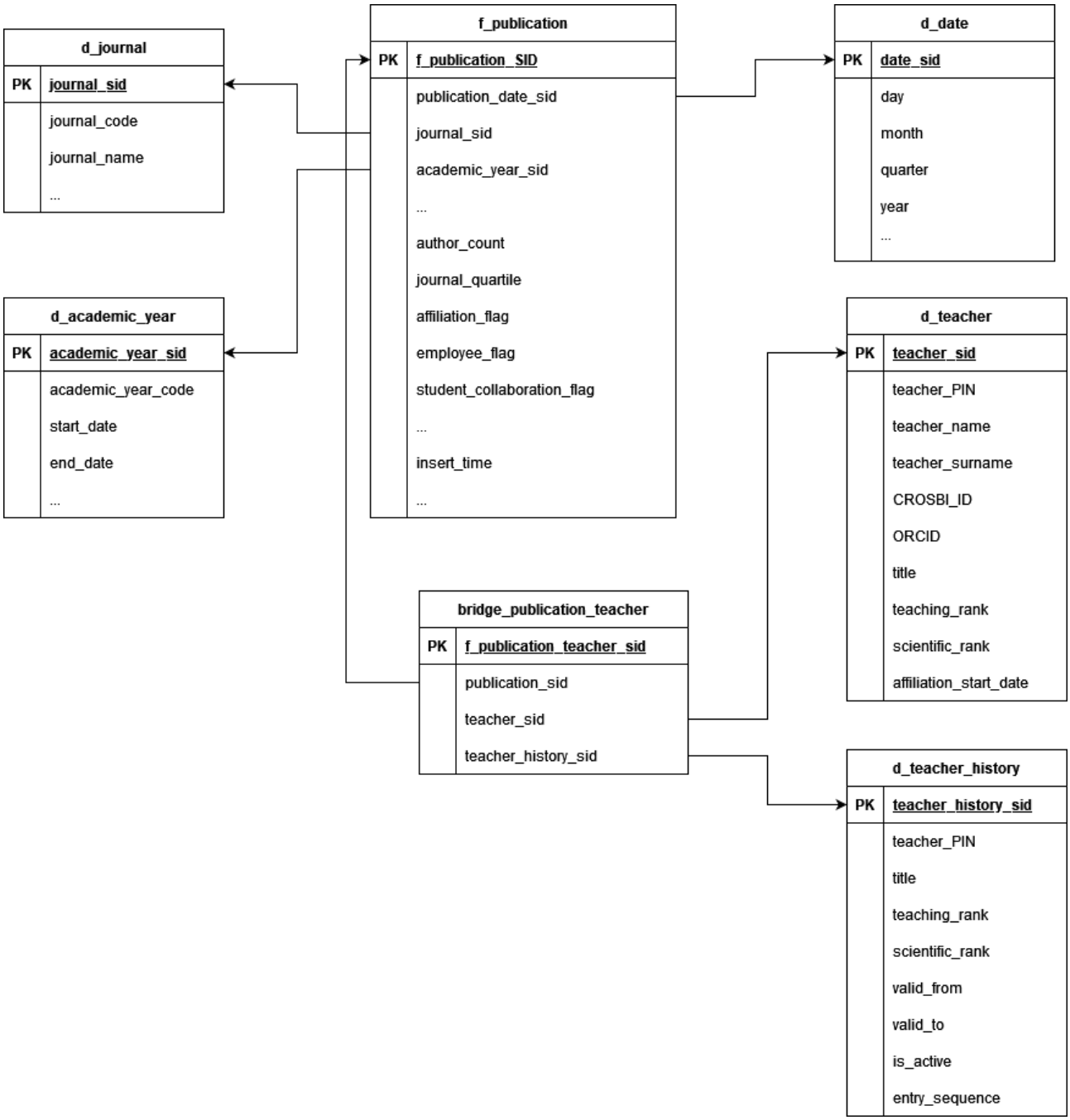

Figure 2. Star schema of the proposed data mart

Further on, those data are enriched with additional data from other data sources, to have the most possible detailed description of a certain entry. Initially, only several data sources are planned besides CROSBI [8] [9], although the extension is easily possible by adding new attributes and updating the existing entries by using their identifiers. After creating the full scope of data for a certain entry, the next step is to integrate them and adjust them to the fact table model through integration packages and lookups to dimensional tables inside them. It should be emphasized that the time-variance of the entry depends on the available information and the publication date, if not available through the data, should be derived based on the set of rules (e.g., if the day of the publication is not available, but month and year are, the date is set to the last day of the given month). Some of the mappings between the fact table and dimension tables are deliberately redundant to enable easier ad-hoc analysis and report creation. The last step before loading the data into the fact table is creating derived measures and flags. These measures and attributes are created based on the set of previously defined rules, such as calculating the number of authors, marking the affiliation to the parent institution, defining whether the publication is a collaboration between a teacher and a student, etc. Those measures are later used in various reports, impacting the performance of the reports (since the reporting model is simpler and the calculations do not have to be done during the report creation), and can be used in multiple different reports. Finally, after the fact table is loaded, the bridge table that connects publications with authors is filled.

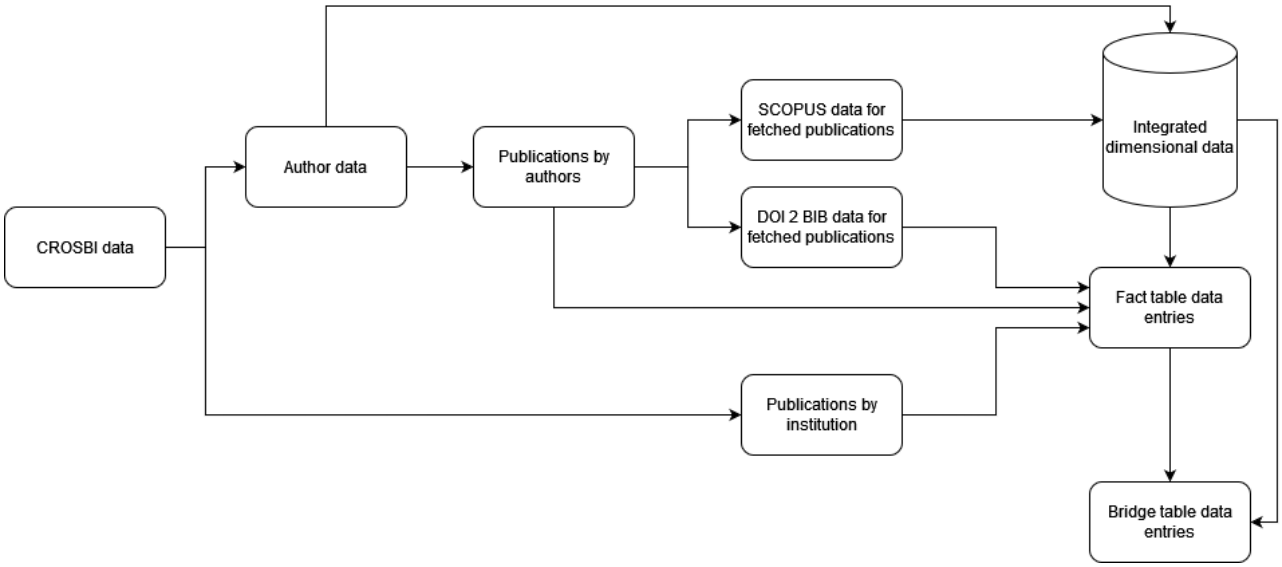

Figure 3. Publication data mart load data flow

The load of the bridge table comes in two phases: during the first phase, the data about the publications currently located in the staging database are loaded, while the second phase identifies the new teachers loaded in the teacher's dimension and loads the history of their publications. The identification of new entries is done through a timestamp stored as a technical attribute in the dimension. As the publications belonging to a certain person are retrieved during the staging phase, this allows adding entries for both new and existing publications.

III. CONCLUSION

In this paper, an analytical system built on a data warehouse is introduced, to implement an architecture to support teaching staff evaluation within private higher education institution. It merges existing data sources, like internal operational applications, structured and semi-structured files, and websites and their interfaces, as well as adds new sources of data. By providing the integration of the data sources, operations that require several human entries or share the same data dispersed across multiple files may now use a single source of truth and automatically integrate open-access data. The paper identifies the required processes and proposes the appropriate data mart as the proof-of-concept for a few selected operations. Future work will cover the implementation of new data marts needed to make the proposed analytical system more comprehensive, covering the majority or even all the needs that higher education institutions have regarding their teaching staff evaluation.

REFERENCES

[1] Baranović, M., Madunić, M., & Mekterović, I. (2003). Data warehouse as a part of the higher education information system in Croatia. Proceedings of the International Conference on Information Technology Interfaces, ITI, 121–126.

[2] Mekterovic, I., Brkic, L., & Baranovic, M. (2009). Improving the ETL process of higher education information system data warehouse. Proceedings of the 9th WSEAS, 8(10), 265–270.

[3] Mekterović, I., Brkić, L., & Baranović, M. (2009). Improving the ETL process and maintenance of higher education information system data warehouse. WSEAS Transactions on Computers, 8(10), 1681–1690.

[4] Dell'Aquila, C., & Tria, F. Di. (2007). An academic data warehouse. AIC'07: Proceedings of the 7th Conference on 7th WSEAS International Conference on Applied Informatics and Communications, 229–235.

[5] Mihai, P., Marian, L. M., & Alexandru, L. M. (2010). Measuring The Performance Of Educational Entities With A Data Warehouse. Annales Universitatis Apulensis Series Oeconomica, 1(12), 176–184.

[6] Kimball, R., & Ross, M. (2002). The data warehouse toolkit: the complete guide to dimensional modelling. In Nachdr.]. New York [ua]: Wiley.

[7] CROSBI - Croatian Science Bibliography. https://www.bib.irb.hr/ (Date of access 6. 2. 2023).

[8] Rose, M. E., & Kitchin, J. R. (2019). pybliometrics: Scriptable bibliometrics using a Python interface to Scopus. SoftwareX, 10, 100263.

[9] DOI2BIB. (n.d.). https://www.doi2bib.org/bib/ (Date of access 6. 2. 2023.)