

Svilen Dobrev *making software - for people, by people*

g'day.

i've been making *software* for 35+ years, on several continents, as projects of all sizes and levels of impossibility. Seen many *different* terminals, languages, designs, projects, customers, teams, organizations and cultures, and have met even more different *attitudes*.

So i found that the more interesting and difficult part is.. the *people*, while the technicalities are somehow solvable. Software is just a limited, twisted way of *communicating knowledge* between people, through space and time - and it's people who cause anything. Making software is a game of *cooperation*, now vs future-and-past, exercise in *correct thinking*, with all the participants being *translators* - and *Trust* being the most important feature.

And i found i can successfully *mediate* between the extremes - western and eastern thinking, technical & applied, experts & newbies, gears & art, real & virtual.. i put *passion* in my work and *do care* for it. i do trust people.. and thus weigh them. i work With, not For. i fight For, not Against. i do Work, not Time. Some toys go at github.com/svilendobrev.

Reality.. is very complex. Complexity is tamed with structure. Structure means *languages* - specific for the notion, domain, product, project, task.. even people. i like making languages that talk patterns of problem at hand.. expressed in host language. And i push for *attitudes* that empower people. So one can go solving the real issues. i love idea-generating, *nice* software, mentoring, team and organization making, steep changes, persistence, cultural diversity, improvisation. i can help *efficiently* make *effective* software, and tune the *culture* to allow for it.

i make (software) projects from ideas, people and software. Be it possible or not, regardless what has to *change* - software, organization, people, or.. *me*. As life is change is life.

So.. *Hands-on* software mentor, advisor, "playing-trainer", leader, architect, CTO, hacker.. programmer.. Friend? Could be. More about me, software and life at www.svilendobrev.com.

An Impossible dream project? What about?



az@svilendobrev.com

Personals

- now in *Varna, Bulgaria*. Bulgarian and Australian citizenship
- fluent Bulgarian, Russian, English
- born 1970 Bulgarian, family+kids; non-smoker ; mountains, bicycle, e-foil and sea lover

Specialties and interests

- the *Team* as live, self-tuning instrument for *translating knowledge* into software; *organisational* patterns, cultures and health ; leadership and empowerment; proximity + communication patterns, de-hierarchisation
- *mentoring, teaching*, inside-trenches organisational analysis and coaching... and learning ; philosophy of software
- *knowledge* and technology management, *requirements* engineering, audit + analysis - of code, workflows or organisation
- methodologies, *agile* CrystalClear vs CMMi/formal process; outsourcing + distributed global development; mental gears, *cultural* differences, context-switching, *attention* to detail, selective stupidity
- *making languages*; using general languages as carriers of much higher-level *specific* ones; *generating* code; *self-interpreting* objects; grammars, compilers, interpreters, virtual machines; interfaces, protocols, frameworks and aspects
- *usability* engineering, use-cases; user experience, interaction, interfaces; perspectives - tech vs human, end-user vs organization; human-machine systems
- *testing* as art; version, configuration and project management, life-cycles, estimation, negotiation, risk and resistance-points; documentation workflows, traceability, trackability; licensing
- *business analysis* in many domains – software itself, media, content/metadata, finance/accounting/banking, HR, chips-design, bureaucracy, betting, CAD/CAM, mechanics, physics, hardware, maps, graphics, audio, pre-print, simulation
- *modeling, architecture, flows* - data, code, events, *time*; analysis; optimisation; efficient-effective; persistency, databases, bitemporal data+code, durable execution, messaging, semantic relations; sync/async; client-server/ standalone/ distributed
- languages: *python* 25y, C/C++ 25y, javascript 7y, java 3y, graphql, html/css 25y; perl make sh; many many more.. from prolog to erlang/clojure to various assemblers and machine languages
- toolkits/frameworks: django flask fastapi react nodejs express ; qt wx gtk kiwi; sqlalchemy, STL; many more
- platforms: linux, android, web, desktop, dos; apache; many compilers, assemblers, .. ; docker
- paradigms: object-oriented, functional, event-, rule-, dependency-driven, markup, behavioral, ..
- persistency: xtdb (bitemporal), couchdb, mongo, sql - sqlite postgresql ; rdf, sparql, key-value; corba

- tools of the trade: vim, linux, vcs: bzt/git/hg/svn/cvs; e2e testing, selenium; Jira, redmine, wikis in general
- my *own* python course (~90+ people in last 10 years), a coding style, methodology, UIs, python clone, tools..
- www as runtime env - service vs library; API-design; search, accessibility, multi-linguality; digital signatures, smartcards
- devOops.. or more. Get job *done*, be part of it - software, hardware, or psychology; fix the dripping tap, broken machine, drivers or paperwork; teach, hack, negotiate, *find* a way, *fight* for it.. It's all one jazz.. *Engineering*
- *reverse engineering*; search, anonymity; protections; social nets, digital identity
- reflecting, exchanging ideas, having opinions, Do-It-Yourself, repairing.. things
- artistic design and layout; applying knowledge across fields; making toys and playing
- *caring* about what i do. With extreme attention to detail – which assumes freedom + no noise... and nothing fake..

Motto-s and thoughts

- Find a *friend* to be your senses.
- With time, places *change* people. Other way happens noticeably only while *coming in* - or *switching on*.
- Software is a (twisted) way of *communication*. Sometimes to yourself from Tuesday.
- *Languages* are your tools. Make them if no suitable ones.
- One can't make decent *tool*/thing if has never been *user* of it.
- Do not try to outsmart the user.. who might be your self.
- If you make something usable by idiots, only idiots will use it.
- Association is a great thing - trust your common intuitional sense.
- Trust is the *essence* of software. Machines *trust* 100%, while people.. don't. How much trust or distrust can you handle?
- Software is about *people*. Okay, mostly about (some) people somewhere in the chain.
- Software is made of decisions (and so is life, but it's less obvious, many decisions are materialized and don't look as such).
- www is *global* village, anyone at *near-zero* distance.. but never zero. And one may need that - warm touch - or good kick.
- In making software, doable - sellable - wanted - needed, rarely overlap. Finding the dream spot = ruining illusions.
- Don't always believe yourself. Always believe in yourself.
- There's no "i cannot" - there's only "i don't want". And, there's no "have to" - there is only "want to".
- Search for (people-who-bring-) questions. They make life interesting. Not.. answers - whole industries and institutions exist just to produce such.
- First rule of Leadership: it's all *your* fault.
- Manager's choice: change *People* or.. *Change* people.

Experience

CTO / Chief Architect - eCollect.org, remote/Varna, 2021-2024

- receivables scaleup, FinTech. Team of 8-12, remote, in-house node.js+mongo+vue.js event-sourcing micro-services. "No coding, just.. other 90%". Develop team, educate management, other depts, find devel and software bottlenecks, buy time+resources to fix. Streamline processes, reviews, CI/CD, UX studies. Be product owner - stop everybody-tells-tech-do-this-now. Represent software to clients - and investors. Even architect, reverse-engineer, code some! Then.. acquired. Hope? Lead a remake for huge volumes - durable execution, bitemporal data.. But no.. too much bureaucracy, even more politics.
- no-coding the in-house js system, but quite some coding outside it; node..+html side projects, and python for the remake.
- Lessons - "Remote" does not help teaming.. And.. "Chief" may mean nothing.

CTO, mentor, team lead, architect, main coder - Resolve, Varna.bg + FleetPal, NJ.us, 2017-2021

- small startup with friends. About dutch dental-insurance B2B workflows, then american truck-maintenance housekeeping. Doing most hats.. machinery, product road-maps, frontend and UX, methodology, workflows, team of 3-7. Max efficiency-per-teammate.. python, django, graphql-instead-of-drf, React+Blueprint single page app via browser-editor generative framework. Mentoring 2 younger colleagues into better persons. (~ 90k python + 90k javascript and up)

software expert and methodologist, leader of software group .bg - ZMDI / IDT Varna, 2014-2017

- branch of german fab-less chipmaker. Caring for software teams in Varna+Sofia (8 people), as capabilities and "health"; represent software perspective in complex projects; mentor and teach people from various backgrounds; push improvements in culture, methodology, technology; hands-on - deeply embedded or application software, assemblers to python ; tools for requirements engineering, chip-description and chip-testing ; functional safety.

mentor, coach, analyzer – python, teamwork, organisation – Strypes.eu, Sofia, 2014/contract

- Dutch nearshoring company. Python courses for 20+ people - beginners to advanced.. Noticed fast-growth pains.. so.. did a study, interviewing in trenches and above - on ways of work, culture, technology, relations, teaming, pains, funs.., and suggested some changes towards healthier company structure, agile patterns, attitudes, culture.

python team mentor – Ericsson software campus, Athlone, Ireland, 2013/contract

- dozen new people, most straight from the uni... to become programmers.. and a real software Team.. in about 4 months. Impossible? Depends.. As long as proper mentality gets in, the rest is.. up to them. (update 2019: they are all seniors now)

architect-maker - distributed personal places+chat platform - messaging via couchdb. 2012-2013 Varna-Tokyo

- Store data as-is, let it flow, react on changes. Focus on user, personal+usability. Centralizing is tool, not goal. 2ppl, 8mo. Friend did iphone, me- all else. Design, "http-rpc-with-memory"/replicate, back-end, android+web apps (17k py+12k java).

...dreaming - a software house-school, making real projects and learning by doing - and other way around. 2012 Varna

- Idea: Making software is actually *learning*.. Didn't work, a chicken+egg - students want \$\$\$, companies want ready teams.

senior pythonista, someones.com, Melbourne, Australia, 2012 (short contract)

- social-media python-backend-integration framework (flask, nose, selenium). in place/team that opposes change.

hands-on CTO/ Dev-Lead/ architect/ all-thats-needed, Woosha.it; 2010-2012 - risk-it-all startup, Amsterdam+Varna

- redefine *search* and *decision* in human terms.. Tech lead+ philosophy, analyze, invent, code; mentor young entrepreneur
- moodyTV: *content-discovery* of many metadata sources; shown IBC2010-Amsterdam. Apps: android thick on/off-line, web, qt. Personal channels/ suggestions, manual faceted shareable filters. (3ppl, 6mo, 8k python, 20k java); brightBird - movie search *social* service. Match/classify metadata, user patterns. (3ppl, 5mo; 10k python/django, sql, sparql)
- notionery: wild idea- a garden to grow *your personal notions* - remember, learn, express yourself, in your own terms - and translate to and from all else, incl. past versions. Few months of concept digging.. Semantical web5.. Too far fetched

mercenary senior dev at Wyplay.com, Marseille, France; 2009 (short contract)

- UI-engine in python, embedded, linux/gentoo, slow media-player device; very entangled work / repository structure.. A bit extreme independence - Viva la liberte! - yeah, but without communication, software ain't edible..

owner, Svilen Dobrev Consulting; 2008-... - advisor, methodologist, architect, programmer

- [my site](#) (static html php css javascript python/generation, shared apache, mod_rewrite)
- desktop client app for customs.bg B2B document-flow. (7k python, 4mo, soap, XML-signing, smartcards, gtk/win)

linuxteam leader, SIStechnology.com, Varna; 2003-2008 – team-lead, architect, methodologist, manager, mentor, programmer

- HOR - enterprise human resources+payroll, 90k python, sql, wxpython; 8ppl 2years; replacing legacy, represent *ever-changing*, entangled legislation+relations, max flexibility and expressive power. Notions like pay-flows+values; languages for them, multilingual bi-temporal data *and* code. *Leading*.. agile, architecture, code, requirements, mentor, manage, invent..
- STC- Singapore Turf Club's betting terminals+kiosks renovation – *global outsourcing* effort; 2 end-user-apps: 130k C++, 15k python, 2k java, 6ppl, 2+years; about 6 months onsite in Singapore, 2 in Germany. CORBA, realtime payments. Being *team* leader+manager, core dev, architect, all relations - to WN-Konstanz, WN-Singapore, STC: Difficult cultural mix.. like Requirements gathering via 3 hops who don't talk to each-other. Future-proof, refactor, wrap, generate code+docs.. Most difficult but rewarding project.. Hey, my team survived crossing 4 cultures, and "end-users said they like your software"!

senior software engineer, Vintech, Varna, Bulgaria; 2002-2003

- topological parametric modeling language + control of mechanical technology - in CAM/CAE plant system; python
- graphical 2D backend API from python into C++; fix + wrap equation-solver [cassowarypy](#)

senior software analyst, Petrosys, Adelaide, Australia; 2000-2002 - small, fast company (geophysical maps)

- a drawing framework - 2D hierarchy, MVC, polygon-processing, C++,python; visualise GIS map-data from arcSDE
- organisational patterns - success in changing company culture - both as technology and people communication

tech lead/ test lead, Motorola, Adelaide, Australia; 1999-2000 - EDA - multi-cultural env in huge global corp...

- new compiler for analog/mixed-signal Verilog-A/AMS, merge and maintain legacy compilers; testing and build-automation; C/C++, sh,perl,make, lex,yacc,pccts; solaris hp linux; full waterfall, requirements, autogenerate documentation
- tasting another language-culture-continent-society
- learning about software made industrially - "pipelines"+ CMM4, and teaching about software made efficiently

software engineer/consultant, Vintech, Varna, Bulgaria; 1997-1998 - small company in a narrow, chaotic market

- made personalizable *accounting* engine, in *human* terms+documents, user *languages* for reports, templates, operations, access-control. wave-calc. 70k C++ btrieve novell, 3 ppl. Sometimes reworking human-workflows. Still used.
- prototypes of statewide medical record system, doctor's workplace. "obj-oriented" structural documentation flow

sysadmin/ software+accounting analyst, Biochim Bank; 1995-97 - strategic midsize branch of a big state bank

- accounting software - rule-based correctness, ledger conversion, account-scheme on-the-fly-translation, automation
- LAN maintenance - PCs, novell/dos/win/os2, Clipper, C/assembler

lead programmer, Inmeco & Navel, Varna; 1992-1994 - small international companies, naval & military stuff

- Missile-launcher simulators - 3D real-time animated simulation of combat for soldier training - real weapon, overhead projecting on huge screen, modeling airplanes/vehicles, trajectories.. physics; trainer workplace; C++, assembler, 3ppl
- large accurate digital maps: make, compress, view, radar/GPS; i386/penpad
- Danube river map+radar – GUI+map technology; prototype for Gabchikovo channel navigation

publisher+programmer, 1991-93 – many books, papers, cards, art and:

- DTP/preprint software+fonts - reverse-engineer, customize, use many systems, make fonts+tools.. so all works in cyrillic

partner, Specialized Intelligent Systems, Varna 1990-92

- with 2 schoolmates: accounting with replication over modems, Clipper, C, i86 asm

in university..

- set-based unification and Prolog interpreter - thesis, 500Loc prolog ; Text editor in C/asm (PC i86)

in school..

- statistical trends graphical analyzer-visualizer ; simple run-length graphics-compression lib (M6502 Basic + asm)

Education

Technical university Varna (VMEI), 1986-1990

- M.Sc. in Electrical Engineering / Computer Science
- advancing course: 5 years taken in 4, twice more disciplines - all excellent

Mathematical High college - Varna (2 MG), 1983-1986

- "advancing education" class of 4 students, 7 years taken in 4, gold medal

Some community projects

- curated collection of audio records of tales, songs, theaters from LPs, CDs, radio.. and all the machinery behind it - from recording and comparing to publishing.. some old frozen version at <http://gramofonche.chitanka.info> .. Still not abandoned
- making and hosting a community forum and supporting Sladka voda alternative school
- do-it-yourself.. and repair - software, hardware, house, cars, toys, .. anything - www.svilendobrev.com/napravisam
- teaching reverse-engineering/analysis – a few essays

Contacts

e-mail: az@svilendobrev.com

my site: www.svilendobrev.com

work, software, and me: svilendobrev.com/rabota

code, toys and projects: github.com/svilendobrev ; svilendobrev.com/rabota/mysoft.html

cv-resume-biography: [svilendobrev.com/rabota/cv - printable-pdf](http://svilendobrev.com/rabota/cv-printable-pdf) -- [linkedin.com/in/svilendobrev](https://www.linkedin.com/in/svilendobrev)



SvD