Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Engineering Concept of Virtual Automation Networks

Christian Diedrich[1], Martin Hoffmann[2], Harry Hengster[3]

[1] Univ. of Magdeburg, Institute of Control Systems
Universitaetsplatz 2, 39106 Magdeburg / Germany
christian.diedrich@ovgu.de

[2] Univ. of Magdeburg, Center Distributed Systems at IAF
Universitaetsplatz 2, 39106 Magdeburg / Germany
martin.hoffmann@ovgu.de

[2] Schneider Electric GmbH, Automation Business Unit, System Consistency
Steinheimer Straße 117, 63500 Seligenstadt / Germany
harry.hengster@de.schneider-electric.com

Abstract

This paper presents the concept for the engineering of Distributed Control Systems (DCSs) based on heterogeneous communication systems as defined in the European Integrated Project "Virtual Automation Networks" (VAN). This heterogeneous communication characteristic has to be faced by the engineering concepts and tools.

The VAN project has defined a common engineering workflow as basis for a VAN information model. The information model covers the functional, topological and network view and their relation to each other at detailed level. It integrates the VAN service specification on the basis of the OSI Reference ASE model.

The paper presents also the mapping of the engineering concept to FDT/DTM and TCI technologies in general which is used to evaluate the information model and the usefulness of the VAN information model and the interaction of the engineering tools with the VAN devices.

1. Introduction

New technologies for devices, components and systems in distributed control systems force the engineering and its workflow to adapt their methods and tools. The paradigm of Virtual Automation Networks integrates existing public, private and industrial communication technologies to a scalable system with defined quality of services. The resulting engineering has to meet all existing engineering steps and tools which are faced with new functionalities and features. Within the European funded research project VAN (Virtual Automation Networks) solutions for the usage of LANs and WANs – public and private, wired and wireless – forming Virtual Automation Networks (VANs), for application in industrial domains as mentioned above will be investigated and developed. The concept of Virtual Automation Networks is to provide a reliable and secure end-to-end industrial communication connection which is similar to Virtual Private Networks (VPN) known from the office world.

Today there is a variety of different engineering tools used for a manifold of tasks in different applications. Engineering tools can be classified based on the performed functionalities (e.g. composition, parameterisation, programming and monitoring) to get a structured view of the engineering tools.

Additionally, the engineering tools can be grouped with respect to the devices (e.g. PLC, remote IO, drives, transmitter, HMI and network components), for which they can be used and the life cycle phases (e.g. modelling, planning, configuration, commissioning and operation) the tools are mostly used in.

A single tool can be assigned to multiple functional classes if it serves for different purposes. Also it can be assigned to multiple device classes if it can be applied to different types of devices. Analogously: different life cycle phases

This paper presents the concept for the engineering of Virtual Automation Networks as developed in the VAN European project (2).

The introduction of the VAN concept and the ideas behind the VAN project will be given by other papers of this invited session (1). Therefore it can be skipped in this paper. For further information please have a look at the other once.

The fusion of local and wide area networks between the geographically distributed parts of the automation functions over heterogeneous networks and the use of dedicated standard IT -technologies are new objectives for the engineering introduced by VAN.

A main scenario of a VAN is shown in Fig. 1. It shows the definition of so called VAN domains. The VAN domains can contain both office networks and industrial networks. This connection can be realized via a public communication environment which can be wired and wireless. This means for the engineering that the application and communication configuration of the individual network segments (industrial, office, private and public) has to be engineered as well as the overall application of the VAN domains including the network transitions.

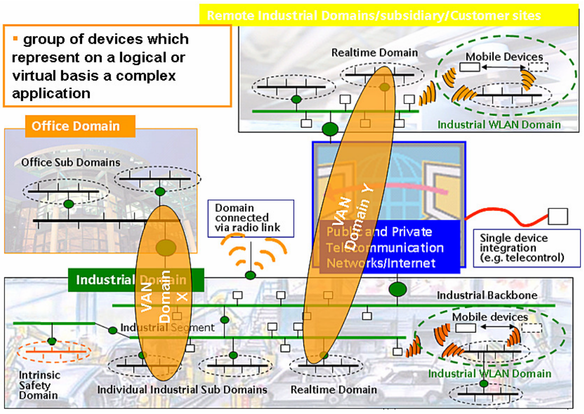

Fig. 1. Virtual Automation Network

The paper proceeds with the description of the starting point of the VAN engineering following by the engineering workflow. The engineering needs an information model which is introduced. A proposal of technologies used to realize the concept is described before a synopsis concludes the paper.

2. Basic artefacts of the VAN engineering

VAN Device Classes

The devices within a VAN domain belong to a well defined set of VAN device types. Each device type contains specific characteristics and a set of functions, e.g. a VAN automation device to fulfil an automation function or a VAN access point to connect different domains over a heterogeneous network.

To fulfil its functions, a common architecture of a VAN device was developed also to obtain the integration into a VAN.

Each VAN device represents mainly two aspects. First it provides functions within the automation application process. Second it provides a connection to a VAN network. It will be shown later that this is an important separation.

Devices with non VAN communication technologies can be integrated into a VAN domain by using a VAN Proxy Device.

Based on the definitions of the ISO OSI Reference model (10) and the harmonised specification format of industrial communication standards in IEC 61158 series and IEC 61784 series VAN specifies the top layer access point using so called Application Service Elements (ASEs). ASEs are the model describing the service object of the communication protocols of layer Seven. They contain the service primitive syntax and semantic. Among others they act as container for configuration and diagnosis data. To access the ASE objects, web services are used as a common access and communication technology. The web service technology is a state of the art approach for communication in heterogeneous network environments, to connect objects implemented on different platforms.

Within the VAN environment, the web service technology is used for device configuration (e.g. configuration, diagnosis, connection establishment), for tunnelling of data. The defined device types (2) are:

- *VAN Access Point (VAN-AP)*

A VAN device, which belongs to a class of VAN-APs, contains the VAN services and protocol implementation. The function of a VAN-AP within the VAN network is to connect VAN domains. They do not have an automation function or an automation application process.

- *VAN Automation Device (VAN-AD)*

VAN-ADs are the devices which provide the (automation) functionality in the VAN context. For this purpose, each VAN-AD contains at least one automation function or application process.

- *VAN Proxy Device (VAN-PD)*

A VAN-PD contains the VAN capability to "speak" VAN to work in the VAN proxy application. It is a proxy for automation devices in a VAN application context which can be accessed within the VAN domain.

- *VAN Virtual Device (VAN-VD)*

Devices which have no VAN network capabilities but automation function within the VAN context. VAN-VDs can be integrated by using a VAN-PD.

Addressing within a VAN

All VAN devices obtain a VAN name within their VAN domain to guarantee the accessibility within VAN. General aspects within this context are:

- The VAN domain is a name based domain (URL like) with a unique name space. This provides an easier handling instead of IP based addressing for the engineer.
- The naming conventions within a VAN domain shall conform to the DNS unique indicator and to IEC 61158 type 10 naming definitions to avoid naming conflicts.
- Each VAN device shall have a unique name according to the defined naming conventions.
- Address resolving in a VAN with sub domains and communication paths with different IP-subnets is done by DNS service.

These device type definitions, naming conventions and ASE model for the VAN protocols access are essential aspects for the engineering concept.

3. Engineering of a VAN System

The state of the art engineering method for networked (and by the way for non networked systems also) is characterised by sequence of modelling, planning, configuration, commissioning and production (Fig. 2). Modelling and planning covers the functional aspects and defines the requirements of the intended application as well as the selection of used technologies (11). Configuration, commissioning and production cover the implementation aspects. Both aspects have to be linked i.e. the results of the functional requirement specification have to be available for the implementation task. The VAN engineering concept (4) defines for this link a so called VAN Instance Model. The VAN Instance Model is derived from the VAN Information Model. It contains the system design including the definition of the quality of service for the communication connections. The VAN Instance Model represents the information necessary for the engineering of a VAN system which is described below.

In the scope of the VAN concept two types of tools are defined for engineering activities:

- The VAN Engineering Client System (ECS) to cover all engineering aspects of the complete automation system.
- The VAN Engineering Client Device (ECD) which is used for the configuration and commissioning of a single VAN device.

The phases of the engineering are characterised as followed:

Modelling

The first step in a VAN engineering process is to create a model of the customer's requirements. A formalised model provides a solid specification for the planning phase. VAN applies UML for the formalised description of the VAN Instance Model. It includes also the relation between these requirements and the VAN specifics, e.g. relations between requirements and communication parameters or device configuration

As essential basic for all further work the information model represents the following views (4):

- Functional view: formal representation of the results of the modelling and planning phases of the engineering process.
- Topology view: basic configuration of a VAN device or a group of VAN devices
- Network view: correlations between a VAN domain and device specific classes.

The functional model of a VAN divides the holistic system into function components and describes the interactions of these function components. The scope of this functional model in VAN is the view of data exchange and data transmission between the function components. The reason for doing it in this way is the separation between the functional design and the device implementation.

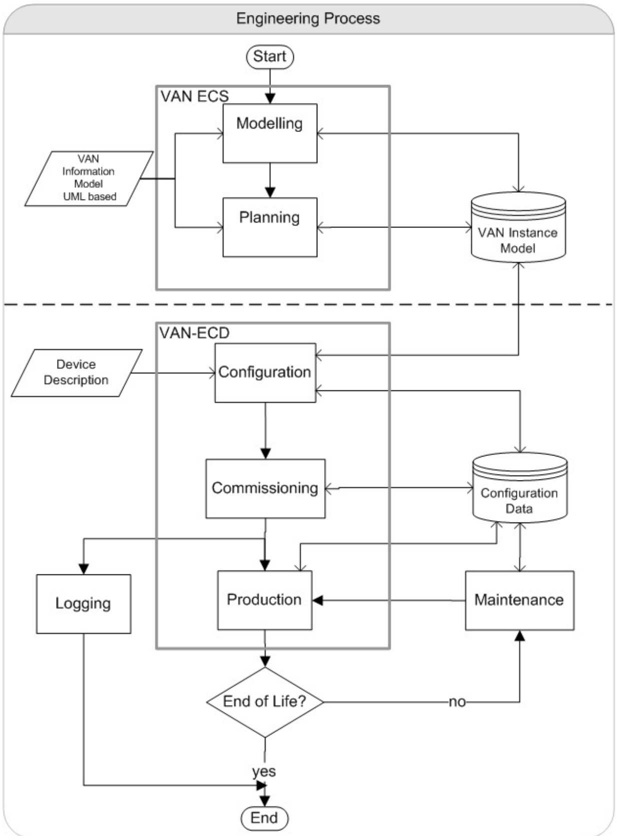

Fig. 2. VAN Engineering Workflow

Planning

During the planning phase the devices and network technologies of the automation system are chosen which are based on functional and non-functional requirements and the available devices with their characteristics. The network topology for the plant and the hierarchy of VAN domains is also defined.

VAN planning is based on a strict separation of functional design and device implementation of the functions. This allows a top-down-approach. (4, 13)

Configuration

In this phase the network settings are completed and the devices are configured. The engineering tools can be used for configuration in offline mode based on the information provided by the device description. For parameterisation and download of configurations and programs a connection to the devices is established via a direct link or through the network.

After completion of the configuration for each device the corresponding engineering tool has to store the new or updated information in the repository. The configuration data is not saved in a central database but each engineering tool can use an independent repository.

Commissioning

To install the complete system all pre-configured and pre-assembled devices and subsystems are combined step by step. The engineering tools are used in online mode, i.e. they are connected to the automation system. For each composition adjustment of the configuration parameters may be necessary. The diagnostic features of the engineering tools are used to identify and solve problems. The engineering tools are connected to the devices through the system's network or via a direct link.

Production and Maintenance

The production and maintenance phases cover the operation of the system. During production the engineering tools are used in online mode for diagnostic of the automation system. During maintenance intervals the engineering tools are used to verify and adjust settings, like re-calibration. Moreover data acquisition is requested and logging of operator activities and modifications on the configuration may be necessary (e.g. due to regulations of FDA).

4. The VAN Information Model

VAN specific engineering tools are covering modelling and planning of the automation systems, configuring and parameterisation of automation components such as VAN devices as well as identification of VAN domains and sub-domains. The VAN architecture offers a large range of ASEs, which attributes have to be parameterised using web services. Web services are the implementation means for the interaction between the engineering tools and the VAN devices. The data resulting from modelling, planning, configuration and commissioning are part of the project data base and have to be written to or read from the VAN devices. Therefore the structure of the engineering data should follow the VAN architecture.

The VAN information model (5) represents all information necessary for the engineering of a VAN project. A VAN project is logically an instantiation of a selection of VAN devices in a VAN domain. All defined classes inside the information model correspond either to a defined functionality of the system architecture or to the definitions of this VAN communication protocol specification.

The model is developed in UML (6), a well accepted design technique supported by several UML tools on the market. The exchange of information between different UML tools is based on the standardized XMI format (7). XMI is a XML based schema definition, which is supported by most important UML tools. This XMI based representation is used for the VAN information model.

The top layer of the information model separates the functional, topology and network view (Fig. 3).

All classes of the information model are derived from a *VanObject* class which provide basic attributes such as UUID for unique instance identification.

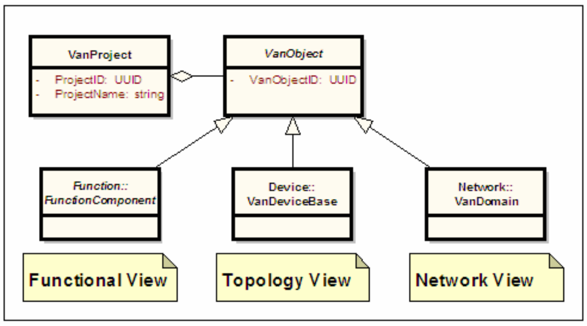

Fig. 3. Views of the VAN information model

The VAN project class aggregates all data of all views in the project data base. Therefore the *VANProject* class has according relations to the parent classes of all three views (*VanDeviceBase, Function-Component, VanDomain*). The views structure the entire information model in the roles which the engineering tools have to offer to the operators.

The relations between the parent classes of the views are modelled as associations. These associations are instantiated during the engineering process activities. The instantiations are defining:

- which functions are implemented in which VAN devices,
- which VAN device is assigned to which VAN segment,
- which VAN segment is allocated to which VAN domain or sub-domain.

Functional view

The functional view represents the results of the modelling phase in terms of the application (Fig. 4). The VAN project proposes a component based approach for the application design. Components are connected via data connections which perform a function block application. The information model contains the necessary objects such as components (*VanMainFunctionComponent*, *VanFunctionGroup* and *VanFunctionBasicElement*) and data connections.

Topology view

The topology view represents the VAN devices, certain roles of VAN devices, the relations of the VAN devices to the network segments and the VAN access point (Fig. 5).

VAN devices are implementing functions (*VanMainFunctionComponent*, *VanFunctionGroup* and *VanFunctionBasicElement* from functional view). VAN devices have configuration data which are described as aggregation to *VAN_ASE_Parameter*. VAN devices can use several network technologies (*VanNetworkTechnologyLayer*).

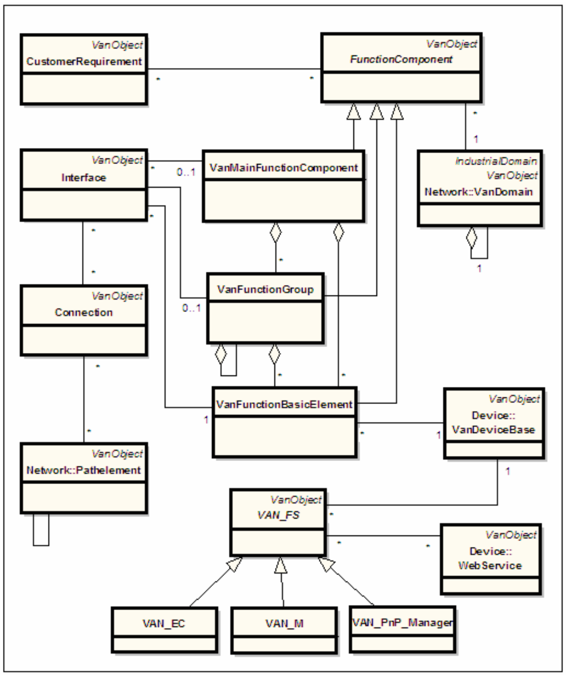

Fig. 4, Functional View information model

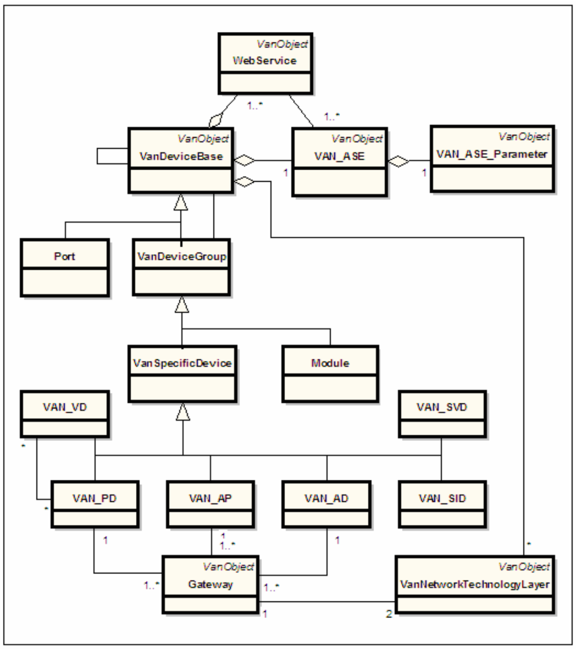

Fig. 5. Topology View information model

Network View

A heterogeneous communication system needs a network view similar to the view of system administrators of office networks additionally to the today's common functional and topology views.

The network view represents the hierarchical structure of VAN domain and sub-domains as well as their relation to network segments (Fig. 6). The hierarchical structure of VAN domains and sub-domains is realized by the aggregation *subdomain* of *VanDomain* to itself.

Two types of network domains need to be distinguished: office domains and industrial domains. As each VAN domain is an industrial domain, *VanDomain* is inherited from *IndustrialDomain*. The relation to the office domain is represented by the association between *VanDomain* and *OfficeDomain*.

The assignment of a VAN device to the VAN domain and sub-domain is given by the association between *VanDeviceBase* and *Network_Domain*, which is inherited to the different types of VAN devices (inherited from *VanSpecificDevice* in the topology view) and to *VanDomain, respectively.*

A VAN device can be connected to different network segments. The possibility to have different types of the communication cards is represented by the aggregation to *VanNetworkTechnologyLayer.* Via each communication card connections to different segments of the corresponding network technology are possible. This is represented by the association between *VanNetworkTechnologyLayer* and *NetworkSegment.*

Moreover, the network segments are also assigned to VAN domains and sub-domains, which are represented by the association between *Network-Segment* and *VanDomain*.

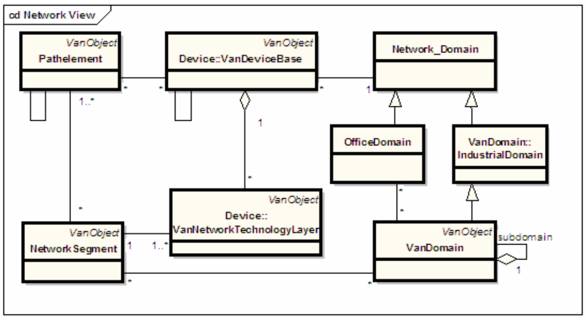

Fig. 6: Network View information model

The attributes of all classes are not shown. They are defined within the ASE specification of the various VAN protocols and they are part of the VAN information model.

5. Technological Solution

For the architecture of the VAN Engineering Tool two different concepts are presented. The first concept targets the development of a stand-alone VAN Engineering Tool, which is independent from the existing engineering tools present on the market (see also (12)). The focus of the second concept is to facilitate the enhancement of existing engineering tools. This integrated concept is elaborated to allow the use of standardized technologies and interfaces as FDT/DTM (8) and TCI (9) (Fig. 7). In the scope of the VAN project the role and a specification of a VAN Device Description (VAN-DD) is worked which covers the new features of VAN devices and builds a bridge to the existing device descriptions. Because of

the limited numbers of pages this paper contains no explanations about the FDT and TCI technologies. Only their specific use is mentioned.
The Tool Calling Interface (TCI) permits to call one software tool from another tool and to pass parameters between these tools. In the integrated concept TCI is used to couple an engineering system with a commissioning tool of a specific device of two different manufacturers. Within the scope of the VAN project a VAN Device DTM and a VAN Communication DTM are provided according to the FDT/DTM specification. These two elements are used by the commissioning tool in order to perform the configuration of the VAN related attributes of these devices and to realize the communication to the VAN devices.

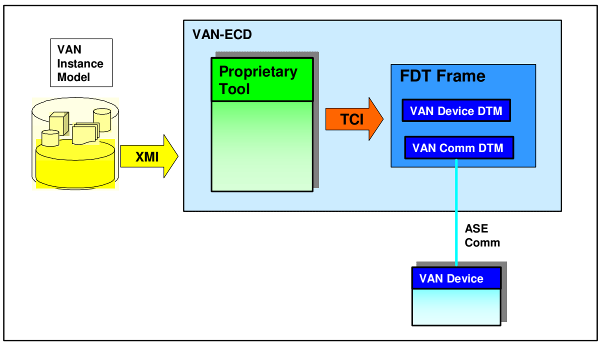

Fig. 7: Integration with TCI and FDT/DTM

The stand-alone and the integrated concept will be used for the realization of first prototypes of VAN Engineering Tools. Therefore, a simplified use case of a VAN system is used in the VAN project, in order to derive a VAN Instance Model of reasonable size and complexity. This instance model is used to validate the architecture and interfaces of the two concepts by prototypes of VAN Engineering Tools for each concept. Based on the prototype for the integrated concept also the data exchanged between the tools is analyzed.

6. Synopsis

VAN focuses on heterogeneous networking of production and manufacturing processes. These heterogeneous characteristic has to be faced by the engineering concepts and tools. Main focus is oriented to a balance between well established technologies and tools and new necessary system components to provide the bridge between the industrial segments.

The today's engineering process can basically subdivided into the phases modelling, planning, configuration, commissioning, production and maintenance. Existing tools are well advanced with respect to the phases configuration, commissioning, production and maintenance. But there is a lack of methods to derive a formal model of the automation system from customer requirements systematically and refine this model for the planning of the automation system continuously.

Therefore, the VAN project has collected system requirements for the engineering, has defined use cases and has developed a VAN information model. The information model covers the functional, topological and network view and their relations among each other at detailed level. The details are derived from the VAN specifications which are based on the ASE model.

Scope of the project is among others the development of prototypes and demonstrators which enhance existing tools and show the interaction between different tools, the usefulness of the information model and the interaction of the tools with the VAN devices.

ACKNOWLEDGMENT

The paper has been supported by the "Virtual Automation Networks" project (FP6/2004/IST/NMP/2 – 016696 VAN) funded by the European Community under the "Information Society Technology" Programme (http://www.van-eu.eu). The authors want to thank the whole VAN project team.

REFERENCES

(1) P. Neumann, R. Messerschmidt, A. Poeschmann: "*Architectural concept of Virtual Automation Networks*", IFAC World Congress 2008, Seoul.

(2) VAN project page, *http://www.van-eu.eu*

(3) Deliverable D02.2-2 of the VAN project "*Specification of the Open Platform for Automation Infrastructure*", The VAN Consortium, November 2006

(4) Deliverable D08.2-1 of the VAN project "*Specification of engineering process*", The VAN Consortium, June 2006

(5) Deliverable D08.4-1 of the VAN project "*Engineering Tool Integration Concept and Tool Interfaces*", The VAN Consortium, July 2007

(6) Object Management Group: *UML 2.1.2 Superstructure Specification*, 2007

(7) Object Management Group*: XMI 2.1.1 Specification*, 2007

(8) IEC 62453 series CD2: *Field Device Tool Specification*. Geneva, July 2007.

(9) PROFIBUS International: PROFIBUS PROFINET Specification: Tool Calling Interface, Version 1.0, October 2006, Order No: 2.602

(10) International Standard Organisation: ISO/IEC 7498: Open Systems Interconnection Reference Model, 1994.

(11) Ch. Diedrich, G. Franz, K.-H. John, J. Krause F. Poignee: Support of Control Application Design Using Digital Design and Planning Of Manufacturing Cells. IFAC WC, Prague, 2005.

(12) M. Hoffmann, M. Muehlhause, M. Chiari, C. Schwab: Uniform Engineering of Distributed Control Systems - The VAN Approach, ETFA Congress 2007

(13) Deliverable D08.1-1 of the VAN project "Overview on existing engineering tools and Requirements for engineering tools for the VAN platform", The VAN Consortium, January 2006