

**Name**

Mark de Does

**Address**

Magdalenastraat 6  
3512 NH Utrecht  
Holland  
phone ++ 31 30 2314150  
mark@mdedoes.xs4all.nl

**Born**

January third 1956 in Amsterdam

**Education**

1968 to 1974 High School, *Athenaeum B*, Coornhert Lyceum in Haarlem.  
1974 to 1982 Student in Biology with Mathematics at Nijmegen university.  
1982 Master's degree, *doctoraal examen*, in biology. Subjects: Mathematical Biology. Prof.Dr.A. Lindenmayer, Utrecht university; History of Biology. Prof.Dr.P. Smit, Nijmegen university; Theoretical Production Ecology. Prof.Dr.C.T. de Wit, Wageningen university.

**Professional Experience**

Research assistant at the mathematical biology department.  
1983 Study at the Department of Biochemistry and Biophysics of the University of California in San Francisco. I collaborated with Prof.Dr.H.M. Martinez on algorithms for comparing and searching DNA sequences.  
1983 - 1984 Statistical Programmer at the epidemiology section of the Dutch National Institute of Public Health (RIV) in Bilthoven.  
1984 - 2000 Chief Engineer/Product Architect at SuperNova Technology in Bilthoven, Holland.  
2000 - 2002 Senior Software Engineer at w3Creative.com [better known as simcountry.com]  
2002 - 2007 Software Engineer/Specialist with HP OpenView in Amsterdam  
2007 - 2008 Software Architect with Ordina J-Technologies  
2009 - 2015 Software Architect/Consultant with TriOpSys  
2015 - now Lead Java developer with Mylette

Please refer to the separate *experience* text for technical details.

**Publications**

- Does, M. de, & A. Lindenmayer, "Algorithms for the Generation and Drawing of Maps, representing Cell Clones", in Graph Grammars and their Application to Computer Science, Lecture Notes in Computer Science 153, pp 39-57, Springer, Berlin, 1983.
- Waal, F. de, J.W. van der Velden & M. de Does, "De Invloed van het relatieve Lichaamsgewicht op de Prognose van Borstkanker bij Vrouwen", Ned. Tijdschr. Geneeskd. 1985, 129: nr 10, pp 454-458.
- Boer, M.J.M. de, & M. de Does, "The Relationship between Cell Division Pattern and Global Shape of young Fern Gametophytes", The Botanical Gazette, 154(4), pp 423-434, Chicago, Dec 1990.
- Kreis, I.A., M.de Does, J.A. Hoekstra, C. de Lezenne Coulander, P.W.J. Peters, G.H.Wentink, "Effects of Cadmium on Reproduction, An Epizootic Study", Teratology 48, pp 189-196, 1993

**Fields of interest**

Software development, history and philosophy of science, biology, literature, walking, cycling.

**Memberships**

- Association for Computing Machinery (ACM)
- Nederlandse Vereniging voor Theoretische Biologie (Dutch Society for Theoretical Biology).
- NLJUG (Dutch Java User Group)

## Personal details

Mark de Does  
 Magdalenastraat 6  
 3512 NH Utrecht  
 born January 3, 1956  
 phone +31 30 2314150  
 email mark@mdedoes.xs4all.nl

## Profile

Mark is an experienced software architect that always tries to find the simplest possible solution even if it is not a standard one. He can explain technical subjects in simple everyday language and to convince. As an impatient individual, he likes to design and to implement software quickly, but not at the expense of technical maturity.

His long experience helps Mark to easily pick up new subjects and technologies. Mark is a good listener who understands functional requirements and knows how to translate them to a technical solution. First of all, Mark is a goal oriented doer that knows to focus on the essentials without completely ignoring the details. As an enthusiastic craftsman with a hands-on attitude, Mark is a better coach than he is a manager.

His current focus is on Java/JEE and internet technology. He has not forgotten his long experience with databases and in C/C++ development for Unix and Linux however. He even side-tracked into integration, mobile and legacy technology subjects.

Mark is a member of the following professional associations: NLJUG (Dutch Java User Group), ACM (Association for Computing Machinery).

## Potential Roles

Initial design and proof of concept development  
 Solution architect  
 Lead developer/ Technical lead/ Technical coach  
 Technical trouble shooter  
 Software architect, Technical designer

## Overview of technical knowledge and experience

Subject	Level	Years	
Software Product Design	Senior/Expert	25	
Software/System Architecture	Expert	15	
Build Systems and Software packaging	Expert	20	
Maven 2 and 3, Ant, Make, deb, rpm, msi, unix PKG, HP swinstall			Maven: 7, Ant: 6, Make 20
Quality Control and Test Suites	Senior	15	
JUnit, Regression test suites, Scripting VMware VIX, Vagrant, VirtualBox and Docker to run them, Fitnesse			JUnit: 5
GUI Programming	Senior	15	
Angularjs, Swing, AWT, GTK, Motif, NextStep			
Document Formats (and their relative merits)	Senior/Expert	15	
RTF, PDF, PostScript, XML based, ePUB, HTML, markdown			
Linux, Unix	Expert	25+	
Windows	Medior	15	
Java	Expert	15+	
C/C++	Senior/Expert	25+	
Unix cc,gcc,MS-C,from K&R via iso/ANSI 9898-1990 to full C++			
HTML/CSS	Senior	15	
Javascript/jQuery/AngularJS	Senior/Expert	10,	
Scripting	Senior/Expert	20	AngularJS: 3
sh, bash, perl, vbscript			
SQL	Senior/Expert	15	
Relational Databases (general experience)	Expert	20	
Oracle	Senior	15	
MySql	Senior	10	
DB2	Senior	5	
Various other databases	Senior	15	
RDB, Solid, Informix, HP-Neoview, NCR			

Teradata, C-Isam, PussyCat/Berkeley DB, PostgreSQL, MS-Sql, Sybase			
Java IDE's	Senior	10	Eclipse: 8
Eclipse, IntelliJ, RSA, Oracle J-Developer, NetBeans			
Source Code Control Systems	Senior	20,	GIT:4
GIT, CVS, Rational ClearCase, Subversion			
Network Protocols	Senior/Expert	15	HTTP/HTTPS: 10
Mainly HTTP, raw socket programming and some SMTP and WebSocket			
JEE Application Servers	Senior	10	Websphere AS: 6, JBoss: 4, Tomcat: 10
JBoss, WebSphere AS, Tomcat,			
JEE Integration technologies	Senior/Expert	10	
Spring Framework/CDI			
Java Object Relational Mapping	Senior	6	myBatis: 10, Hibernate: 4
IBatis/myBatis, Hibernate, JPA, Spring JDBC-Template, HP ObjectServer			
JEE Front End Technologies and web templating	Senior	7	JSP: 5. Spring MVC: 5. JSF: 5 on an occasional basis.
JSF, Plain JSP, Spring MVC, Freemarker, Velocity			
Web Services/ REST	Senior/Expert	10	
JAX-RS 1 and 2, Spring WS, XML-RPC, Spring REST through Spring MVC, JAX-WS, CXF, Jersey 1 and 2, JAXB			
Java Messaging	Senior	5	WS-MQ: 5, JBoss * 2, Others 2
IBM WebSphere MQ, JBoss-MQ, JBoss Messaging, Active MQ			

## October 2015 - now -- Lead Java Developer with Mylette

### November 2016 - now -- Mylette Tax Reclaimer

Reimplementation of the existing Mylette tax reclaimer application as a REST/AngularJS application using Bootstrap CSS. As the data model is relatively flat, the application can easily be mapped to the single page paradigm and the majority of the transactions can easily be mapped to the REST model. As an extra challenge I used open source components rather than home-brewn directives. The build of the project includes a grunt web build in the regular maven Java build to make sure that **the** build command results in a single (war) deployment unit. The core of the application is a highly configurable PDF form filler.

Technologies: JAX-RS, Angularjs, Apache PDFBox, MyBatis, Spring Framework 4, Apache Shiro, Apache POI, MySQL, npm/grunt/bower/maven

### October 2015 - November 2016 -- Financial Derivative Pricing

Migration of a legacy pricing application of one of the major Dutch Pension Funds to a REST based solution based on the open source OpenGamma platform. The solution consists of an Excel frontend plugin that communicates (via WCF) with a JAX-RS (Jersey) based Java backend that runs in Tomcat. We use OpenGamma as a library, rather than as a stand-alone system. Apart from the Excel plugin, the solution has a batch style module that performs the valuation of complete portfolios.

Technologies: JAX-RS(1), JAXB, OpenGamma, JSON, Maven

## 2009 - October 2015 -- Software Architect with TriOpSys

Software Architect / Consultant with TriOpSys. TriOpSys is a medium sized systems integrator with a focus on mission critical systems and mobile solutions. Besides that, TriOpSys is a specialist in traffic management systems.

### April 2015 - October 2015 -- Software Architect Traffic Management Systems TriOpSys

I was responsible for the architecture and the design of traffic management software projects at TriOpSys. In this short period I was the technical lead and coach of two teams. Both teams did projects for the Dutch infrastructure ministry. (RWS) One team built an experimental user interface based on AngularJS for traffic management on top of a simulator of Dutch inland navigation traffic. The backend implementation was based on GeoServer and the Unity game engine. The other team brought a set of data services into production. We managed to make the software that was messed up by another party about good enough to go into production.

Technologies: Java, (some) Spring framework, JMS, Hibernate, C#, Unity game engine.

### **July 2014 - March 2015 -- Application Engineer (Software architect) KID at RABOBANK**

The *Klantinstellingendienst* is the application at RABOBANK that manages configurable settings that control customer authorizations. It augments the KAD with settings for individual customers that can be configured by the bank or by the customer. Together with the KAD that manages the role of the customer and her authorizations based on business rules, the KID is the heart of the customer authorization logic at RABOBANK. A first version went into production in spring 2015.

The work involved technical consultancy, algorithm and software design and coaching. I built a POC implementation and I also actively participated in the Scrum implementation team.

Technologies: SOAP, Java, Spring Framework 4, WebSphere Application Server, DB2, myBatis, Apache CXF, JAX-RS, JAX-WS, Oracle OSB and SOA Suite

### **July 2012 - Summer 2014 -- Application Engineer (Software architect) DVO-A at RABOBANK**

The *Transparante Vergoedingen* application at RABOBANK is a simple application to collect fees owed by RABOBANK customers. Technically, it was a simple project but the challenge was to finish it fast without cutting too many corners. RABO had the legal obligation to go live in 2012 and we finished just in time. The software is gradually extended to become the standard tool to collect fees for advice.

Technologies: Java, Spring Framework 4, Oracle database, DB2, myBatis, Linux, Connect Direct

### **September 2011 - March 2015 -- Application Engineer (Solution architect) KAD at RABOBANK**

The Customer Authorization Service (KAD) is the system that is responsible for all customer authorizations for self-service applications at RABOBANK. The system is based on the customer and product data in Siebel and the administration of personal bank passes. The expected rapid growth of the use of self service applications over the internet requires high performance and high scalability. For that reason authorizations are precalculated and stored in a single record in the database. For performance and security, we (ab)use DB2 as a key-value-store.

I designed the technical software architecture and the general authorization interface for all RABO internet applications. I actively participated in the implementation team as a senior member and as a coach.

As defining the rules for customer authorizations is a difficult subject with many legal details, we expected the authorization rules to change often during and also after the project. For that reason, it would be almost impossible or at least very error-prone to code the rules in a programming language. We have chosen to implement the business logic in *JBoss Rules* decision tables that are maintained in Microsoft Excel by the analysts. The developers only check the syntax of the decision tables before they are included in the software.

Storing authorizations in a database is a handle for fraud. For that reason we sign the records in the database with a digital signature. (RSA) Authorizations from records that do not match their signature are denied.

The KAD covers the authorizations for 100 000 000 customer arrangements besides those for 11 000 000 customers. Timely recalculation from the Siebel database is a performance challenge, that we just meet with traditional relational database technology. We achieved this through several rounds of performance tuning with the help of extensive monitoring of every single step in the process.

The work involved technical consultancy, algorithm and software design and coaching. I also actively participated in the Scrum implementation team.

Technologies: Java, Spring framework 3, JBoss Rules, JAX-RS (Jersey), WebSphere Application Server, myBatis, DB2, Oracle OSB, SOAP, JAXB, JAX-WS, Zabbix

### **2013 CIV -- Database migration (RABOBANK)**

We Migrated the 130 table/200GB CIV database from HP NeoView to IBM DB2 on Z/OS. The challenges of the project were: (1) Keeping a system with 20 000 users in production while we changed the basis. (2) To recover the original database design and to remove concessions that were made to cope with NeoViews limitations.

A secondary goal of the project was to improve the availability of the CIV application and the underlying database to 7\* 24. We managed to re-implement the legacy data logistics in such a way that reloading a database table does not hurt availability. (Combining DB2 Clone tables with ITWS scheduling.)

Technologies: WebSphere Application Server, iBatis, DB2, IBM ITWS, perl scripting to derive the DB2 schema from a data dictionary in MS-Excel

### **March 2011 - March 2015 -- Application Engineer (Software Architect) CIV at RABOBANK**

The Customer Information View (CIV) is a web application embedded in an IFRAME in the Siebel console. In this way CIV supplements the Siebel CRM system of RABO bank. The application gives customer and account information to all customer facing staff at the local banks and at all call centers.

We gradually upgraded the application from traditional Java web technology (JSF) to a modern HTML5 and JavaScript based application.

The work involves technical consultancy, design, coaching and a little too much trouble-shooting.

Technologies: SOAP, Java, Spring Framework 2.5, WebSphere Application Server, DB2, iBatis, Apache CXF

### **November 2010 - September 2011 -- Software Architect KNMI Climate Information System**

TriOpSys reimplemented the Climate Information System of the Dutch Royal Meteorological Institute. (KNMI) The new system has a uniform architecture and is more robust than the original system. The focus was on future extensibility and on making the data available to researchers, public relations and even to the general public.

The system stores all weather and climate observations from 1850 until now. The data is exposed through a web based system for ad hoc queries. The result of the queries can be downloaded in CSV format or formatted in a wide variety of different formats including publication ready graphs and diagrams.

I participated in the general design of the system and implemented the data layer that is at the basis of the system. Throughout the project I have coached the development team with architectural and technical advice.

Technologies: Oracle database, Spring Framework 3, Spring MVC rest support, JSON, jQuery, Apache Tomcat, JFeeChart, Linux, PostScript/PDF, Spring JDBC template.

### **November 2010 - July 2011 -- Software Architect Prikroute**

TriOpSys built a mobile application for a Health Care laboratory. Nurses take blood samples at the patients home and take the samples to the laboratory. Every day, requests for examinations are collected, planned and then sent to the nurses PDA. The progress of the nurses during the day is monitored and the results are reported at the end of the day.

The application consists of several components: (1) Receiving orders over HL7. (2) Automatically planning the orders based on geographic information. (3) Sharing the information with the PDA. (4) Reporting. Special attention was paid to the reliability and the statelessness of the communication between the PDA and the application server.

Technologies: Spring Framework 3, myBatis, JBoss, Jopt SDK, HAPI HL7 Library, .Net, MS-SQL.

### **January 2009 - November 2010 -- Software architect iMOS**

Software architect for TriOpSys iMOS field service application product. iMOS is a mobile application that supports the field service process. It consists of a set of standard modules for planning. On top of the base functionality, TriOpSys develops and delivers custom modules for its customers. My task was to gradually modernize and optimize the iMOS application architecture without completely overhauling it. A non negligible part of the energy went into the task of convincing the organization that change was necessary if we want the product to survive.

Technologies: JBoss, Hibernate, JBoss MQ, HTTP, CreMe JVM, Windows Mobile, MySQL, Oracle, MS-SQL.

### **May 2007 - December 2008 -- Software Architect with Ordina J-Technologies.**

#### **September 2007 - December 2008 -- Software Architect CIV at RABOBANK**

CIV is customer/product information system that supplements the Siebel CRM system of RABO bank. The application gives customer and account information to all customer facing staff at the local banks and at all call centers. It is embedded in an iFrame in the Siebel user console. Since end 2008, the system is in production with up to 22,000 concurrent users, serving 800,000 screens per day.

The integration with Siebel on one side, and various legacy database structures on the other side were a challenge. The high number of external interfaces and the evolving specifications forced us into a very modular component based architecture based on the Spring framework.

As requirements were not completely clear at the beginning of the project, we have used an incremental approach to development to get commitment in the organization even though this was completely against corporate culture.

Technologies: JSF, Spring Framework 2.5, Oracle, WebSphere Application Server, WebSphere MQ, HP-NeoView

### **June 2007 - August 2007 -- Implementing iDEAL at Europeesche Verzekeringen**

This is a proof of concept implementation that actually went into production.

Technologies: Plain JSP, EJB, WebSphere Application server, XML-RPC, Prototype JS (too much of it)

### **May 2007 -- Legacy system replacement for a Transportation Company**

I participated in the reimplementation of a legacy FOXPRO application with Spring MVC. A major part was fixing the reporting module that was built with BIRT.

Technologies: Spring Framework 2, BIRT, iText.

### **May 2002 - May 2007 -- Senior developer and Team Lead HP OpenView Service Desk**

Senior developer and Team Lead of the Service Desk Team at HP-OpenView in Amsterdam. Service Desk is a Java application that implements the ITIL service management standard. My position was partly in design, partly in implementation. Many of the activities relate to making the product more mature:

- I designed and built an automatic regression test framework. The framework fully automatically installs the packages that come from the latest build, then creates a database and runs the tests.
- I am the designer of the Service Desk installation and upgrade procedure. It is a major challenge to implement and maintain a complex database schema like the Service Desk schema at the customer site.
- I was the architect of the ServiceDesk/Database interface: A home brewn object relational mapping mechanism. Though standard technology might have been preferable, it was quite successful in separating the details of the database schema and the application implementation.

### **March 2000 - May 2002 -- Founding Member W3Creative.com**

#### **March 2000 - May 2002 -- Member of the technical team**

W3Creative.com is the author of the 'Game of the Worlds' internet game that can be found at the URL [www.simcountry.com](http://www.simcountry.com). It is a virtual world consisting of oceans, continents regions and countries. The players are the presidents of the countries. They decide on the budget, education, social policy and economic issues or they just work on trade contracts for their countries. Behind the scenes is a realistic economic world model that runs at the pace of three month in one day on Linux servers. I contributed to the general architecture of the application and the implementation of the model. It was quite a challenge to optimize the economic model process that fully updates a 200000 record database several times a day.

### **1984 - March 2000 -- Chief Engineer with SuperNova Technology**

#### **1984 - March 2000 -- Chief Engineer**

SuperNova Technology started as a software house specialized in UNIX. It now produces the software tools like SuperNova Application Developer, SuperNova Decision Query and SuperNova Visual Concepts. Since I left SuperNova the company has been acquired by WRQ inc. and the tool set was renamed to WRQ VeraStream.

As chief engineer and architect directly reporting to the CEO, I am one of the principal designers of the SuperNova software tools that include:

- SuperNova Application Developer is an object based 4GL and graphical application builder.
- SuperNova Decision Query is a report generator in which the user can build a template report, and let the generator derive the query strategy from the report.

I contributed a major part to the architecture and the implementation of the tool set. (Mostly in C.) My contributions to the architecture and implementation include:

- The abstraction layer between the products and database engines or record managers. It offers simultaneous access to different database systems through a uniform interface.
- The core SuperNova engine. It is an object based engine that executes the business logic in an application system.
- The Query By Report algorithm of SuperNova Decision Query.

### **1983 - 1984 -- Statistician, Programmer with RIVM**

I wrote programs for statistical analysis and data management of the epidemiology department of the National Institute for Public Health in C. An important part of the work was statistical consultancy.

## 1975 - 1983 -- Student in Mathematical Biology

Statistical analysis programs and simulation programs, mostly in a biological context. At university I worked on graph rewriting systems and formal languages. I built a generator for Map L systems, producing graphical output.

## 2000 - 2015 -- Technical achievements outside the context of a job

I am the author of a set of spelling servers originally for the NextStep environment. It is a set of production quality spelling checkers. Its main interest lies in the application of finite automata for approximate search.

I am the author of the *Ted* RTF text formatter. As one of the more complete and fastest open source text formatters *Ted* is widely used to produce appealing PDF output from application software. I have done several projects where I have extended the *Ted* text formatter to support more RTF features and to integrate it with database applications. In particular for two Italian service providers to local banks that use the *Ted* text formatter to produce contracts and policies. I am a specialist in various document formats and their conversion. In particular in RTF, PostScript, PDF and HTML.

I am the author of an easy lightweight wysiwyg rich text processor for Unix/Linux and GTK (or Motif). It is based on the *Ted* text formatter. Though it runs on any POSIX implementation it is a Linux application in practice. It is available from <http://www.nlgg.nl/Ted>. It implements most of the RTF 1.5 standard. Besides that it is an experiment into the limitations of one single individual when he has to implement a full software product.

For pleiade management and consultancy I built a framework for conducting surveys over the internet: [www.pleiadesurvey.nl](http://www.pleiadesurvey.nl). Once installed, it allows a surveyor such as a market researcher to conduct a survey without the help of a technician. For pragmatic reasons the system was implemented in perl. The survey questionnaires are defined as an MS-Excel spreadsheet and uploaded to the application server. The survey is conducted over the internet and the reporting module produces complete fully formatted statistical reports with publication quality tables, statistic tests and bar charts in Microsoft Word format.

I have recently rewritten the front end of Pleiade survey as a modern responsive single page web application using AngularJS 1.x and Bootstrap CSS. The new frontend consists of a WYSIWYG interactive survey builder and a fully responsive single page survey module to take the surveys. The new front end communicates with the old server side through rest-like JSON messages. Evidently, the new front-end is built (through grunt) as a single uglified deployment unit.