

Aggregate Design of a Conference Management System in Object-Oriented Database

Emecheta N. C.

Dept. of Computer Science, University of Port Harcourt
Choba, Port Harcourt, Nigeria

Publication Date: 2025/02/13

Abstract: This paper presents a design strategy that may be used to aggregate a set of things with complex interactions into a unified construct that is simple to use and understand. The architectural design of object-oriented databases in data storage systems uses this architecture. This research was driven by the need to enhance the object data and object process communication in a data storage system. The object-oriented system analysis and design methodology (OOADM) is the setting in which it is utilised. Conference Management System (CMS) was chosen as the study's use case to demonstrate how the object-database aggregate architecture is used to address the complexity of information representation utilising object-oriented database systems. The conditions that apply in a CMS system's operational environment served as the basis for the system development parameters. The system's outcome demonstrates how an intricate system may be created by employing aggregate design to streamline the system modelling and facilitate effective communication between object-oriented processes and object-oriented data.

Keywords: *Aggregate, Object-Oriented, System Analysis and Design Methodology, Conference Management System.*

How to Cite: Emecheta N. C. (2025). Aggregate Design of a Conference Management System in Object-Oriented Database. *International Journal of Innovative Science and Research Technology*, 10(1), 2347-2357. <https://doi.org/10.5281/zenodo.14863099>.

I. INTRODUCTION

Since complicated data is not mapped to relational rows and columns before being stored, object databases are well suited for applications that deal with extremely complex data. Large volumes of complicated data are needed by many contemporary business applications. Therefore, it is necessary to store a lot of complicated data and combine it with simple data in a single application. In order for data to be inherited, extended, polymorphed, and hidden (protected) from external programmes, it is also necessary to make databases simple to integrate into the way contemporary programmes are produced. This suggests novel methods for simultaneously storing, processing, and retrieving simple and complicated data, as well as leveraging aggregate design to make the design and integration of such systems simpler.

A. Aggregate Design

The majority of business areas have Entities that are extremely intertwined, sometimes to the point where there is always a path connecting any two Entities. We can attempt to reduce the amount of associations in our design, which is a good idea, but it may result in a loss of translation between business and software. It is challenging to distinguish between the limits of a change in a standard object model. In systems with concurrent access, like Web

applications, this is especially acute. In a model with ambiguous object borders and complicated associations, it is challenging to ensure the consistency of changes made to the objects. While treating each object as though it were independent of the others is not a sound strategy, refreshing each object after a change is also impractical. Finding a balanced answer is necessary.

We may use the Aggregate pattern to assist us with this tricky issue. An aggregate is a lifecycle pattern originally described by Eric Evans (Eric, 2003). It is a group of related items that are taken into account as a single entity for the purposes of data updates. A collection of things with complex relationships can be aggregated into a unified construct that is simple to use and make sense of. Each aggregate has a boundary and a root that defines what belongs inside of it. Any number of objects can be included in an aggregate, but one entity is always the root. A particular Entity found within the aggregate is known as the aggregate root. It is the only Aggregate Entity that client objects can hold references to because it is the sole Aggregate Entry-Point. Only the context of the aggregate root provides access to other aggregate objects.

Nothing can hold a reference to anything inside the confines of the aggregate. This indicates that the root serves as the conduit for all of the aggregate's capabilities. Roots

may occasionally return a reference to an item inside the border, but they can only be momentarily used by the client. The roots of other aggregates can be referenced by objects inside the aggregate. All of the aggregates' invariants must be guaranteed by the root object. An aggregate's items are all deleted when the aggregate's root is removed.

A properly designed aggregate is one that can be modified in any way required by the business with its invariants completely consistent within a single transaction. Since aggregate must be designed with a consistency focus, it implies that the user interface should concentrate each request to execute a single command on just one aggregate instance. If user request to accomplish too much, it will force the application to modify multiple instances at once. Therefore aggregate are chiefly about consistency boundaries and not driven by a desire to design object graphs (Yoder and Barcalow, 2019).

B. Rules Applying to Aggregates

- The aggregate root is in charge of examining invariants within the aggregate and has a global identity.
- Non-root entities contained within an aggregate have a local identity that is exclusive to that entity.
- Only references to the aggregate root may be contained in external code. The root can provide references to internal things, but these references are only used momentarily and the root holds onto them. Value Objects can be given without worry because they are unchangeable and unaffected by side effects.
- The persistence can only be directly loaded from aggregate roots. The traversal of associations must be used to locate all other items.
- All aggregate invariants must be satisfied for any change to occur within the aggregate border.

C. Typologies of Aggregate

To create a model for a particular business domain, there are practically unlimited design options. Each of these approaches will have unique qualities in terms of effectiveness, reliability, simplicity, and other factors. To construct a system that will have the appropriate characteristics, good aggregate design is essential.

There must be consistency at the aggregate level. This implies that it is the responsibility of an aggregate root to ensure that business invariants—i.e., business rules that must be true at all times—are satisfied. This implies that a system that is correctly constructed always alters a single instance of Aggregate per transaction. Although it may seem rigid, this should generally be the aim. This is the main justification for using aggregates in modelling.

There must be consistency at the aggregate level. This implies that it is the responsibility of an aggregate root to ensure that business invariants (i.e., business rules that must be true at all times) are satisfied. This implies that a system that is correctly constructed always alters a single instance of Aggregate per transaction. Although it may seem rigid,

this should generally be the aim. This is the main justification for using aggregates in modelling.

A graph of objects that serves as a consistency limit for our domain regulations is what we mean by an aggregate. We can either enforce them (making them invariant) or be compelled to have corrective policies in place, depending on how the aggregate is designed. Since they affect the behaviours that are modelled in our domain, it is crucial to carefully define aggregate boundary limits.

D. A Conference management system is the application area.

The usage of pointers allows for the many-to-many access to objects. A collection of entities and value objects (domain objects) that theoretically belong together is referred to as an aggregation. It is a group of related items that are taken into account as a single entity for the purposes of data updates. It also includes a set of operations that can be performed on those domain objects. As a user case, we'll use a conference management system. Object-Role Modelling (ORM) is a technique for conceptually modelling, querying, and mapping an information system to logical (e.g. relational) levels. The Natural Language Information Analysis Method (NIAM) is one of many flavours of ORM.

II. ANALYSIS

Analysis is the breakdown of the activities needed to construct a system in order to identify the component parts needed for the system's design. This will entail breaking down the individual components required for a database item to communicate with a piece of code that manipulates it.

A. The Universe of Discourse (UoD)

In this study, we emphasise the data perspective and presumptively develop a formal model of the application domain or universe of discourse (UoD) as part of the design process. The UoD must be understood thoroughly in order to be done correctly, and there must be a way to define this understanding in a straightforward manner. By using plain language and intuitive diagrams that can be filled with examples, as well as by analysing the material in terms of basic or simple facts, Object-Role Modelling (ORM) streamlines the design process. It offers a conceptual method of modelling by expressing the model in terms of common notions like objects and roles. The research's chosen application domain or discourse universe is a conference management system. This was chosen because paper reviews in a dispersed context need multitasking.

B. Analysis of UoD -Conference Management System

A conference is a gathering of people where they can present and talk about their work. The topic may be intellectual or relate to other spheres of life. Academic conferences provide as a forum for researchers to discuss their findings. Work is typically presented in the form of a brief, succinct presentation lasting between 10 and 30 minutes, usually with time for discussion. The work may be

collected in writing form as academic papers that are assessed by a group of chosen scholars from a wide geographic area and then published as conference proceedings.

A web-based program called a conference management system aids in the planning of academic conferences. It supports the relevant tasks of the program chair(s), conference organizers, writers, and reviewers. A conference management system can be regarded as a domain-specific content management system. Similar systems are used today by editors of various journals (Barrett, 2000).

In this paper, we will investigate conference activities and use their domain components to create a conference management system that can automate the processes that make up a conference system's domain content.

There may be one or more keynote speakers at conferences, who are typically renowned academics. The conference may include panel discussions, round tables on diverse topics, and workshops, the latter ones in particular if the conference is focused on the performing arts. Before their presentation is approved for the meeting, prospective presenters are typically required to provide a brief abstract of it. Depending on the discipline, presenters may be required to submit a paper of between 6 and 5 pages for peer review by the program committee or a referee of their choosing. Presenters frequently build their talks in particular areas, such as the sciences, on a visual presentation that highlights crucial statistics and research findings. A meeting can have one session at a time or many sessions; the former has just one session at a time, while the latter has multiple concurrent sessions with speakers in various rooms speaking simultaneously. Depending on the conference's focus, social or recreational activities might also be provided. If the conference is big enough, academic publishing houses might set up booths with discounted books. Business meetings for academic organizations or groups may also be held at a bigger conference. Themed conferences, general conferences, and professional conferences are the three types of academic conferences.

Small gatherings with a particular theme are known as themed conferences.

The general conference is a larger meeting with sessions covering a wide range of subjects. These conferences are frequently hosted annually or on another regular schedule by regional, national, or international learned societies.

Large conference with professional themes that is not just for academics but also deals with topics relating to academia. The 'call for papers' or 'call for abstracts', which defines the meeting's subjects and instructs potential speakers on how to submit their abstract or papers, is used to publicise academic conferences. Online submissions are becoming more and more popular.

We will examine all conference-related activities in this work and identify the most important ones to include in a computerized and automated "information system". In this context, the activity entails managing the conference "information system" in a way that minimizes and gets rid of known bottlenecks related to the local and manual handling of such information. The following are some of the conference tasks and workflows that conference management systems support:

- Getting paper submissions (uploading PDFs, gathering metadata from sources)
- Submissions to Zing anonymously
- Gathering the preferred topics for reviews
- Keeping track of conflicts of interest
- Giving papers a reviewer's name
- Making submissions available to reviewers
- Gathering ratings
- Keeping track of review coverage
- Review exchanges within the program committee
- Ensuring the impartiality of reviews (reviewers cannot see other reviews for a submission before they have submitted their own)
- Offering the reviewers a discussion board for each submission
- Review evaluation and threshold-setting
- Reporting program committee feedback and reviewers' comments
- Choice of authors
- Assembling the final, approved versions

➤ *Some Systems Include Further Features in Addition to Just Facilitating the Peer-Review Procedure:*

- Developing the conference's website and schedule
- Registering participants
- Publishing results

Therefore, when seen from the perspective of the conference organizers and managers, it is evident that conference management system is a subset of information management system and also connected to management information system.

C. The Database Infrastructure.

A data infrastructure is created by the database. They offer data storage for one or more organizational operations and one or more functions. Numerous databases based on organizational activity will exist. In order to plan the database infrastructures, it is necessary to decide what information should be saved, how those data should be related, and how access should be controlled. The ability to supply data for both application and ad hoc needs is the end result of database strategy and implementation with database management system. Data warehouses are extensive databases created for ad hoc use.

D. Development of Conference Strategies

Like any other business activity, strategic planning starts with deciding on social responsibility before defining the vision and goals of the company and the means by which they will be attained. The social and economic duties of the organization must be established and communicated to all parties involved from the very beginning of the planning process in order to fulfill these obligations. The goal of the organization for which it works must be decided. Many businesses refer to it as their mission. Setting the organization's goals is the next step after agreeing on the mission or aim. The objective is more precise and has a deadline. The top management uses the goals as a guide for planning the company's operations. The next step is to define numerous goals for the organization after deciding on its mission and goals. The objective is laid out in terms of business outcomes to be attained in a year or two. With the aid of business tools and technologies, the objectives can be measured and tracked. When the objectives are accomplished, the aims and consequently the mission will be accomplished as well. The next step in the planning process is to set targets for more detailed working and reference. The objective of the business is to be translated in terms of functional and operational units for easy communication and decision making. The targets will be the direct descendants of the objective(s) (Doyle, 2000). The management's strategies directly affect whether the aims and objectives are accomplished. An organization's resources are put to use in the game of evolving strategies and winning counterstrategies.

➤ Types of Strategies

A strategy is a precise choice made, typically but not always, in order to allocate resources in a way that advances the organization's mission or goals. While a poor strategy fails to accomplish the goals, a good strategy outperforms the competitors and ensures goal achievement. A bad plan can be corrected and improved, but doing so comes at a very high cost; this is referred to as a strategic failure. A strategy is considered fixed if it only takes into account one point of attack and one technique of attack. A strategy is considered mixed if it employs a variety of tactics to engage multiple fronts. Multiple external influences could be managed simultaneously by a variety of pure business strategies. Therefore, the strategy may apply to any part of the firm and could cover any topics.

➤ Tools of Planning

Long-range, short-range, tactical, or strategic planning all requires managers in the organization to make a number of decisions. Therefore, when we discuss planning tools, we also discuss decision-making tools in relation to planning. Various parts of corporate business planning are impacted by decisions. While planning a firm, there are numerous alternatives, choices, and options available. Additionally, resources are chosen and allocated in the best way possible to maximize earnings. The choice of method is the next step, which involves coordinating the efforts at all scales in the direction of a single objective. As a result, using tools to make decisions is part of planning. These instruments are founded on one or more variables. These

elements are: Innovation: Systemic thinking Analysis of sensitivity and modeling.

➤ Creativity

An individual or group of people's experiences, judgments, and intuitions are the sources of creativity. The only method for handling the decision-making challenge when it arises in a circumstance where there is no precedence is innovation. An individual's conceptual abilities lead to creativity. The following skills are included under conceptional abilities:

- The capacity to quickly come up with several concepts.
- The capacity to shift quickly between frames of reference.
- Originality in analyzing a situation and coming up with new perspectives.
- The capacity to handle a complex interaction between many aspects in a given scenario with clarity and ease.

One who exhibits these abilities is referred to as conceptually fluent. A group of individuals with conceptual fluency, at least in important positions, transforms a company into a creative one. Such a company develops fresh concepts and innovative business plans. Plans are created using experience and conceptual fluency as a foundation.

➤ Systems Approach

The systems approach to planning takes into account all the pertinent variables and how they interact. It leads to an analytical examination of the entire system, generates potential courses of action, and aids in the decision of which is best under the circumstances. It is applied in risky or unclear situations and considers several options for action. It is beneficial to resolve issues.

The systems approach aids in a clear understanding of the situation. The principles of important and non-critical, substantial and inconsequential, relevant and irrelevant, and lastly controllable and uncontrollable are useful for classifying the factors. It evaluates the viability of the technological, operational, and financial options. It investigates the issues with the use of the remedy in further detail. The systems approach generally possesses the following traits:

- It makes use of all the disciplines and knowledge branches.
- To pinpoint the issue, a scientific analysis is used.
- To solve the issue, a model of a complicated circumstance is used.
- To evaluate the choices, it compares cost and benefit.
- It addresses issues where a futuristic temporal context exists.
- It takes into account the surroundings and how it affects the problematic issue.
- Each solution is put to the test for feasibility and rationalism, and it accepts a predetermined standard for choosing the best option.

- If the problem is clearly stated, operations research models are used.

The systems approach is a method of systematic problem-solving that applies the principles of rational decision-making along with scientific methodology.

➤ *Modeling*

A model is a realistic, scaled-down representation of the real world in which only the key elements are highlighted. A model's goal is to simplify a complex situation so that only the most important elements are considered. Models come in a variety of forms. An actual model of a house, a park, a sports complex, etc. could be the model. A scale model that shrinks a large body to a small one could be the model. The model could be mathematical model like break even analysis model, linear programming model, queuing model, network model, etc. (Yoder and Barcalow, 2019).

E. Conference Management System

Every step of the conference organization process needs to be supported by a web application. This comprises handling conference participant registration, assigning reviewers, submitting amended and camera-ready papers, and submitting papers for consideration.

- *Software Interfaces*

The DBMS shall be one of the external systems. To provide database calls independent of the DBMS suppliers, there must be an abstract layer between the system and the DBMS. The JDBC 3.x driver type 4 accomplishes that. Such drivers are usually developed by the corresponding database vendors (e.g. MySQL DBMS provides such a driver) (Plasmeijer & Achten, 2005).

➤ *Product Functions and Features*

- The system must support numerous sub-conferences.
- Special sessions shall be supported by the system.
- Tutorials must be supported by the system.
- The system must support plenary or invited talks.
- Paper submissions will be supported by the system.
- System support for review processes is required.
- The system must support the construction of conference programs.
- Registration must be supported by the system.
- The system will assist in the development of proceedings.
- On-site participation shall be supported by the system.

➤ *Security Requirements*

All user-provided information that will be used in the SQL query must be preprocessed so that a backslash (an escape character) comes before a single or double quote. Otherwise, because these letters are a component of the SQL syntax, the SQL query could be misinterpreted.

- *General Role Consideration*

There are some information and principles which are common to all roles. These are mentioned in this part, and later specific roles are explained (Sarah, 2020).

- *Personal Data*

Personal data includes:

- ✓ Title, name, last name, institution/affiliation, nation, address, e-mail, and areas of interest (choose several from a list of conference interest areas) are all required.
- ✓ Department, fax, phone, and home page URL are all optional.
- ✓ Password and user name. Both the username and the password must be distinctive. They establish a person's duties within the system so that the proper modules can be displayed.

Personal data is presented to all roles. Some roles can have additional items related to their functionality. Users can change their personal data later after login (Plasmeijer & Achten, 2006).

Activating accounts and invitations new roles can be assigned in two ways:

1 An email invitation with an active link to the account activation page is sent by a user with authority. The invited party can next fill out his own personal details. If a person is already registered with the system, all they need to enter is their login and password in order to be given a new position. The user should include the following information about the invited person before sending this mail:

- ✓ Individual's name
- ✓ Individual's last name
- ✓ Individual's email
- ✓ Title (optional)
- Note: This method of registration is used for all subsequent roles.
- By filling up the form and selecting the relevant role, a user with authority provides an account. Form consists of:
 - ✓ Individual's name
 - ✓ Individual's last name
 - ✓ Individual's email
 - ✓ Individual's username
 - ✓ Individual's password
 - ✓ Title (optional)

After filling the form and activating account user sends an email which informs an invited person about filled data and given account. Note: That way of registering is only available to Conference administrator. User gets account when visiting web site.

- *Administrator*

- ✓ *Personal Data*

Administrators of systems have usernames and passwords. The System Administrator has initially predefined these settings. You can get in touch with him via email if you have any system-related issues.

- *Conference Creation*

The system administrator has the highest position and is responsible for ensuring that the system runs well. A new conference must be created by the system administrator. The process of creating a conference involves filling out a questionnaire and the conference data (just the first tag from). The following items on the questionnaire have options of yes or no. The questions are:

- Is there a need for sub-conferences at the conference?
- Are tutorials required for the conference?
- Is there a need for specific sessions at the conference?
- Does the conference call for plenary or invited talks?

The answers to these queries activate or disable certain features.

Additionally, the system administrator must designate a conference administrator. Conference administrators will be able to explore conferences if there are multiple conferences in the database.

- *Browsing Conferences*

A system administrator must have access to the database's list of current conferences. By selecting the desired conference, the user takes on the role of conference administrator and has the power to modify all data.

- *Conference Administrator*

Conference management

All users and conference data can only be changed by the conference administrator. He is granted the following powers: All users can be listed by him. He is able to modify user information and provide new login credentials.

- *Conference Data*

The following conference parameters can be set or changed by the administrator using a form:

- ✓ Title, location, and dates that the conference will begin and end.
- ✓ Date constraints.
- ✓ Author registration
- ✓ Author notification
- ✓ Paper submission.
- ✓ Instructional proposal
- ✓ Submission of the completed paper.

- *Paper Management*

All articles are collected by the program chair (PC) from the corresponding authors. He delivers papers to sub-conference programme chairs (for a specific sub conference) after reviewing paper metadata.

- *Creating Conference Program*

Building the program from sessions is the topic of this section of the paper chair. Each sub-conference PC provides the PC with the prepared session plans. The PC chair compiles all of these sessions and creates the daily conference schedule. He completes these by hand, but a database is filled with all the information relating to where and when sessions are place. A list of sessions may be updated with new events by the PC chair. Such as excursions, lunch breaks, and coffee breaks... The only way to identify these events is by name. The special categories include invited speeches and tutorials.

The PC chair will be able to designate presentation spaces and rooms. The only indication of these rooms is their names. Sessions and events are the building blocks of the program. Each session or event needs:

- Date/day
- Time of beginning (accurate to five minutes)
- Finishing time, to the nearest five minutes
- Allocated space

If there are any timetable conflicts, the system ought to be able to detect them and issue a warning.

- *Notification System*

It is believed that all alerts are sent to PC Chair immediately by email. If the author won't be attending, he or she may notify the organizer that another person will give the paper, obtain an invitation letter showing that the work has been accepted, or request specific presenting equipment.

III. DESIGN

Design entails the synthesis of a system's component parts to produce a new system that gives a solution to a given problem that has already been studied. The design and the tools used have a significant impact on the effectiveness of a solution that is proposed for a problem. It is best to specify the design of the object-database architecture for the construction of an object database system communication at the conceptual level first, using concepts and language that people can easily grasp, to assist assure accuracy, clarity, adaptability, and productivity. The actual database management system (DBMS) used to implement the design can be based on object-oriented logical data model, and the conceptual design may encompass data, process, and behavioral aspects.

In this article, we emphasize the data perspective and presumptively develop a universe of discourse (UoD). A thorough comprehension of the UoD and the ability to articulate that understanding in a straightforward manner are necessary for doing this well. By using plain language and intuitive diagrams that can be filled with examples, as well as by analyzing the material in terms of basic or simple facts, Object-Role Modeling (ORM) streamlines the design process. It offers a conceptual method of modeling by expressing the model in terms of common notions like objects and roles.

The research's chosen application domain or discourse universe is a conference management system. This was chosen because paper reviews in a dispersed context need multitasking.

When an old system becomes unusable, design is an unavoidable process of change; this is when a new system development is required and offered. The system design process will be carried out from two different perspectives in this chapter; the limitations and shortcomings of the manual system that is now in use will be used to develop the new system in order to address those issues. We use PHP to implement the created system. Utilizing a design cue from an existing packaging is an alternative.

A. UoD Design – Conference Management System Design

A conference is a gathering of people where they can present and talk about their work. The topic may be intellectual or relate to other spheres of life. A conference presentation could be collected in writing as academic papers that are then examined by chosen academics from various geographic areas and published as the conference proceedings. A web-based program called a conference management system aids in the planning of academic conferences. This approach is based on conferences and publications and is domain-specific. A conference management system's concept comprises the submission of conference papers, their review, and communication with authors regarding conference specifics. Each submission topic is treated as a thread.

The following features of the E-conf web conference management system will be implemented. The ability for users to:

- Choose the thread in which they want to submit.
- Responding to deadlines for paper submissions
- Examine the posted calls for papers.
- Review papers downloaded, and then post comments on them.
- Send the author your thoughts and any necessary revisions via email.
- Submitting updated versions
- Author acceptance email

➤ *Design Consideration of the CMS*

The primary stakeholders (users) for the conference management system domain are represented by the papers, authors, editors, and reviewers, as well as the conference's program committee and chair. The user of the program chair is depicted as (PC), as are the reviewers of the papers and the publisher of the proceedings. Stake holders' goals are then identified and, for every goal, the analyst can decide, on the basis of the domain documentation, if the goal is achievable by the actor itself or if the actor has to delegate it to another actors revealing a dependency relationship between the two actors, such as in the case of the dependency between Author and PC for the achievement of the goal publish proceedings (Schumacher et al., 2006).

➤ *The Late Requirement Phase*

This is meant to document the modifications made to the domain as a result of the implementation of the system as well as the system's actual characteristics. The phase begins by adding a new actor to the domain model to represent the future system.

The system clearly demonstrates administration of submission, reviews, judgment, and proceedings aggregate to conference coordination in Figure 1, which presents a partial perspective of the model that was created. To handle reviews, assign papers to reviewers and compile individual reviews.

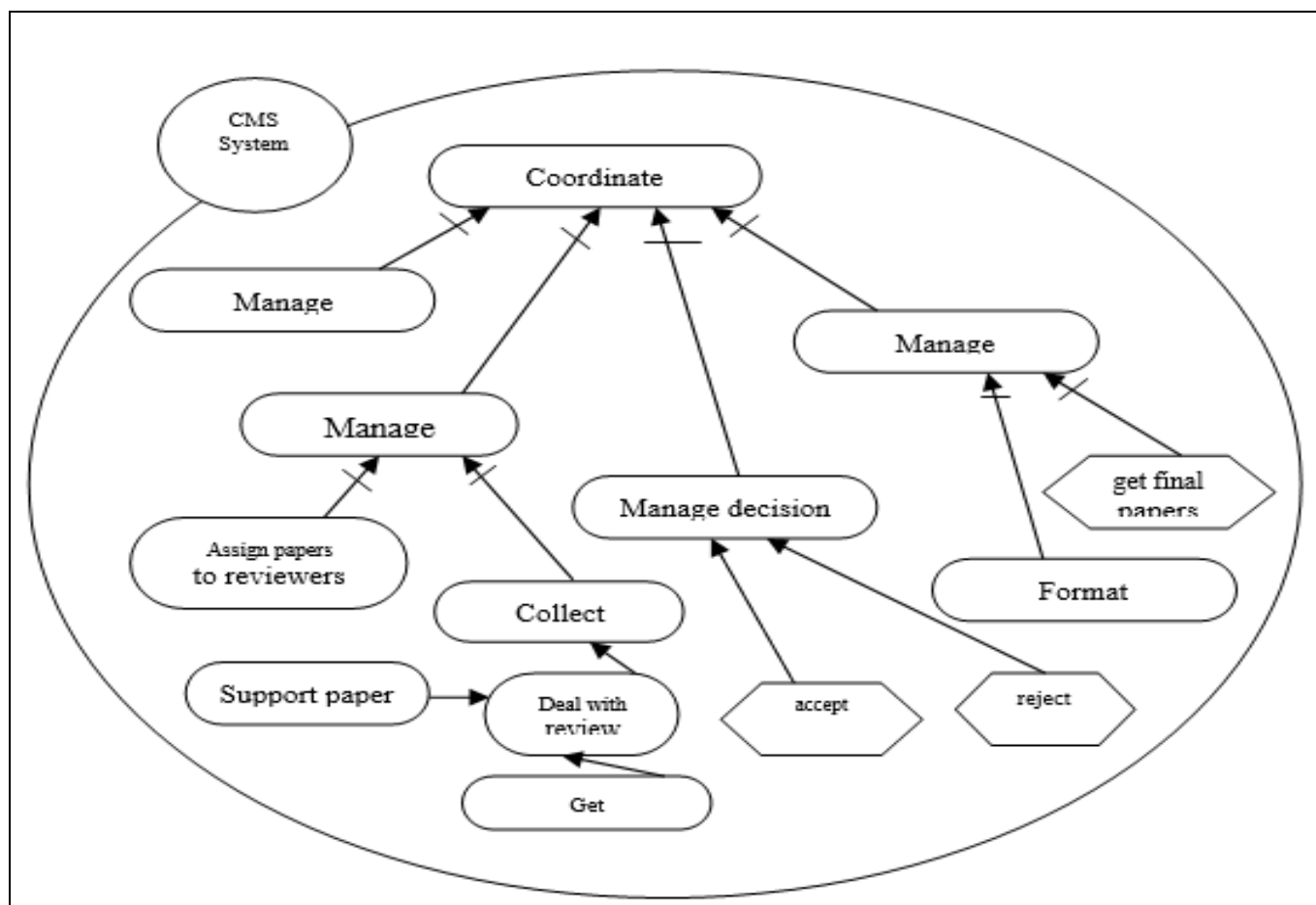


Fig 1: Aggregate Design of Conference Coordination

Final papers, the format of the proceedings, and accept-or-reject judgments are additional aggregation contributors. When these and other system components come together, they create the Conference management system, which advances system development from the components of the analyzed requirement system to the real system.

B. Design Details of the CMS

The new system is built using the conference components and their interactions as identified in the design aggregation system. A database is also utilized as the system's resource base for storing all the data it will need to function.

➤ Existing System Process Design

The CMS used as a use case in the design of the system is first presented using the existing object relational database to illustrate the possible variation with the proposed design system that is deployed on web server so that authors and reviewer can have access all over the globe. The key variation is clear from storage of data to the way the data is manipulated in the system. (Thomas et al., 2021)

Figure 2 depicts the design utilizing the current system. The database in the proposed system design uses a relational model to store simple data, and middleware is used to enable the media server to process objects that are

stored in various locations. The middleware in the system controller can conceptually link the objects when the Author, Reviewer, or Editor interacts with the system since the Controller clearly handles the simple data and allows the objects to be stored individually in the Object data store.

• Proposed System Process Design

The proposed solution is a web application that is installed on a web server and accessible to authors and reviewers everywhere in the world. It has a client interface that the writers utilize to communicate with the computer and other conference organizers. Additionally, the client will be utilized for preconference call_for_paper advertisements, which are typically. The routing application within the system will store the PDF files or other file formats like pictures as files, and the system information associated to the file submission such as the author, data of submission, and other associated information are sent into the database for storage. Authors submit their papers using the web browser in Figure 3. Other users can evaluate the information from their own browsers, including PC users, reviewers, and conference coordinators. The publisher receives the accepted articles to press electronically or in print after the reviewers have finished and decisions have been made on the chosen papers. When the system is put into place, it will assist in transforming academic and organizational conferences by showcasing all of the conferences in their shop on an online

catalogue that is available from any web browser in the globe. By allowing reviewers who are specialists to evaluate the papers because they may be gathered from all around the world, it will also improve the quality of the papers. Since writers will be able to view and publish their papers immediately online and the data and papers will be managed jointly as both objects and databases, paper

submission will also be well received from different regions of the world. The database management system must handle the data and any related objects in a way that promotes simple database access, storage, and querying. The design makes it evident that objects are not divided from simple data as they are in the current system architecture.

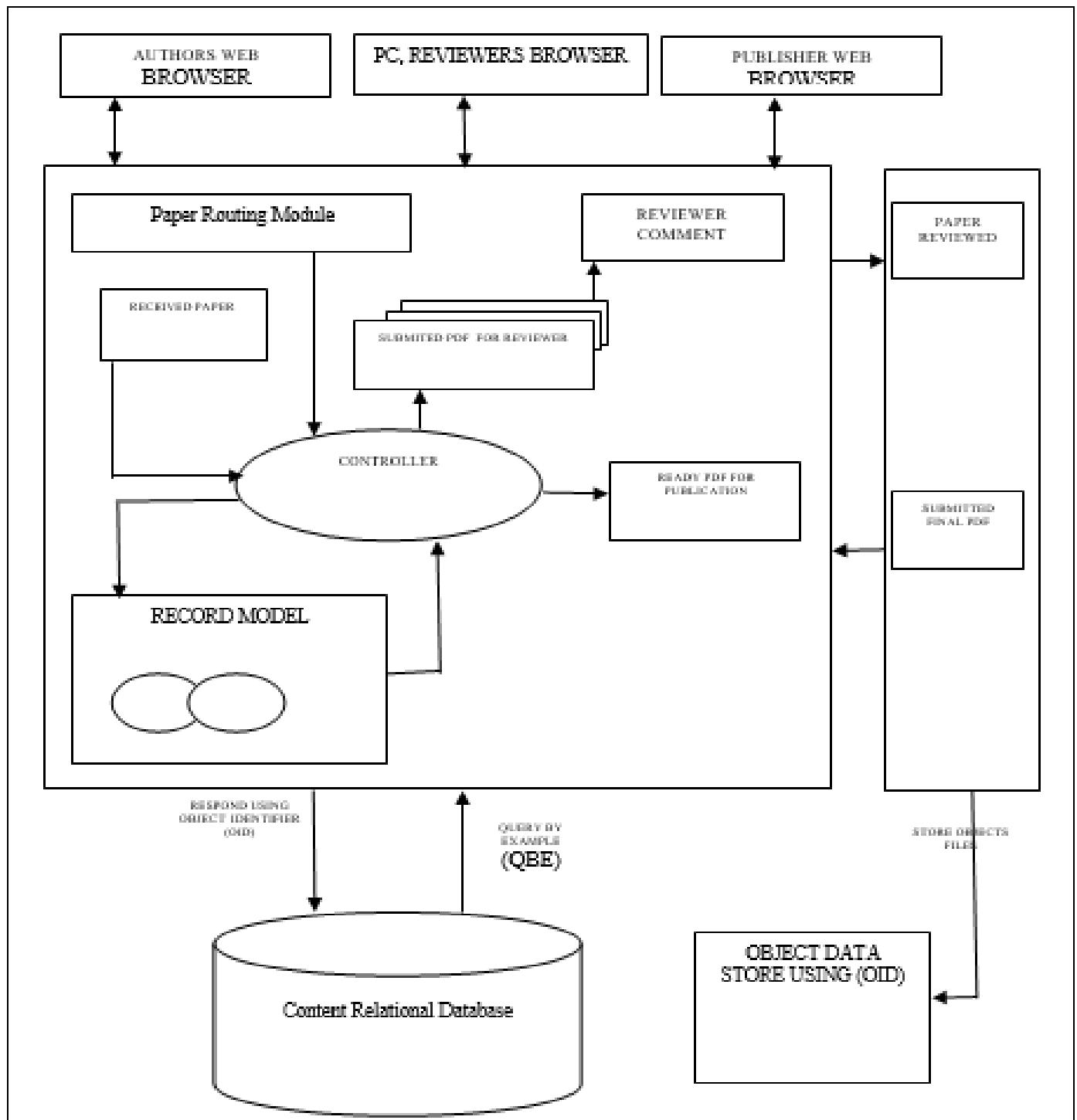


Fig 2: Process Design of the System Using Existing Object Relational Database

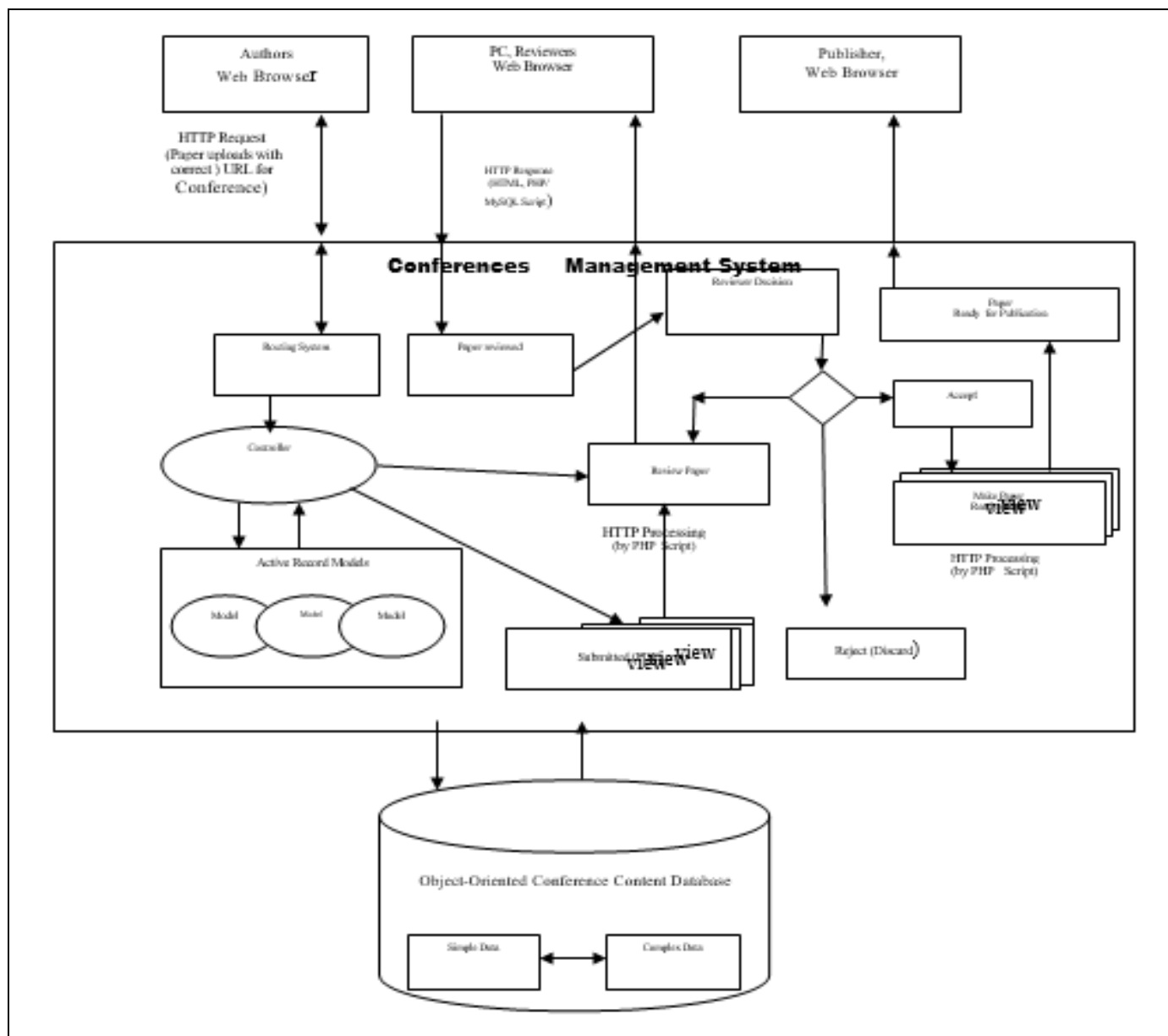


Fig 3: Design of the Conference System using the Proposed System

IV. SUMMARY

This essay has discussed the need for object databases and how they can handle complicated data. The study examined the use of aggregate design to incorporate the elements into contemporary system development. A conference management system was used as a use case in the study to evaluate this idea. The proposed concept, the design, and the execution of the conference management system were all used as the basis for the analysis.

A conference management system improves the delivery of conference services for the purpose of streamlining and automating conference procedures. It offers resources to improve the pre-conference, conference, and post-conference activities. The provision of conference services also includes using information and communication technology to improve access to conference material and submit and review papers. The conference

management system acts as a conduit for the delivery of practical and reasonably priced solutions for bridging the gap between authors, conference organizers, and conference paper reviewers. This makes use of communication technology, which enables the participation and empowerment of conference contributors from all over the world. Additionally, it promotes increased involvement in the conference and conference activities. Hence The creation of conference content using submission and processing of pages, authoring based on content creation, management of the site and its pages, as well as the complete upkeep of conference web applications, is made simple with the aid of conference management systems, which were developed using AMP technology. In general, the CMS system was chosen as a use case because to its extensive database usage and the requirement that these databases connect with all user objects and application objects within the application usage environment.

V. CONCLUSION

In this study, an aggregate design for the creation of object database systems was created utilizing the Conference Management System (CMS) as a use case for object-role modeling. The development of practical object database systems for use in enterprises and organizations will be guided by this model, which will be used as a guide by database management vendors and researchers. The system that has been put into place serves as a tool for conference communication and publication information dissemination that empowers both writers and publishers and fills the gap between the call for conference papers and the release of conference proceedings. By expanding the submission and paper review scopes for submitted papers, this can improve conference quality. Additionally, it offers a level of expertise, information, and conference quality that was previously impossible. Conference planners have an obligation to deliver services as effectively as they can and to ensure that attendees get the most out of their time and financial investments in attending conferences. As a result, creating a conference management system entails more than just creating a web application; it also entails creating a fundamental tool to govern how pages function when information is transferred, allowing for the efficient provision of conference services.

➤ *Contribution to Knowledge*

This project has contributed in the following ways:

- The system also produced object role models, a cutting-edge tool for displaying object database entity relationships in an understandable model.
- To illustrate the ideas put forth in the thesis, we also constructed a conference management system. The program might be helpful in managing conferences, particularly in Nigeria, and programmers can use the system as an example for creating programs that are far more difficult.

➤ *Recommendation*

We advise academics who need to streamline their development processes or deploy systems that require complicated data to look at the research presented in this paper. Academic conferences can also benefit from the conference management system built in this study, and the development approach can be applied to the creation of more complicated systems leveraging AMP technology and other resources used in this paper. The approach created via this research can help improve how efficiently and effectively academic institutions and other organizations manage conferences based on paper submissions. We recommend this study to academic conference planners as well as professional organizations like the Computer Society of Nigeria (NCS) and other groups that plan conferences on various topics for actual deployment in order to reap the benefits for our society.

REFERENCES

- [1]. Doyle, L. (2000) Content Management System work shop Report, Fourth Institutional web management, University of Bath, UK.
- [2]. Eric E. (2003). Domain-Driven Design- Tackling Complexity in the Heart of Software. AddisonWesley, ISBN 0-321-12521-5.
- [3]. Plasmeijer R. and P. Achten (2005). Generic Editors for the World Wide Web. In Cenrtal-European European Functional Programming School, EÅotvÅos Lorand University.
- [4]. Plasmeijer R. and P. Achten.(2006) The Implementation of iData - A Case Study in Generic Programming. In A. Butterfield, editor, Proceedings Implementation and Application of Functional Languages - Revised Selected Papers, 17th International Workshop, IFL05, LNCS 4015.
- [5]. Sarah A. (2020) UISA: User Interface Security Assessment, USENIX Symposium on Usable Privacy and Security (SOUPS) 2020. August 9 -- 11, 2020, Boston, MA, USA.
- [6]. Thomas Fischer, Ahmad-Reza Sadeghi, Marcel Winandy (2021) A Pattern for Secure Graphical User Interface Systems, Schumacher M, Fernandez-Buglioni E., Hybertson D, Buschmann F, and Sommerlad P., (2006) Security Patterns: Integrating Security and Systems Engineering. John Wiley & Sons, 2006.
- [7]. Yoder J. and Barcalow J,(2019) "Architectural patterns for enabling application security," in PLoP'97: Conference on Pattern Languages of Programs.
- [8]. Arbaugh, W. A., Farber, D. J. and J. M. Smith, (1997) "A secure and reliable bootstrap architecture," in Proceedings of the IEEE Symposium on Research in Security and Privacy. Oakland, CA: IEEE Computer Society, May 1997, pp. 65–71.