

# Milan Ignjatovic

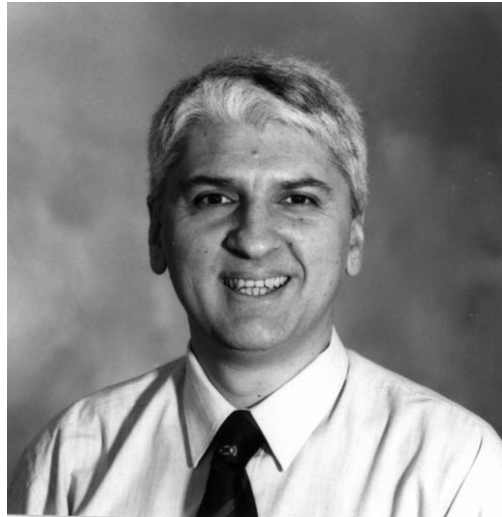
ProSoftwarica GmbH  
Technoparkstrasse 1  
CH-8005 Zürich  
+41 (0) 79 376 59 11  
info@prosoftwarica.com

MSc Electrical Engineering  
University Belgrade

Postgraduate Studies  
Computer Science  
University Belgrade

Software Engineering Consultant  
Software Engineering Trainer & Coach

E-Mail: mig@prosoftwarica.com



## Professional background

2006  
President and Founder ProSoftwarica

1998 – 2006  
Zuehlke Engineering AG  
Software Engineering Consultant,  
Architect, Coach and Trainer

1995 – 1998  
Metromec Software AG  
Software Engineer,  
Project Leader

1989 – 1995  
Sulzer Chemtech AG  
Software Engineer, EDP Analyst

1986 – 1989  
Aerospace Institute Belgrade  
Software Engineer / Analyst

1984 – 1986  
Research and Development Institute  
Belgrade  
System Engineer, Acquisition and Control

1984  
University of Belgrade  
MSc Electrical Engineering

## Major areas of expertise

- ✓ Management of Complexity, Abstraction, Separation of Concerns, Decomposition, Model Driven Development, Software Product Lines
- ✓ Requirements Engineering, Use Case & Feature Modelling,
- ✓ Architectural definition and evaluation
- ✓ Software Development Processes
- ✓ Design of distributed Enterprise, Embedded and Real-Time systems
- ✓ Design Patterns, Refactoring Technique
- ✓ Object-oriented Software Metrics
- ✓ Middleware: Corba, Connexis, Tao, Ace, Zen, JacORB, IDL
- ✓ Natural Language Processing: GATE
- ✓ Tools: Topcased, Papyrus, Eclipse, oAW, MetaEdit+, IBM Rational, Rhapsody, Enterprise Architect, MagicDraw, Poseidon, Together, Visual Studio
- ✓ Platforms: VMS, Unix, Windows, Linux, Jbed, VxWorks, PERC, VenturCom, embedded NT, DCS-RTOS(Yokogawa)
- ✓ Languages: UML2, xUML, QVT, ATL, XSLT, Mofscript, AOP, Xpand, Xtend, Check, Java, Javascript, C/C++, Ada, Delphi, APT Graphcet, Modula2, Pascal, Basic, DCL, TPU, Fortran, Assembler
- ✓ Web technology & Portal architecture: Glassfish, WebSphere, WebLogic
- ✓ Application integration & web services
- ✓ Database management systems: Hibernate, OO2RDB mapping, Oracle, Informix, SQL Server, MySQL, HSQL

## Project examples

### *Development of JEE Application Platform*

A large Swiss bank develops and operates a massive JEE based platform including a reusable software stack. The objectives are to provide a reliable platform based on JEE technology and achieve a high level of reuse for front-end applications, currently numbering in hundreds. Reuse is understood as a cross cutting concern affecting the full software development lifecycle of in-house components.

Governance of in-house, as well as 3rd party components is also a related concern.

The first task is to assess the current state of the platform with respect to its objectives and propose solutions for overcoming identified problems. This is accomplished by an introductory study of abundant documentation, acquisition of anecdotal evidence and stakeholder interviews. The analysis reveals several emerging, wicked problems, hard to foresee and manage early enough, in order to mitigate all involved risks. The applied governance models based on independent component life cycles without clear ownership and accountability, leads to quality problems. Solution is found in introducing quality gates to synchronise independent lifecycles and guarantee required component quality attributes, thus enforcing service level agreements between applications and their customers. Further problems and solutions include topics such as platform layering, mandatory and optional usage of components, component specification and qualification, conformance to licensing models, deprecation policy, immature interface evolution and obsolete user documentation.

The second task is to provide an architectural solution to extend the front-end application dominated JEE platform, into providing enterprise strength SOA services. Gap

identification is driven by an analysis of missing pieces within the architecture, processes, methods, tools and technologies, currently available on the platform, with respect to managed services and needs thereof. More than 20 stakeholders are interviewed, their input critically analysed, redundancies eliminated and structured into high level requirements and use cases. Documented gaps include OLAs for testing environments, location transparency, dynamic load and capacity management, SLA control (sustainability of service quality attributes), complete service dependency information, service monitoring, service retirement, cost accounting, separation of testing environments, monopolising test environment resources and end-to-end business transaction monitoring. Architectural solution is accompanied with a rationale for refactoring, defining new centralised processing components aiming to factor out redundant or similar processing across the entire enterprise.

The third task is the elaboration of a concept and architecture defining a standard solution for distributed caching on the JEE platform based on Oracle Coherence. The need is driven by a number of new applications migrating to the JEE platform that require distributed caching support. Initially, a number of common use cases are identified shared by the migrating applications. The elaborated architecture extends the existing platform architecture to satisfy the identified use cases. The result is a fast and smooth migration of a number of applications to the JEE platform with standard platform support for distributed caching functionality.

Technologies: JEE, Oracle, WebLogic, Coherence, Ordering Tool, Eclipse, Maven, UML/MOF, Windows, monitoring, logging and incident management.

### *Emerging System Paradigm*

The effort for human management of N evolving components, using conventional technologies, is non-linear. The conceptual idea behind the Emerging Systems Paradigm is to linearise such task complexity.

ESP splits a single complex executable system into a set of "super-loosely" interacting, individually manageable and executable components. ESP captures the evolving system interaction, resulting from natural system evolution, into a separate interaction model, enabling the "super-loosiness" feature of the system.

ESP processes the "super-loose" component set programmatically to deliver the executable target system. Thereby, the target system has an emerging character to the individual executable system modeller. ESP is equipped with following features:

- Non-invasive modification of source models by multiple extension models.

- Executable target model is composed, both statically and dynamically from several input executable models and interaction models.

- The composition process is driven by the interaction model and business rules.

- ESP scales well by enabling programmatic processing

- ESP is robust, fully automated, therefore well suited for human exploitation.

Therefore, ESP enables simultaneous management of N interacting components to be reduced to the management of each component in isolation from the others, thus linearising task complexity for the system developer.

Technologies: J2EE, Eclipse, Topcased, MOF, UML, UML Profiles, XML, Mofscript, Business Rules, Swing, TouchGraph, SQL, MySQL, Windows.

### *A new generation of Road Construction Machinery*

A large manufacturer of road construction machines is preparing to develop a new generation of systems. The elaborated vision covers embedded systems, the control centre and supply chain information processing. A new reference architecture is elicited for the various vehicles and respective embedded systems aiming to reduce complexity and maximise reuse. The control centre is furnished with functionality ranging from on-line data acquisition to automatic task dispatching and early warning capabilities.

Additionally, an assessment of existing real-time embedded systems is carried out to define software quality of existing components. The assessment also answered questions such as which software parts are reusable and which need to be reworked to be incorporated into the new reference architecture.

Delivered benefit to the client:

A working prototype of the reference architecture is built demonstrating concepts and benefits. The elaborated concepts, reference architecture and assessment results, provide a framework for project sizing i.e. project plan, resource and budgeting needs.

Technologies: RUP, Product Lines, Requirements Engineering, UML2, Topcased, C, C++, Windows, Eclipse and 16-bit microprocessor tools, Metric Analysis Tools.

### *Web Service and Confluence Solutions*

A large Swiss bank has demanding needs for Web Service definition and realisation. This need is driven by the high complexity of the underlying communication infrastructure and heterogeneous subsystems, varying in technology from mainstream to legacy. The existing software solution is not well suited for this task because it does not automate the complete

workflow. For example, during web service specification and artefact creation, the engineer is forced to execute repetitive, tedious and error-prone manual tasks resulting in schedule delays and realisation bottlenecks.

In order to alleviate these problems, an automation solution is realised based on custom XSD and WSDL artefact generation. A central UML repository is realised by porting and re-engineering the existing models into well-structured service packages. Each service package and elements thereof are furnished with meta-data which is defined within UML profiles. Artefacts, XDS and WSDL service descriptions, are generated using type-based recursive drill-down algorithms by either referencing or de-referencing elements according to custom requirements of the target subsystems e.g. de-referenced XSD element description for Cobol Copybook generation.

For presentation purposes, models are programmatically harvested and only meaningful i.e. meta-data defined end-user information is plucked. Further down the processing chain, this information is packed into SQL statements, typically Oracle MERGE statements, which are executed against an Oracle Database to populate the Confluence DB schema. In this way, the modelled web services are shared on the intranet with the end-users using a Confluence system. The process of information sharing is automatic and results in the generation and execution of many thousands SQL statements. End-users access the published Confluence information to check and verify existing web service specifications or to guide themselves during new web service definition.

Technologies: proprietary software development process, Enterprise Architect, Eclipse, Topcased/Papyrus, XSLT, UML and UML Profiles, ATL, Mofscript, SQL, Oracle, Wiki.

### *Enterprise Portal Solution*

A large Swiss bank engages in a project aiming to deliver a web portal system as a solution for its numerous external asset managers.

The task is to define and specify the enterprise system architecture by elaborating the prescribed documents and getting them through the internal software development process.

Furthermore, a kick-start with the implementation team is to be accomplished and the project safely guided into the realisation phase.

Technical specifications such as non-functional requirements, solution blueprint and design, infrastructure design, security and test concepts are elaborated. Additionally, disaster recovery, availability, scalability and monitoring concepts are also elaborated to the required detail.

Following subsystems comprise the architectural solution: secure entry servers, load balancers, WebLogic cluster, WebLogic portal, portlets and business services subsystems. Legacy subsystems include the Host backend, output management subsystem as well as archiving and monitoring subsystems. To complete the scenery, Identity and Access Management subsystem is extended to handle extravagant external company requirements such as definition and management of team hierarchies with client-to-team assignments and 2 user-to-team relationships. LDAP and Active Directory subsystems are reused.

Special attention is given to security aspects such as single sign-on, DMZ, protocol isolation, firewalls, denial of service, penetration tests and vulnerability detection based on automatic scanning and testing tools.

Technologies: proprietary software development process, UML, WebLogic Application Server and cluster, ALUI Portal, MSSharepoint, Siemens DirX, IBM Rational AppScan, IBM OnDemand, Oracle, Eclipse, Java, XML.

### *Client Reporting Solution*

A large Swiss bank requires a reporting solution to be built to complement its front end suite of applications.

The development life cycle starts with requirements engineering, technical specifications, architectural definition and solution design. Large reporting volumes e.g. at the end of the month processing, are recognised early enough in the project and formally recorded in the non-functional requirements document. Elaboration of a multi-threaded working prototype follows and proof of concept for the proposed architectural solution is obtained with guaranteed throughput. Software construction is carried out with a team of 3-5 developers and 2 testers over a period of 6-9 months.

Technologies: proprietary software development process, asset/portfolio management system Odyssey TripleA, output management system, archiving system, monitoring system, UML, WebLogic, WebSphere, WebSphere Message Queue, IBM OnDemand, Oracle, MSAccess, Eclipse, Java, XML.

### *Test Automation*

A large and renowned insurance company decides to publish its 500+ products on the internet in order to enable its agents to quickly access and accurately service their clients.

In the past, testing of the 500+ products has been done manually and was plagued with massive effort estimates, scarce resources and bottlenecks. It was clear to the test department that going on the web required substantial practice changes.

To the company architects and managers faced with this growing problem, test automation intuitively resonated as a viable solution, though no one on the team had direct experience with this field of practice. Before launching into a full-scale

change initiative, they decide to expand their understanding of basic automation principles with assessment, training and consulting by an external company specializing in test automation best practices i.e. ProSoftwarica.

During assessment, need is recognised for meta data that would drive both the development and test activities. The meta data available was obsolete, so the team decides to consolidate and extend the definition of the meta data with test data.

A testing tool is extended to handle the meta and test data and used to drive the tests in a fully automatic mode.

Within 5 weeks, positive results motivate the team to go further and provide a completely automated testing solution.

The fact is, that after 5 months of automatic testing, defects have been uncovered that existed within the subject-under-test applications, for years.

Technologies: RUP, UML, Java, XML, Rational Functional Tester.

### *Reservation, Visitor Management, Operations and Workplace Automation System*

Workplace workflows such as meeting/conference room reservation, visitor management, conference operation control and workplace sharing are fully automated. The architecture is based on a product line consisting of core services integrating the 4 above mentioned products. Since each product is based on the same core, clients upgrade their systems simply by purchasing additional products. All products are made highly configurable using properties and business rules. Each client may define and maintain their own business rules e.g. multi-tenant capability and deploy them instantly. This capability promotes client independence by reducing the possibility of vendor lock-in.

The realised system interacts with existing client/server applications e.g. Outlook/Exchange or Lotus Notes Domino environments without invasive modification of these systems i.e. data synchronisation is done periodically using server APIs. This eases installation, maintenance and training. Furthermore, Building Automation Systems, Access Control Systems, ERPs, Billing, Back-Office and other end user systems are integrated.

Additional features are: 3000-5000 users, 300 simultaneous users, horizontal scalability within a server cluster, load balancing, fail-safe, single sign on, software as a service i.e. hosted version, web services and remote installation, test and support.

Responsibilities: CTO, Chief Architect, Team lead of 8 software developers.

Technologies: RUP, UML2, Windows, Java, Jboss, MySQL, XML, Clustering, Business Rules.

#### *Innovative algorithms for the foreign exchange market*

Highly computerized, programmatic and algorithmic trading approaches in the foreign exchange domain have become a dominant market force.

The client is an expert in the foreign exchange market and is interested in developing a set of innovative prediction algorithms. Models have been elaborated for the proposed algorithms based on client's proprietary know-how. Data modelling of persistent entities satisfies often changing and creeping requirements typically found in exploratory and prototyping applications. Complexity has been contained using flexible structures for hosting of extremely large data volumes. Calculation intensive transformation operations are readily applied, extended and modified according to constantly changing requirements.

Technologies: RUP, UML2, Windows, C#, .Net, MetaStock, Metalib.

#### *Multi-server capability for a host-based asset management system*

A large multi-user asset management system used in Banking, Fonds, Insurance and Investor markets comprises a number of tightly coupled software applications deployed on a single multi-processor host machine. The vertical scalability of this single host machine has been exhausted to the point where it cannot be scaled any more i.e. by replacing existing multi-processors with more powerful units or by permanently adding more memory resources.

The proposed architectural solution enables the system to scale horizontally using a high performance networked cluster consisting of a dozen of medium sized server machines.

Implemented extensions to the existing system include: efficient load balancing, fault tolerance e.g. reliable fail-over scenarios and real-time cache synchronisation mechanisms. Cluster communication is based on high performance computing technology.

The resulting system is able to accommodate more users and offer an enhanced quality of service.

Technologies: RUP, UML2, Windows, Linux, Solaris, Java and MPI Parallel Computing Technology.

#### *Preparation and evaluation of a tender for a new generation of Self Service Point of Sale - Billettautomaten*

A large Swiss railway transport company prepares to develop a new generation of "Billettautomaten" in cooperation with one of 5 competing hardware and software suppliers. The supplier of choice is to be evaluated based on the carefully elicited set of requirements and evaluation criteria. The task is to identify and formulate a tender consisting of the necessary requirements at a specific level of abstraction which are to be implemented by the hardware as well

as software supplier of choice. A hardware abstraction layer is identified with a number of interfaces satisfying the necessary requirements. After legal approval, written documents are sent out to the competing suppliers. The received feedback is analysed, questions answered and guidance provided until all details concerning the hardware abstraction layer have been clarified. The final evaluation of returned documentation identified clearly the preferred suppliers for the project.

Technologies: RUP, Requirements Engineering, UML2, Windows and Linux (Debian).

#### *A new generation of security control systems*

A Swiss market leader has decided to make the step towards developing a new generation of security control systems. Previous development was carried out using a waterfall life cycle, procedural programming languages and was dominated by scarce documentation and the single head of knowledge syndrome. A paradigm change is necessary in order to reap the benefits of innovative technologies. A software development process is introduced based on the iterative, incremental, use case driven and architecture-centric principles. Training courses in OOA/D, UML and modelling Tools guide the developers into new ways of thinking. Requirements capturing results in over 200 identified use cases. Architectural analysis identifies a framework for a product line i.e. common infrastructure for all future systems. The elaboration phase is used to implement critical parts of the required framework. At the end of the elaboration phase a cluster of 5 to 12 CPUs was deployed and necessary metrics were collected. The results provide a basis for a proof of concept of the elaborated architecture.

Delivered benefit to the client:

The elaborated framework together with quality training and on-the-job-

coaching enables the client team to autonomously face the construction phase. Identified risks related to the paradigm change and the project itself have been either eliminated or effectively mitigated. The result is that the client team is enabled to the stage where domain know-how and diligence are required to lead the project into a success.

We deliver the prerequisites to make the project a success with full functionality, on time and within budget.

Technologies: RUP, Requirements Engineering, RequisitePro, UML2, Rational Rose Real-Time, C++, Windows, Eclipse and embedded Linux.

#### *An automatic, black box test system*

A well-known medical device manufacturer needs to shorten the automatic black box testing cycle and increase test coverage in spite of a growing number of interacting devices comprising the system under test. An evaluation of pervasive, commercial testing tools reveals that not one of them satisfies the set requirements. Therefore, a custom 5-layer architectural solution is defined to bridge the elicited market gap. The automatic tests are script driven, whereby each command is executed in 2 phases. In the first phase, one device is navigated to specific behaviour e.g. to generate configuration updates, that result in ripple effects propagating to several other devices. In the second phase, each participating device is navigated to and verified for compliance against a test specification. Test data is acquired directly from cameras observing LCD displays, tactile and other sensors. Since the system availability is specified to 5\*24 hours, the system enables script testing and debugging in the absence of target devices as well.

Technologies: RUP, UML, Rational Rose Real-Time, Java, Windows.

Environment: RUP, UML, Rational Rose Real-Time, Connexis, XML, Spirit Parser, VxWorks, Interrupt Handlers, Profibus, Windows.

### *Project and Architecture Coaching*

The client is in a position to expand the production of his electronic card payment devices and enter new markets that were previously unimaginable. In order to achieve this target 2 software packages are to be developed i.e. the existing software needs to be made EMV compliant before EP2 functionality can be implemented. Both functional blocks pose a development challenge for the client's development team, 12 developers, which is heavily burdened by maintenance and product release activities. Main activities are project definition, RUP coaching, architectural coaching during framework development, team coaching in OO analysis and design as well as conducting reviews.

Technologies: RUP, UML, C++, Rhapsody, Black box and Unit Test Environment

### *Train yard design and control system*

The train yard design and control system is realised as part of the new railway automation project. The task is to provide a genuine test platform for automation control, whereby each part of hardware is simulated in its entirety. A simulation programming language is provided that enables each simulated component to exhibit behaviour as defined by the current command set e.g. generate faulty behaviour on predefined events. The simulator is a reusable system component relying on configuration e.g. infrastructural topology description rather than hard-coding. Quality attributes of the system are: usability, flexibility, reuse, remote, distributed, asynchronous and real-time. The project was staffed with 5 developers for a period over 12 months.

### *Real-time Java evaluation*

The task is to evaluate a new target platform that will provide better isolation from the underlying operating system, as well as satisfy real-time and performance constraints. The evaluation platform is Perc, NewMonics, a clean room Java VM implementation. Two test applications were written, a client and a server, communicating over TCP/IP that could be executed collocated or remote. Performance dependencies were analysed between interpreted, JIT, accelerated bytecode and AOT code. Control of the size of generated image was enabled by specifying methods to be AOT compiled using the Accelerator tool. The Romizer was used to "prune and shake" i.e. free the image from unused methods. Alternative performance tests concentrated on exploring the run-time features of the Perc VM. Variations in dynamic class loading and timing were tested. Finally, programmatic control of the garbage collector allowed fine-tuning of system performance where and when it was actually needed. The project was complemented with a workshop that enabled know-how transfer to the client.

Technologies: VxWorks, NewMonics Perc VM, PowerPC based system.

### *Automation vision*

A new generation of automation systems is to be developed. A vision document is elaborated exposing a set of emerging user needs by recognizing the opportunity of integrating various subsystems and providing efficient user access to the integral. A number of involved subsystems are identified and the degree of their interaction is

systematically quantified. This knowledge is used to propose an overall system architecture that is to serve as reference for the integration and development of next generation systems with a projected life span of 10 to 15 years.  
Technologies: RUP.

#### *Virtual reception system*

A prototype system is built with the task of automating the reception desk business process with a respective workflow. The use case model provides for the necessary understanding of the domain concepts. Consequently, a clear-cut architecture emerges, identifying contained, well-manageable components. Contracts are specified that the components have to honour, in order for the system to fulfil the set requirements. The chosen software infrastructure, on which the application is built, consists of JSP pages, a web server and a Model2 MVC framework. A user interface running within the familiar and widely available browser is provided. The workflow requires integration of external subsystems, namely badge dispenser, ISDN telephony and access control within the premises. Main activities include project management, use case modelling, architecture definition, as well as team support during design, implementation, testing and deployment.

Technologies: RUP, UML, Together, Forte, Java, JavaScript, JSP pages, Model2 MVC framework Struts, Tomcat web server, C++, JNI, JTAPI, RS232, USB.

#### *UML consulting*

A PBX system is developed based on voice over IP technology. The project introduces new concepts to the team such as RUP, UML and object-orientation. The main tasks are UML

training and coaching as well as architectural consulting.

Technologies: RUP, UML, Rational Rose.

#### *Real-time consulting*

A new generation of real-time control systems based on embeddedNT and VenturCom is designed to be used in three similar domains in the cement industry. Control and supervision are physically distributed via middleware. Control is realised using the VenturCom extension running on dedicated hardware. Supervision clients use the standard Windows NT platform.

Technologies: RUP, UML, embeddedNT, VenturCom, C++, COM.

#### *Elevator system*

Providing consulting to the IT-Department at a technical college. The project consists in offering assistance for the introduction of Java-based, real-time systems with an aim to promoting Java know-how among the students and eventually replacing the existing and out-dating Ada-based environments. For this purpose, a graduate project is carried out with a small team of diploma students. An elevator system is specified, designed and implemented, providing a number of valuable scholarly examples and hands-on experience for the students.

Technologies: RUP, UML, Java, Jbed.

#### *Reengineering of a photo-printing system*

Project lead, architectural definition of the system, coaching and consulting a team of 12 developers. A large software system consisting of +500kLOC is reengineered in two phases. The project assessment phase provides an insight into the metrics and chronic architectural

problems and the rework phase defines and implements the solutions. Proprietary, error-prone communication mechanisms are replaced by Corba-based solutions making use of asynchronous method invocation, naming service as well as the notification service based on the publish-subscribe mechanism. This provides a standard-based system eliminating fragility and promoting maintainability, reliability and robustness. Refactoring techniques provide reusable class trees, enhancing extensibility. Application of real-time techniques enables the machine to reach the specified performance.

Technologies: RUP, UML, embeddedNT, VenturCom, C++, Sniff, tools for software metric analysis.

#### *Framework for a new product line*

Flow meters are devices that must interact with clients as well as provide real-time i.e. deterministic behaviour. The primal forces for the framework are to define the reference architecture and provide for extensibility, maintainability, reliability and performance of the product family. The framework should also reduce the development time for new devices. Enabling techniques include generation of UML classes from a proprietary system description into Rational Rose using Rose scripts and ODBC. Code generation is employed to obtain C++ like code whereby the real implementation language is C. Inheritance, polymorphism and encapsulation are available by implementing the well known OOP concepts in C.

Technologies: RUP, UML, Rational Rose with Extensibility Interface, MSAccess, VC6.0, OO to C Mapping.

#### *Reverse engineering and extension of an ICCP gateway*

ICCP (Inter Control Centre Protocol) is a standardised protocol for the transmission of data in the energy producer/consumer sector. A gateway enables the data exchange between control centres using proprietary protocols. The device itself is a Windows NT-based PC, using a SQL server database to define bilateral communication capabilities. It is required to analyse the existing gateway software, to document and extend it, and provide means for arbitrary client applications to access the available ICCP data.

Technologies: Microsoft Developer Studio 5.0, Microsoft SQL Server 6.0, ICCP.

#### *Fly-by-wire*

Analysis, design and implementation of real-time software for data acquisition and flight control in the aerospace industry. The system architecture is based on a redundant, serial bus (1553b) with more than 20 autonomous, real-time subsystems such as radar, laser-tracker etc. The man-machine interface and process visualisation are distributed on several monitors i.e. 3 head-down displays and 1 head-up display. A flight simulator is developed enabling the superposition of the real world scenery with computer generated images using vector graphics.

Technologies: ADA, C++, VMS, HP-Unix.

#### *Informix database systems*

Analysis, design and implementation of enterprise database systems using 4GL database technology based on the relational DBMS system from Informix running on Novel servers.

### *SCADA systems*

Evaluation, configuration, analysis, design and implementation of real-time process control and visualisation systems for the control of chemical processes. A custom database is the source for flexible control of batch processing. More than 200 field devices i.e. pumps, valves, PID controllers etc. are processed within the 1 second real-time period. The redundancy of processor cards, controller hardware and networking, provides guarantees for a continuous operation in a hard environment.

Technologies: TI-APT Graphcet, DCS-RTOS Yokogawa and Unix.

### *Coordinate measurement machine software*

The automotive industry relies heavily on precise measurements and early production qualification. Work parts are designed using standard CAD tools e.g. Autocad. These models are exported for manufacturing machines as well as CMM (IGES files). After work part production and CMM verification it is possible to quantify and qualify production errors with a qualification report. Dimensional Measurement Interface Standard is also used to ease system interoperability.

A single CM machine exhibits 6 degrees of freedom. Each machine may be fitted with a motorised probe head, augmenting the domain complexity. Clusters of up to 3 CM machines may be organised. Software features include teach-in program mode, program mirroring etc. Computer aided accuracy algorithms provide corrections for 21 machine and temperature errors as well as work part errors. About 40 CMM hardware producers are integrated into the CMM software system including Renishaw,

Zeiss, Stiefelmeyer, Brown&Sharpe, Wenzel, Pantec, Trimesures, Dea, Romer, Imetric, MetroNor.

Hands-on experience includes object oriented analysis and design, implementation and test for the complete coordinate machine measurement suite of applications as described above. Domain of client expertise includes such automaker companies as BMW, VW, Audi, Porsche and Peugeot.

Technologies: Windows NT, C++, Delphi, Modula2.

### *Probe head servo*

Analysis, design and implementation of software for the PHS probe head. Algorithms for calibration, positioning and control of the probe head system allow for integration into existing 3 axis machine controllers.

### *Geometric evaluation using optical systems*

Images of work parts are produced with at least 2 digital cameras. Evaluation of points of interest is done on the Sparc workstations followed by general geometric evaluation using the CMM software.