Development of a modular interface of the computer support system for aircraft design

N M Borgest¹², S A Vlasov¹³

¹Institute of aeronautical engineering, Samara University, 34, Moskow highway, Samara, 443086, Russia
²borgest@yandex.ru
³vlasov.ssau@mail.ru

Abstract. The article summarizes the experience in the field of education of the designer's aircraft with the support of modern information systems, programming languages, databases and knowledge. We consider the problem of interface design assistant job designer based on thesaurus of subject area. We present the existing type of the interface work of the designer and its possibilities. Further ways of development of the program, communication and interaction with the thesaurus of the subject area are considered. The conceptual description of the model of ontological support of the intelligent assistant of the robot designer is shown.

1. Introduction
The article describes the concept and implementation of the interface for automated design support system "Robot-designer". The system described in the article is aimed at automating the preliminary design of the aircraft. A general approach of the organization to the interface is presented, the modules used to compile the interface are described. The relations between the subject of the design process and the CAD meta-model are considered in the appendix to the interface design. The article also considers the formalization of knowledge, design procedures, semantic and mathematical modeling. The automated design system described in the article is based on the thesaurus of the subject area. The thesaurus serves as the basis for the integration of databases, design procedures and interaction scenarios. Formalization of design procedures, achieved in the form of symbiosis of parameterized 3D models of airframe parts, in statistical calculation models and corresponding finite element models, for evaluation of aerodynamic loads and structural efficiency of aircraft weight.

2. Formalization of knowledge
Knowledge management systems are increasingly used in technical design support systems. As a rule, knowledge of the subject area is already available to users in the form of technical information, manuals, drawings and experience of specialists. Nevertheless, the problem of translating existing knowledge into a form suitable for processing is still relevant and often complicates the practical implementation of intelligent support systems [1]. World experience in the creation of knowledge formalization systems shows that the development of such software products, as a rule, faces two main dilemmas:

One expert's problem. More often than not, there is not a single person who is responsible for most of the accepted decisions of the project. The same person usually does most of the data collection. As
the project expands, the amount of data increases, making it necessary to distribute the work. On the one hand, it really reduces the load on the lead specialist, but on the other hand it also complicates his ability to control the project as a whole and reduce the quality of the project information model. Therefore, there is a problem of choosing between unity and distribution of the project document flow.

The problem of tool flexibility. Modern knowledge formalization tools are usually designed to solve specific problems in the subject area, which makes them not very applicable to the formalization of knowledge for the project as a whole. In addition, knowledge can be in various forms in the form of tables, text, or algorithms. The expansion of the functionality of the software product in the field of knowledge formalization usually increases the complexity of use, which in turn leads to a decrease in efficiency. Thus, there is a dilemma of choice between the most productive and the most versatile tool.

A broad range of products has been designed using graph-based design languages. From applications in aerospace [2, 3] through consumer products [4] and off-road machinery components [5] up to automotive [6], the method has been successfully applied mainly in a scientific context.

The scope of design languages has been prototypically extended up to downstream stages of the life-cycle by generating and designing the digital factory for a product in addition to the product itself [7].

In order to preserve and transfer the experience and knowledge of people, the whole system of education for the training of specialists is traditionally built. Unlike humans, "education" of machines is more difficult, since modern computers do not yet have creative thinking, do not use natural language. Thus, our domain and process knowledge must be adapted to machine language.

The task of modeling the activities of the designer requires not only the description of the operations and procedures performed in the project, but also their translation into the form of machine language. Since the initial phase of any pre-project study involves the study of past experience, properties and parameters of already created artifacts, on the other hand, it is important to assess what the results of new research in various areas that may have an impact on the parameters of the future project. In the field of conceptual design of the aircraft, such studies are based on the assessment of trends, the construction of statistical models. To formalize this process, a database is created and updated based on these aircraft, engines, avionics, aerodynamic surfaces and so on. The data stored in the database is used to generate trends that are presented in the design robot to help the designer make a decision.

### 3. Interface “robot-designer” remote use

The interface described in the article is designed for the aircraft conceptual design system based on a meta-model of the aircraft (created in CATIA computer-aided design), a database (currently based on MS Excel), a thesaurus of the subject area (created in Protégé) and the interface itself (developed in C# in Microsoft Visual Studio). An overview of the system interface is shown in figure 1.
The interface consists of several main parts: time frame (1), interactive section (2), General information window (3) and model preview window (4).

The timeline is a scenario of creating an aircraft at a high level of abstraction. Therefore, the subsections represent business processes of a lower level of abstraction [8].

At the initial design stage, the user formulates the terms of reference, carrying out an independent manual input of the main design parameters of the aircraft, such as: aircraft type, range, payload, cruising speed, altitude.

According to the entered initial data, scientific and technical forecasting is carried out. To obtain the forecast, the retrospective series is interpolated, the main tendency of the parameter change during the retrospective period is revealed and extrapolation is made to extend this tendency to the future [9]. Identification of the trend of the parameter change is performed by the selection of a mathematical model approximating the dependence of the parameter on time. The parameters of the selected trend approximation function and the accuracy of the retrospective series approximation are determined using the least squares method.

When you select a certain type of aircraft, the intelligent software automatically generates a range of input parameters, based on statistics and forecast.

Each design stage has its own design modules; the design scenario is made as follows:

For example, for wing design, the process modules are:
- determining the shape of the wing plan;
- determination of the narrowing of the wing;
- determination of the aerodynamic profile of the wing;
- determination of wing mechanization.

The principle of modular programming is to divide the computational process into a number of independent elementary processes that are implemented in the form of software modules. During the design process, a specific module is called, with the necessary parameters.

Each of the processes has its own corresponding module, which loads automatically based on the design of the presets. The design scenario determines the level of user participation in the design process. [10].

Each base module belongs to a certain type, which is determined by its nature (for example, the weather numerical variable must be determined, or a specific type of carrier surface must be selected). The module provides a brief overview of the relevant design process and gives the user hints on how changing a certain parameter will affect the project as a whole.

The parameters of the module are in the database and when it is changed, the entire preliminary design is rebuilt and recalculated.

Tests implemented by the interface, conducted on students of the Department of aircraft design and engineering of Samara state aerospace University, showed a significant reduction in the time required to study the meta-model.

The introduction of a modular adaptive interface can significantly improve the comprehensive quality of the design automation system meta-model based on CAD[11]. Adding an external description can enhance the ability to support macro-primitives, the presence of semantically accurate descriptions will allow specialists of the pre-domain interacting with the macro-primitive internal algorithm and structure without the need to consult with the initial developer.

4. Remote use

Assistant robot designer has the ability to work remotely. Any designer can use the assistant being near the stand of the designer. In this version, the application works in the same Wi-Fi network with the system. The designer can connect to the system, having at hand a smartphone or tablet with access to a Wi-Fi network and a specific application on it. The application also has a user-friendly interface for entering parameters. The only difference is that the user is not shown on the tablet screen parametric 3D model of the designed aircraft, but the design process he can see on the screen of the stand. Figure 2 shows the view of the tablet.
All parameters entered by the user are recorded in the Excel database, calculated, optimized and transferred to the CATIA CAD system.

Back the user will receive a report on his model and a screenshot of the application screen with the model.

The remote control interface (client) is greatly simplified compared to the full version of the stand.

Due to the fact that the developed assistant designer is implemented in a highly specialized subject area, in particular, the preliminary design of the aircraft, the functional lexicon of the user is limited to the thesaurus of the subject area.

Given the complexity of the interface of automated applications, leading CAD developers set themselves the task of developing a system capable of communicating with the user in a trivial language, which would minimize the time required to master the program interface. [12]

References
[1] Wriggers P, Siplivaya M, Joukova I, Slivin R 2008 Intelligent support for engineering analysis using ontology and case-based reasoning, artificial intelligence engineering applications, 20 45
[2] Gross J 2014 Aufbau und Einsatz von Entwurfssprachen zur Auslegung von Satelliten. PhD thesis (Stuttgart- Universität Stuttgart)
[3] Rudolph S, Beichter J, Eheim M, Hess S, Motzer M, Weil R 2013 On multi-disciplinary architectural synthesis and analysis of complex systems with graph-based design languages 62 517
[4] Tonhäuser C, Rudolph S 2016 Individual Coffee Maker Design Using Graph-Based Design Languages, Design Computing and Cognition 16 513-533
[5] Vogel S 2016 Über Ordnungsmechanismen im wissensbasierten Entwurf von SCR-Systemen. PhD thesis (Stuttgart - Universität Stuttgart)
[6] Haq M, Rudolph S 2007 A design language for generic space-frame structure design. Int. J. Comput. Appl. Technol. 30 77-87.
[7] Vogell S, Rudolph S 2018 Complex system design with design languages: method, applications and design principles. Ontology of designing 3(29) 323- 346
[8] Wriggers P, Siplivaya M, Joukova I, Slivin R 2008 Intelligent support of engineering analysis using ontology and case-based reasoning Engineering Applications of Artificial Intelligence, 20 709-720
[9] Borgest N M et al 2015 Robot-designer: on the road to reality Ontology of designing 4(18) pp. 429-449
[10] Myshkin L M 2006 Forecasting the development of aviation technology: theory and practice (Moscow - Fizmatlit)
[11] Borgest N M, Korovin M D 2016 Formalization of design procedures in the engineer's educational strategy ACSR-Advances in Computuer Science Research 51 524-527
[12] Borgest N M, Vlasov S A, Korovin M D 2015 The Implementation of remote control parameterized three-dimensional model of the aircraft using a client-server application *Open Semantic Technologies for Intelligent Systems* **34** 417-420