



[TL;DR](#) [FULL TEXT](#)

KNOW ALL MEN BY THESE PRESENTS

that

WONG MENG WENG

or Meng Weng Wong, if you're friends from Saints (92) or upenn.edu (SEAS '97)

or Burning Man (03-'15) or Harvard Berkman (16-'17)

having previously co-founded

**POBOX.COM**

exited to Fastmail.fm

**KARMASPHERE.COM**

exited to FICO

**HACKERSPACE.SG**

Singapore's first hackerspace

**JFDI.ASIA**

Singapore's first startup accelerator

and having previously co-authored

**SPF**

(RFC4408)

a globally deployed opensource email standard based on a domain-specific language

TOGETHER WITH

ALEXIS CHUN

a recovering lawyer

previously at TSMP (M&A) and at

Rajah & Tann (commercial litigation)

VIKRAM

VERMA

a self-taught mathematician

specialising in dependently typed

functional languages

VIRGIL

GRIFFITH

a Caltech PhD

previously founder of Wikiscanner

and Onion.link

SEBASTIAAN

DECKERS

a full-stack Javascript developer

previously teaching web development

at General Assembly

JOB CHONG

a self-taught Javascript developer

fresh out of NUS law school

and thus a  $\pi$ -shaped person

AND

a host of other contributors

each equipped with unique superpowers

which humility, not to mention space constraints, forbid adumbrating at this time

IS NOW WORKING ON

LEGALESE.COM

originally an internal project at JFDI Asia

designed to scratch-our-own-itch

that has by our estimation saved JFDI's 69 startups

up to \$345,000 in legal fees

now being spun out as

A SHINY NEW STARTUP

which will apply

MARC ANDREESSEN's dictum

"SOFTWARE IS EATING THE WORLD"

to the **LEGAL INDUSTRY**

by exploiting the

**UNCANNY PARALLELS**

(this gets technical; bear with me)

between

THE DRAFTING OF  
CONTRACTS

and

THE DEVELOPMENT OF  
SOFTWARE

both of which deal in

variable definitions  
event-driven callbacks  
exception handling  
version control

anonymous functions  
if/then/else statements  
dependency graphs  
collaborative editing

and other notions which

COMPUTER SCIENTISTS

are intimately familiar with

but which

LAWYERS

have no names for

not because lawyers are necessarily **UNINTELLIGENT**, but simply because

law **PREDATES** computer science by **THOUSANDS** of years

and because, if you think about it,

most **SENIOR PARTNERS** alive today were in **LAW SCHOOL**

when **FAX MACHINES** were **NEW**

WITH THE SIMPLE GOAL

THAT

what  
**INTUIT**  
has done for  
**ACCOUNTING**

what  
**ADOBE**  
has done for  
**ART & DESIGN**

what  
**AUTODESK**  
has done for  
**ENGINEERING**

LEGALESE  
will do for  
LAW

specifically,

THE \$80B CORPORATE CONTRACTS INDUSTRY

because

(if one may be permitted to speak frankly)

in a world where **CARS** drive **THEMSELVES**

it is **QUITE MAD** to pay a **HUMAN**

**FIVE HUNDRED** an **HOUR**

to **COPY** and **PASTE**

**WORDS** upon a **PAGE**

FIRST BY DEVELOPING A DSL

dubbed "L4"

a **DOMAIN-SPECIFIC-LANGUAGE**

for **LEGAL AGREEMENTS** and **REGULATIONS**

that does for the **MODAL  $\mu$ -CALCULUS**

what functional languages do for the **LAMBDA CALCULUS**

THEN BY BUILDING COMPILERS

which will target **ENGLISH**

and other **NATURAL LANGUAGES**

(imagine multilingual contracts, each provably identical)

and will also target **ETHEREUM**

and other **BLOCKCHAIN PLATFORMS**

(because bugs in smart contracts cost real money)

accompanied by a powerful **STATIC ANALYZER**

capable of **FORMAL VERIFICATION**,

or at any rate **EXHAUSTIVE TEST SUITES**

that prove, to the extent that

such **PROOF** is **MATHEMATICALLY POSSIBLE**,

contracts written in the language are

**CORRECT • CONSISTENT • COMPLETE**

speaking of which

(if one may be frank again)

most programmers write **PROGRAMS**;

fewer (very advanced) coders write **COMPILERS**

which compile the programs to their final form;

but many lawyers **ACT AS** compilers -

it's as if they missed the memo

THUS ESTABLISHING A FOUNDATION

just as **SQL** is foundational to **DATABASES**

**HTML** is foundational to **THE WEB**

**POSTSCRIPT** is foundational to **DIGITAL PRINTING**

FOR A COMPLETE TECH STACK

translating ideas like

**IDES**, **LINTING**, **BUILD TOOLS**,

**UNIT TESTS** and **FUZZ TESTING**,

**OPENSOURCE LIBRARIES**, **VERSION CONTROL**,

and the rest of the **PROGRAMMER'S ARSENAL**

of **TOOLS** and **TECHNIQUES**

THAT CULMINATES IN APPS

for end-users

not law firms!

the first of which (our **MVP**)

helps **STARTUPS** generate **ALL** the paperwork

needed to close their **ANGEL ROUNDS**

which is to say

not just the **TERM SHEET**

but all the **RESOLUTIONS** and **FILINGS** and

other **PREREQUISITES** which most founders don't know about)

WITHOUT

spending **TIME** or **MONEY** on

**LAWYERS**

which **APPLICATIONS** will

slowly but surely

nibble away at the

high-margin

low-volume

manually operated

jealously proprietary

**LAW FIRM**

replacing

it with a

high-volume

low-margin

automated

largely opensource

**NEW INDUSTRY**

which we call

COMPUTATIONAL  
LEGAL

Copyright 2016 Legalese Pte. Ltd. **LEGALESE.COM**

This work is licensed under a Creative Commons  
Attribution-ShareAlike 4.0 International License.



PDF and PNG available at  
<http://computational.legal>