## **Resume**

**I am a problem solver**. Just tell me "*This cannot be done*", and you have captured all my interest immediately.  
 **Extensive experience with**:

* C#, LINQ, Rx.NET, .NET, C++, C, OOP
* ASP.NET MVC (Web API), IIS, Puppeteer
* HTTP, REST, JSON
* Javascript, React, Angular, AngularJS, TypeScript, RxJS, KnockOut, jQuery, HandleBars
* XSLT, XPath, XML processing
* Declarative and functional programming: XSLT, Haskell
* Content management, Data ingestion and transformation
* Compiler Construction, Text Processing, Computational Linguistics, NLP
* Fluent in two foreign languages: Russian and Bulgarian

**Novel work on algorithms and data structures**:

* Provided the first C# implementation of the *Finger Tree* data structure.
* Implemented in XPath 3.0 the Binary Search Tree and the Set data structures.

**Creator of Libraries and Tools Used by Thousands of Developers**:

* The FXSL Functional Programming Library for XSLT
* The XPath Visualizer

**Member of the W3C XSLT Working Group (2011 - 2013)** -- worked on the design of XSLT 3.0.

# Companies worked for

* 1. **Microsoft Research** (a contract via Tinine Software): **Dec. 2019 to June 2019**.
     1. [**Microsoft Coyote**](https://microsoft.github.io/coyote/)**:** Created complete code sample and their documentation until Coyote was released: [Robot Navigator Failover](https://microsoft.github.io/coyote/learn/tutorials/failover-robot-navigator-actors), [Hello world (actors)](https://microsoft.github.io/coyote/learn/tutorials/hello-world-actors), [Hello world (tasks)](https://microsoft.github.io/coyote/learn/tutorials/hello-world-tasks)
     2. Electronic parts price data retrieval web app: a SPA with Angular, Node.js, Puppeteer
     3. The UI of an Azure ML privacy system
  2. **Apex Learning**: July 2019 to October 2019  
     Fullstack .NET developer: C#, REST (ASP.NET Web API), SQL Server, Angular, CanJS, NUnit, Jasmine, QUnit. CICD, Jenkins. Worked mainly on the middle layer and the front-end of a Tutorial Authoring System
  3. **Microsoft** (a contract via ArtH System): **Dec. 2017 to June 2019.**Worked on creating new[Azure LogAnalytics Office365 solutions (assessments)](https://docs.microsoft.com/en-us/services-hub/health/assessment-prereq-docs) – [Exchange](https://opdhsblobprod04.blob.core.windows.net/contents/3832eb6e467d45c4aed5538d8c2d8797/3336e9244c27c69e69eb1d3492941302?sv=2015-04-05&sr=b&sig=iwSOc3V5Yea%2Fu8PqB3mzHi5KZD6HfUlx4ld%2BY%2FZzyfQ%3D&st=2019-10-17T01%3A48%3A01Z&se=2019-10-18T01%3A58%3A01Z&sp=r), [Skype](https://opdhsblobprod04.blob.core.windows.net/contents/3832eb6e467d45c4aed5538d8c2d8797/676864b9d3fe1f90bc3f0bfab9ea0c9d?sv=2015-04-05&sr=b&sig=IoiePKxSKpfiHwBzZpv1yFO%2FE0HoJscMkrEywOaSlVU%3D&st=2019-10-17T01%3A48%3A45Z&se=2019-10-18T01%3A58%3A45Z&sp=r) and [SharePoint](https://opdhsblobprod04.blob.core.windows.net/contents/3832eb6e467d45c4aed5538d8c2d8797/f0735cc14be30ce5bb85b416c98c618a?sv=2015-04-05&sr=b&sig=FOfrnM4aBYhNrNYBnVFjIv0Ks2fUwGYnX0dOTsC6o%2Fc%3D&st=2019-10-17T01%3A49%3A20Z&se=2019-10-18T01%3A59%3A20Z&sp=r) Online assessments. Developed almost completely the [Azure Assessment](https://opdhsblobprod04.blob.core.windows.net/contents/3832eb6e467d45c4aed5538d8c2d8797/575b4377c3f4aa5d70583e98c304d5fd?sv=2015-04-05&sr=b&sig=PSt7QuvWHFLIEyktvdGaZZilbcjbk1DwR8JY%2BBR2p3o%3D&st=2019-10-17T01%3A44%3A58Z&se=2019-10-18T01%3A54%3A58Z&sp=r). C#, Powershell, AngulsrJS, TypeScript, Javascript, KnockOut, Jasmine, NUnit, Dependency Injection systems.
  4. Client-side transformation of the collector output
  5. Maintained the code for Data uploading
  6. Developed and maintained Powershell cmdlets for configuring an assessment task, running an assessment and removing an assessment task
  7. Worked on the Azure Portal and Services Hub front-end UI

1. **A Place for Mom:** Sep. 2016 to Oct. 2017. Web Services, ASP.NET MVC Web-API, C#, NInject, MySql and SQL Server, Git, AngularJS + Typescript + Javascript – AngularJS directives, controllers and services, code for call-center in-bound/outbound calls, warm transfers. Unit testing with XUnit. Increased testability of legacy code. Agile sprints, Kanban
2. **Expedia** (a contract via Vega Consulting): Sep 2015 to Aug. 2016. Implemented charging of Agent Service Fees for change/cancellation of Orbitz itineraries. Front end web-development, ASP.NET MVC, AutoFac, SOAP, Java, Eclipse, XUnit, GIT, Stash, XSLT, XML Schema, SQL Server
3. **IMS Health** (Appature), 2013 to 2015. Worked in the Data Processing Group:
   1. Implemented a REST API web service component of a cloud-based (AWS) BI staging and optimization product, using Amazon’s DynamoDB NoSQL database.
   2. Implemented the Role-based User Authorization component of Nexxus Marketing.
   3. Implemented a customer dialect for querying the Nexxus Marketing database.
   4. Guided Nexxus Developers in their use of XSLT for XML processing.
4. **Intentional Software** (Jan – June 2013). Worked on the GTD (“Getting Things Done”) project:
   1. Implementation of MVC (API) pages, data objects, data-transfer objects, view models.
   2. Creating, reading, updating/ deleting MongoDB records using MongoDB (C# LINQ).
   3. Javascript – client side views, dialogs, AngularJS states and templates/partial views, Angular AJAX operations, Angular directives and Angular filters.
5. **Microsoft** **(2004 to 2013)**. Worked in several roles and positions:
   1. Implementation of an MVC-based SDK for producing Metro-style web pages for Windows 8 / IE10. This SDK was used by all teams in MSN for producing the revamped Metro-style MSN pages for Windows 8 and IE10+. MVC3 programming for “**[msnNOW](http://now.msn.com/)**”.
   2. Data ingestion for MSN Extreme Mobile.
   3. The Transformation Server of MSN.
   4. Took part in design and test of the “Data breakpoints” feature of the VS XSLT Debugger.
   5. Worked as a Microsoft Consultant at Australian Government departments.   
      **Technologies**: C#, IIS, ASP.NET-MVC, XSLT for use in content publishing
6. **IAEE** (**the UN International Atomic Energy Agency**), 1999 to 2004: A middle-line software development manager in the IT division of the Safeguards Department

# Programming Languages

1. **C#**. Used daily as one of the main programming languages at my job. In my free time developed [**the first C# implementation**](https://dnovatchev.wordpress.com/2008/07/20/the-swiss-army-knife-of-data-structures-in-c/) of the [**Finger Tree functional Data Structure**](http://www.soi.city.ac.uk/~ross/papers/FingerTree.pdf).
2. **TypeScript**. Used daily as one of the main programming languages at my job.
3. **C++.** At the early stages of my career implemented the ActiveX (scripting) control for the Jasmine OODBMS – while working for Fujitsu Australia.
4. **C** – Implemented spelling checkers (morphological analyzers for Russian and Bulgarian), the Tomita algorithm for NL parsing in a Fulbright post-doctorate work at Carnegie-Mellon
5. **Python**. Used in the last five months when working at IMS Health.
6. **Java**. Occasional coding on the Java Voyager service of Expedia. Some work encapsulating Microsoft’s first XML parser, written in Java.
7. **Haskell**. My FXSL library is an XSLT implementation of a subset of Haskell’s Prelude module.

# Other Disciplines/Tools

**Keen interest, familiarity and hands-on experience with:**

1. **Data Bases:** SQL Server, MySql, Non-SQL databases: Amazon DynamoDB, and MongoDB.
2. **Algorithms and Data Structures** and **Algorithm Complexity**.
3. **Compiler – Writing Systems** (Yacc/LEX, [**the general LR Parser of FXSL**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/func-lrParse.xsl?revision=1.7&view=markup)).
4. **Content Management Systems: Storage, Transformation, Caching and Scalability.**

# Work on implementing functional data structures

1. Developed [**the first C# implementation**](http://dnovatchev.wordpress.com/2008/07/20/the-swiss-army-knife-of-data-structures-in-c/) of the [**Finger Tree functional Data Structure**](http://www.soi.city.ac.uk/~ross/papers/FingerTree.pdf)**.**
2. Developed the [**Binary SearchTree datatype in XPath 3.0**](http://dnovatchev.wordpress.com/2011/05/31/the-binary-search-tree-data-structurehaving-fun-with-xpath-3-0/).
3. Developed the [**Set datatype in XPath 3.0**](http://dnovatchev.wordpress.com/2011/02/20/the-set-datatype-implemented-in-xpath-3-0/) .

# XSLT/XPath/XML pioneering work

1. **Functional Programming in XSLT**:
   1. [**Demonstrated how to use XSLT as a functional programming language**](http://fxsl.sourceforge.net/articles/FuncProg/Functional%20Programming.html).
   2. Developed [**FXSL**](http://fxsl.sourceforge.net/) the functional programming library, both [**for XSLT 1.0**](http://conferences.idealliance.org/extreme/html/2003/Novatchev01/EML2003Novatchev01.html)and [**for XSLT 2.0**](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.107.3596&rep=rep1&type=pdf).
   3. Proposed the “[**function invocation pattern**](http://www.w3.org/Bugs/Public/show_bug.cgi?id=7350)” as correction to the initial W3C WD on Higher-Order Functions in XPath, and parts of this proposal is now in XPath 3.0.
2. **Reputation of being a well-known XSLT/XPath problem solver/expert**. For examples see [**my blog**](http://dnovatchev.spaces.live.com/blog/)**,** browse the archives of [**xsl-list**](http://www.biglist.com/lists/xsl-list/archives/#browse) and [**xml-dev**](http://lists.xml.org/archives/xml-dev/)**,** [**lookup my profile at StackOverflow**](http://stackoverflow.com/users/36305/dimitre-novatchev?tab=stats)
3. **XSLT/XPath tools** developed by me, such as the [**FXSL**](http://fxsl.sourceforge.net/) library and the [**XPath Visualizer**](http://www.huttar.net/dimitre/XPV/TopXML-XPV.html)
4. **Participated with papers at the Balisage (Extreme XML) Conferences (2003, 2006, and 2013).**
5. **Participated as a member of the W3C XSLT WG (2011-2013), where I represented Microsoft**.
6. **Solved myriads of problems commonly considered “impossible to do with XSLT:**
   * + - [**Recursion with an anonymous function**](http://dnovatchev.wordpress.com/2012/10/15/recursion-with-anonymous-inline-functions-in-xpath-3-0-2/).
   * [**A general LR-parser**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/func-lrParse.xsl?revision=1.7&view=markup) **in XSLT 2.0**.
   * [**Convert a JSON instance to an XML document**](http://dnovatchev.spaces.live.com/blog/cns!44B0A32C2CCF7488!367.entry) with an XSLT function.
   * [**Binary Search Tree data structure implemented in XPath 3.0**](http://dnovatchev.wordpress.com/2012/01/09/the-binary-search-tree-data-structurehaving-fun-with-xpath-3-0/)
   * [**The Set Data Type implemented in XPath 3.0**](http://dnovatchev.wordpress.com/2012/01/12/the-set-datatype-implemented-in-xpath-3-0/)
   * [**Sudoku solver**](http://ajwelch.blogspot.com/2006/03/dimitre-novatchevs-sudoku-solution.html).
   * [**Spelling checker**](http://www.biglist.com/lists/xsl-list/archives/200504/msg00178.html).
   * [**Concordance builder**](http://www.stylusstudio.com/xsllist/200511/post00190.html).
   * Numerical functions implemented in XSLT:
     + [**Trigonometric functions**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/func-trignm.xsl?revision=1.3&view=markup), [**Exponential/logarithmic functions**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/func-exp.xsl?revision=1.4&view=markup).
     + [**Random sequence number generator**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/randomList.xsl?revision=1.2&view=markup).
     + [**Numerical differentiation**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/easyDiffList.xsl?revision=1.1&view=markup) and [**Numerical integration**](http://fxsl.cvs.sourceforge.net/viewvc/fxsl/fxsl-xslt2/f/improvedIntegration.xsl?revision=1.1&view=markup).

# Author of Training Courses (Pluralsight) on XSLT/XPath 3.0/2.0/1.0

1. [**XSLT 2.0 and 1.0 Foundations**](http://www.pluralsight.com/courses/xslt-foundations-part1), 8:45 hours duration, Pluralsight, 2014
2. [**The Evolution of XPath: What’s New in XPath 3.0**](http://www.pluralsight.com/courses/xpath-3-0-whats-new), 4:29 hours duration, Pluralsight, 2014
3. [**What is New in XSLT 3.0 – Part 1**](http://www.pluralsight.com/courses/xslt-3-0-whats-new-part1), 5:28 hours duration, Pluralsight, 2015

# Education

1. **Ms. Sc**. In Computer science from Sofia University.
2. **PhD** in Math/Computer Science from Sofia University.
3. **Post-Doctoral** work at Carnegie-Mellon University.
4. **Pluralsight Training Courses:** exam certifications for 45+ courses – see [my LinkedIn profile](https://www.linkedin.com/in/dimitrenovatchev/).