Department of Computer Science
University of Petrozavodsk

The Web-SynDic System: Development of the Server and User Interface

Mikhail A. Kryshen

Introduction

Web-SynDic stands for **web**-based demonstration and testing of the **syn**tactic algorithms for solving linear **D**iophantine equations in nonnegative integers.
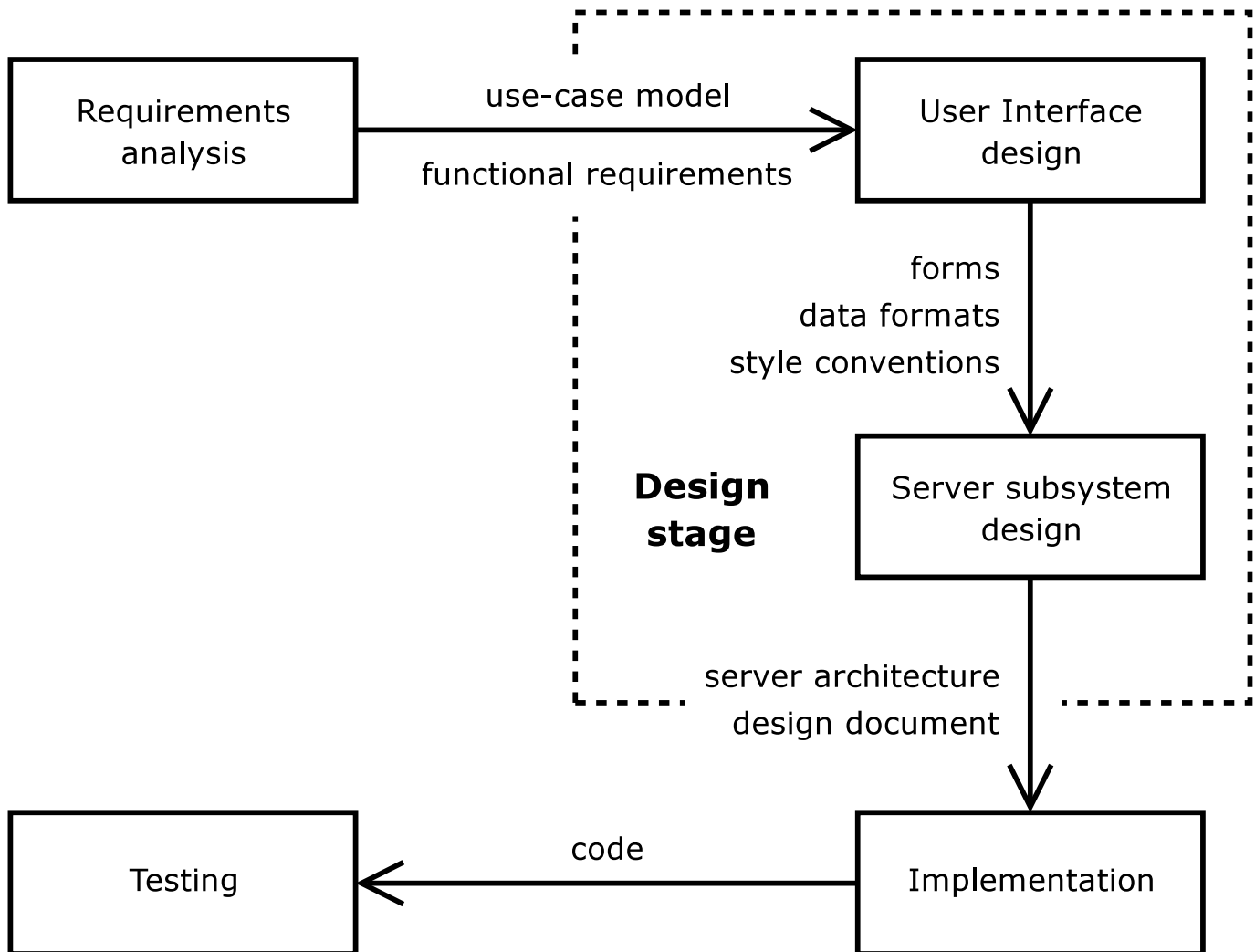
The Web-SynDic project is a student software engineering project of the Petrozavodsk State University (PetrSU), Department of Computer Science. The project is also held in the framework of cooperation between the CS Departments of PetrSU and the University of Helsinki.

The project is related to the research done at CS Department of PetrSU. The research deals with the development of a new type of algorithms for efficient solving some classes of nonnegative linear Diophantine equations (NLDE) by syntactic (parsing) methods.

This poster is about the Server subsystem and User Interface of the Web-SynDic software. The web-based user interface is implemented in the Server subsystem which is responsible for receiving requests from clients (web browsers), forming and sending responses and managing user sessions.
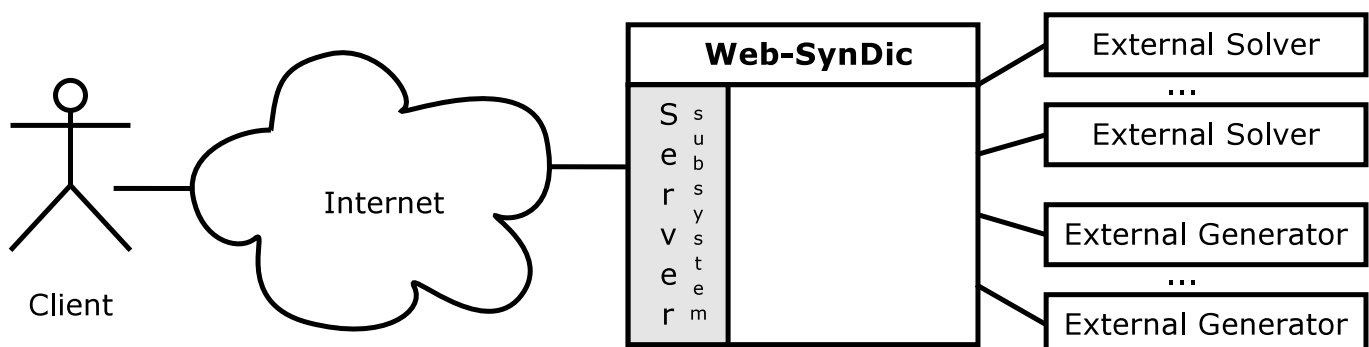
Development Process (Waterfall Model)

```
┌──────────────┐      use-case model        ┌──────────────┐
│ Requirements │ ─────────────────────────> │ User Interface│
│   analysis   │   functional requirements  │    design     │
└──────────────┘                            └──────────────┘
                                                    │
                      Design                   forms │
                      stage               data formats
                                          style conventions
                                                    ↓
                                            ┌──────────────┐
                                            │Server subsystem│
                                            │    design     │
                                            └──────────────┘
                                                    │
                        server architecture         │
                        design document             ↓
┌──────────────┐          code              ┌──────────────┐
│   Testing    │ <───────────────────────── │Implementation│
└──────────────┘                            └──────────────┘
```

Requirement analysis:

    Analyzed the user requirements, produced detailed functional requirements and use-case model.

User Interface design:

    Designed the web-interface forms and data formats (including the input format for the system of ANLDE equations and report on solution format).

Server subsystem design:

    Servlets and Java Server Pages (JSP) technologies were used for the Server subsystem design and implementation.
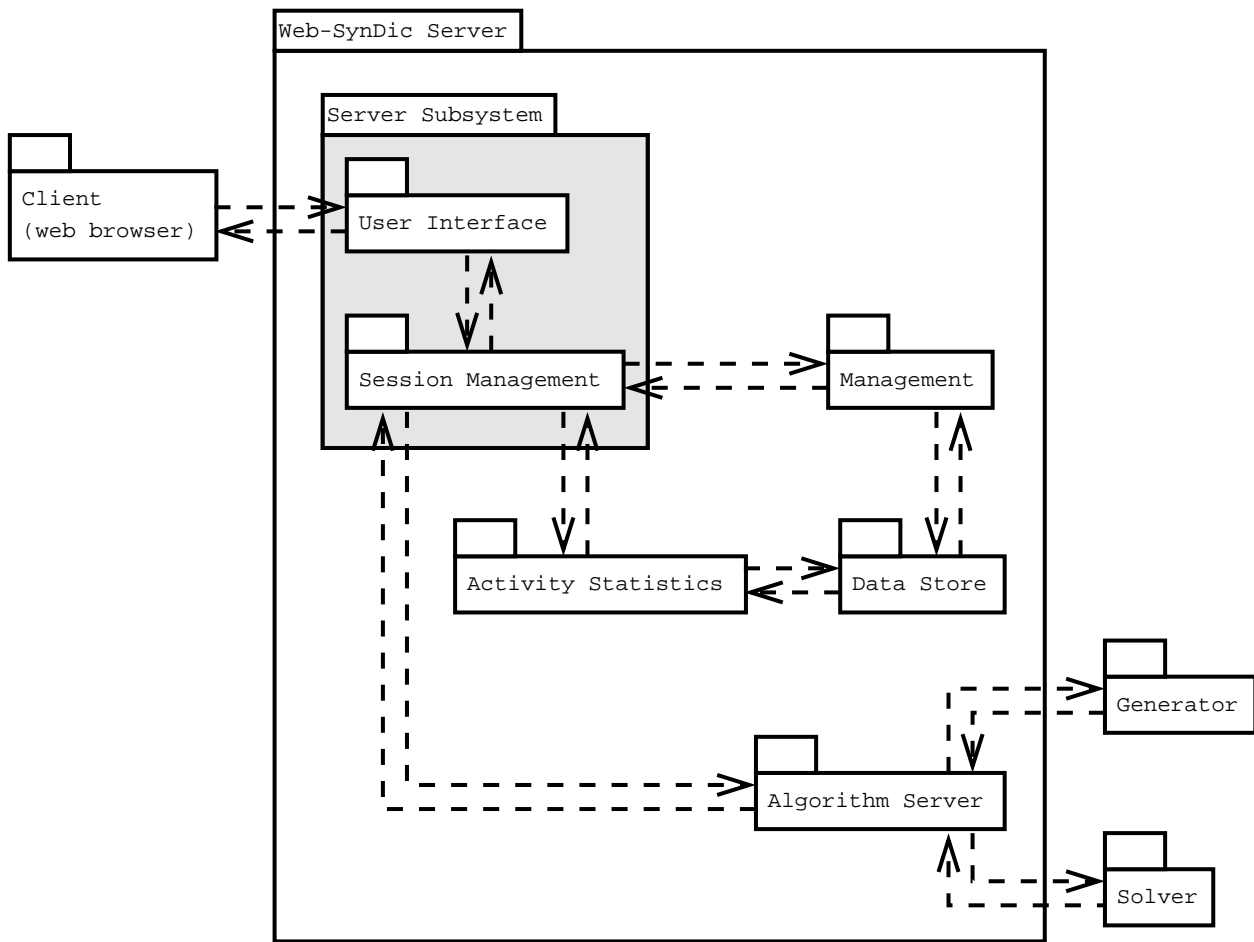
Implementation:

    The Server subsystem was implemented in the Java programming language using the Apache Tomcat package for Servlets class library and JSP compiler.

Testing:

    Some fixes and improvements to the user interface were made. HTML code was checked for compliance with the HTML 4.01 specification.

Web-SynDic Architecture

The diagram above shows the general architecture of the Web-SynDic system.

Server Subsystem Architecture

The Server subsystem is based on the Servlets and JSP technologies. Servlets are Java classes used to receive requests from clients and send responses. JSP is a HTML file with Java code insertions which can be compiled to a Servlet. In the Web-SynDic Servlets are used mostly to process client requests and JSP's are used to generate response pages.
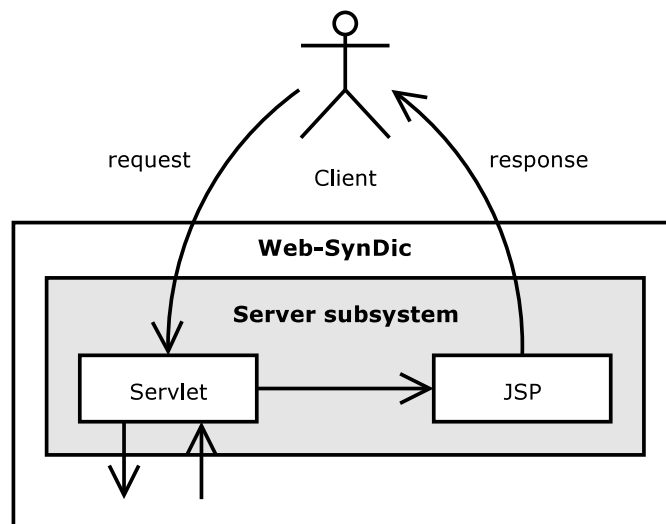
The next UML diagram was produced at the design stage of the project and represents the architecture of the Server subsystem.
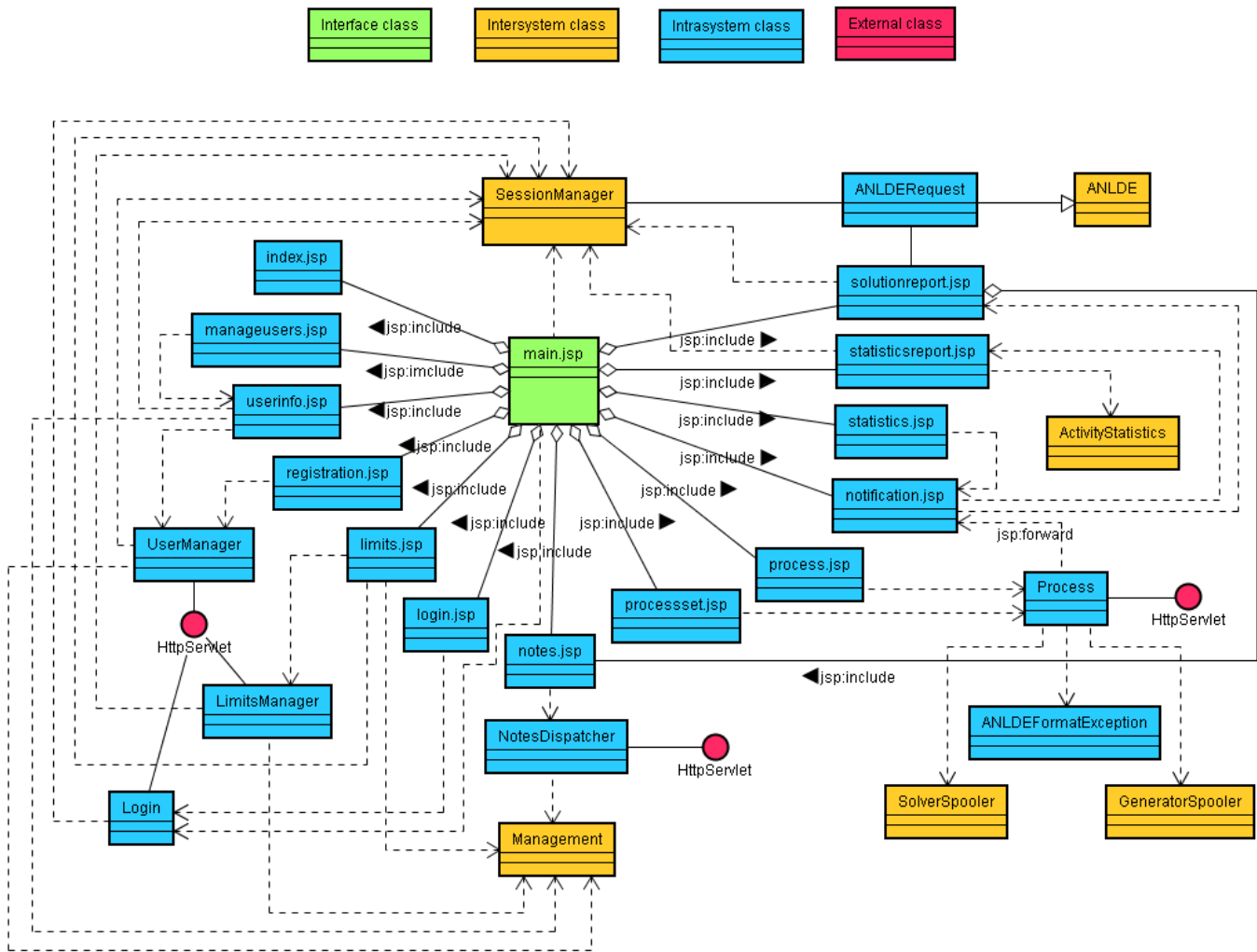
Metrics of the Server Subsystem

The following items were created during the implementation of the Server subsystem:

- **27** JSP files,

- **9** Java source code files (about **1600** lines of code),

- **10** HTML forms,

- **3** report pages.

User Interface Screenshots

The "Process an ANLDE System" form and corresponding solution report.

Web-SynDic: Process an ANLDE System - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost:8080/Web-SynDic/main.jsp?page=process

PETROZAVODSK STATE UNIVERSITY

Department of Computer Science

cs.karelia.ru

WEB-SYNDIC

Process:
- ANLDE System
- Set of ANLDE Systems

Documentation:
- ANLDE Theory
- User Guide

Send Notes

User Limits

Log In

Nickname:
Password:

Log in Register

(you may also continue working as anonymous user, but your profile and limits information will not be saved)

Process an ANLDE System

Input a test ANLDE system (a list of samples):

```
x1 = 29*x1 + 40*x3 + 61*x4 + 56*x10 + 19*x11
x11 + x2 = 82*x1 + 27*x3 + 12*x4 + 13*x10 + 48*x11 + x2
x3 = 42*x1 + 13*x3 + 19*x4 + 58*x10 + 15*x11
x4 = 74*x1 + 35*x3 + 71*x4 + 27*x10 + 59*x11
x10 + x5 + x9 = 32*x1 + 39*x4 + 51*x10 + 63*x11 + 39*x2 + x5 + x9
x6 = 98*x1 + 3*x3 + 19*x4 + 91*x10 + 27*x11 + x6
x7 = 55*x1 + 20*x3 + 66*x4 + 17*x10 + 75*x11 + x7
x8 = 84*x1 + 98*x3 + 2*x4 + 96*x10 + 12*x11 + x8
```

Solve | slopes ▾ | Solve the given ANLDE system with the syntactic algorithm. You may select one alternative solver to compare the solvers.

Generate | gauss ▾ | Generate automatically a new ANLDE system by the selected generator.

Save | Store this ANLDE system in a text format.

Solving and **Generating** are in strict accordance with the current user limits (see the values below). For instance a solver is terminated whenever its run time exceeds *Max. time for solving*, a generated ANLDE system has the same number of equations and unknowns as defined by *Max. number of equations in*

Done

Web-SynDic: Solution Report - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost:8080/Web-SynDic/main.jsp?page=report-solution

PETROZAVODSK STATE UNIVERSITY

Department of Computer Science

cs.karelia.ru

WEB-SYNDIC

Process:
- ANLDE System
- Set of ANLDE Systems

Documentation:
- ANLDE Theory
- User Guide

Send Notes

User Limits

Log In

Nickname:
Password:

Log in Register

(you may also continue working as anonymous user, but your profile and limits information will not be saved)

Solution Report

1. Test ANLDE system:

```
x1 = 29*x1 + 40*x3 + 61*x4 + 56*x10 + 19*x11
x11 + x2 = 82*x1 + 27*x3 + 12*x4 + 13*x10 + 48*x11 + x2
x3 = 42*x1 + 13*x3 + 19*x4 + 58*x10 + 15*x11
x4 = 74*x1 + 35*x3 + 71*x4 + 27*x10 + 59*x11
x10 + x5 + x9 = 32*x1 + 39*x4 + 51*x10 + 63*x11 + 39*x2 + x5 + x9
x6 = 98*x1 + 3*x3 + 19*x4 + 91*x10 + 27*x11 + x6
x7 = 55*x1 + 20*x3 + 66*x4 + 17*x10 + 75*x11 + x7
x8 = 84*x1 + 98*x3 + 2*x4 + 96*x10 + 12*x11 + x8
```

Number of solutions: **5**

2. Performance metrics of the algorithms:

Algorithm name	Summary system time(sec)	Summary work time(sec)	Maximum memory usage(Kb)	Solving result
syntactic	0.010	0.261	2396	solved
slopes	0.360	1.332	2396	solved

3. Solving machine characteristics:
- CPU: AMD Athlon(tm) XP 2000+ (1674.498 MHz)
- RAM: 257084 kB
- Operating system: Linux 2.4.23 i386

Done