
WRD RD Documentation

Release 0.2.0

**WRD R
D**

Sep 01, 2017

1	WRD R&D Documentation	3
2	Open Source	5
2.1	Open Source Licensing	5
2.2	Open Source Initiative (OSI)	6
2.3	Open Source Governance	6
3	Small Business Consulting	7
3.1	Business	7
3.1.1	Solving	7
3.1.2	Books	7
3.2	Strategy	8
3.3	Business Modeling	8
3.3.1	ROI	8
3.4	Business Plans	9
3.5	Marketing	9
3.5.1	Relationships	9
3.5.2	Marketing Mix	10
3.5.2.1	The Four Ps	10
3.5.3	Sales	10
3.5.4	Metrics	10
3.5.4.1	Conversion	10
3.5.4.2	Web Analytics	11
3.5.5	Brand	11
3.5.5.1	Concept	11
3.5.5.2	Copy	11
3.5.5.3	Graphic Design	11
3.5.5.3.1	Color Scheme	12
3.5.5.3.2	Types	12
3.5.5.3.3	Logo	12
3.5.5.3.4	Favicon	12
3.5.5.3.5	Social Media Images	12
3.5.5.3.6	Photography Checklist	13
3.5.5.3.7	Layout Graphics	14
3.5.6	Social Media	14
3.5.6.1	Twitter	14
3.5.6.2	Zapier	16

3.5.6.3	IFTTT	16
3.5.6.4	OpenGraph	16
3.5.6.5	Location Based Services	17
3.5.6.6	Restaurant	18
3.5.7	Voice Searches to Test	18
4	Web Development	19
4.1	Web Glossary	19
4.2	Web Content	24
4.3	Web Design	25
4.3.1	CSS	25
4.3.2	Web Layout	25
4.3.2.1	Image Based Layouts	25
4.3.3	Screen Captures	26
4.3.4	Bootstrap	26
4.4	Web Development Checklist	27
4.5	Hosting / DNS	30
4.5.1	DNS Configuration	30
4.5.2	Web Hosting	31
4.6	WebSec	31
4.6.1	Responsible Disclosure	31
4.6.1.1	Bug Bounty Program	31
4.6.2	SANS/CIS Consensus Audit Guidelines:	32
4.6.3	CWE	32
4.6.4	OWASP	32
4.6.5	WebSec Forums	32
4.6.6	HTTP	33
4.6.7	HTTPS	33
4.6.8	HTTP STS	33
4.6.9	ACME Protocol	33
4.6.9.1	Letsencrypt	34
4.7	Web Development Requirements	34
4.7.1	User Stories	34
5	Software Development	37
5.1	Computer Science	37
5.1.1	Computer Science Curricula	38
5.1.2	Computer Science Courses	40
5.1.3	Computer Science Exercises	40
5.1.3.1	Code Golf	40
5.2	Project Management	40
5.2.1	Waterfall	41
5.2.2	Everyone Codes	41
5.2.3	Agile	41
5.2.3.1	Stand Up Meeting	41
5.2.3.1.1	Three Questions	41
5.2.3.1.2	Digital Stand Up Meeting	41
5.2.3.2	Extreme Programming	43
5.2.3.3	Unified Process	43
5.2.3.4	Scrum	43
5.2.3.5	Effort Estimation	43
5.2.3.6	Planning Poker	43
5.2.3.7	Kanban	43
5.3	Requirements Management	44

5.3.1	Requirements Modeling	44
5.3.2	Agile Modeling	44
5.3.2.1	Test Driven Development	45
5.3.2.2	Behavior Driven Development	45
5.3.2.2.1	Given-When-Then	45
5.3.2.3	User Story	45
5.3.2.3.1	Card, Conversation, Confirmation	46
5.3.2.4	Use Case	46
5.3.3	Requirements Traceability	47
5.3.3.1	Work Breakdown Structure	48
5.3.3.1.1	Mission	49
5.3.3.1.1.1	Project	49
5.3.3.1.1.2	Goal	49
5.3.3.1.1.3	SMART	49
5.3.3.1.1.4	Objective	49
5.3.3.2	Checkboxes	49
5.3.3.3	Codelabels	50
5.3.3.3.1	Codelabel Syntax	51
5.4	Technical Debt	52
5.5	Refactoring	52
5.6	Continuous Delivery	52
5.6.1	Revision Control	52
5.6.2	Distributed Version Control	53
5.6.2.1	Bisection	53
5.6.2.2	Branching & Merging	53
5.6.3	Test Automation	54
5.6.4	Continuous Integration	54
5.6.5	Continuous Deployment	54
6	Computer Engineering	55
6.1	Monitor	55
6.2	Northbridge	55
6.2.1	Processor	56
6.2.1.1	CPU	56
6.2.1.1.1	ARM	56
6.2.1.1.2	PPC	56
6.2.1.1.3	x86	57
6.2.1.1.4	x86_64	57
6.2.2	Memory	57
6.2.2.1	RAM	57
6.2.3	System Bus	57
6.2.3.1	ISA	58
6.2.3.2	PCI	58
6.2.3.3	PCMCIA	58
6.2.3.4	AGP	58
6.2.3.5	ExpressCard	58
6.2.3.6	PCI-e	58
6.2.4	Video Card	58
6.2.4.1	Multiheaded Display	59
6.2.4.2	GPU	59
6.2.4.2.1	CUDA	60
6.2.4.2.2	PhysX	60
6.2.4.3	VGA	60
6.2.4.4	DVI	61

6.2.4.5	HDMI	61
6.2.4.5.1	Mini HDMI	61
6.2.4.5.2	Micro HDMI	61
6.3	Data Device Bus	62
6.3.1	USB	62
6.3.1.1	USB Hub	63
6.3.1.1.1	USB Keyboard	63
6.3.1.1.2	USB Mouse	63
6.3.1.2	USB Type-C	63
6.3.2	SATA	63
6.3.2.1	eSATA	63
6.3.2.1.1	mSATA	64
6.3.2.1.2	eSATAp	64
6.3.3	SCSI	64
6.3.4	IDE	64
6.3.5	PATA	64
6.3.6	Persistent Storage	65
6.3.6.1	Hard Drives	65
6.3.6.2	SSD	65
6.3.6.3	Disc Drives	65
6.4	Network Interfaces	65
6.4.1	RS-232	65
6.4.2	Gigabit	66
6.4.3	NIC	66
6.4.4	8P8C	66
6.4.5	TIA	66
6.4.5.1	T568A and T568B	67
6.4.5.2	CAT-5	67
6.4.5.3	CAT-5e	67
6.4.5.4	CAT-6	67
6.4.5.5	CAT-6A	67
6.4.6	Ethernet	68
6.4.6.1	Ethernet Crossover Cable	68
6.4.6.2	Patch Cable	68
6.4.6.3	10BASE-T	68
6.4.6.4	100BASE-TX	69
6.4.6.5	Gigabit Ethernet	69
6.4.6.6	1000BASE-T	69
6.4.6.7	1000BASE-X	69
6.4.6.8	10GBASE-T	69
6.4.6.9	40GBASE-T	69
6.4.7	Wireless	70
6.4.7.1	802.11	70
6.4.7.2	Wi-Fi Direct	70
6.4.7.3	Wireless mesh network	71
6.4.8	Mobile Broadband	71
6.4.8.1	3G	71
6.4.8.2	4G	71
6.4.8.3	5G	71
6.4.8.4	CDMA	72
6.4.8.5	GSM	72
6.4.8.6	LTE	72
6.4.9	Fiber	72
6.4.9.1	Fiber to the x	72

6.4.9.2	Fibre Channel	72
6.5	Network Glossary	73
7	Information Systems	77
7.1	Information Systems Glossary	77
7.2	Criteria	79
7.2.1	ROI	79
7.2.2	Need / Want	80
7.2.3	Maintainability	80
7.3	Clouds	80
7.3.1	Cloud Questions	80
7.3.2	Cloud Application Layers	81
7.3.3	SaaS	83
7.3.4	PaaS	84
7.3.4.1	AppEngine	84
7.3.4.2	AppScale	85
7.3.4.3	Deis	86
7.3.4.4	Heroku	87
7.3.5	IaaS	87
7.3.5.1	Amazon AWS	88
7.3.5.2	Google Cloud	88
7.3.5.3	Rackspace Cloud	89
7.4	Configuration Management	90
7.5	Information Security	90
7.5.1	Confidentiality	90
7.5.2	Integrity	90
7.5.3	Availability	90
7.6	Business Continuity	91
7.6.1	Backups	91
7.6.2	Reliability	91
7.6.3	Scenarios	91
7.7	Business Systems	91
7.7.1	Google Apps	91
7.8	Handling Money	92
7.8.1	Accounting	92
7.8.1.1	GNUCash	92
7.8.1.2	Quicken	92
7.8.1.3	QuickBooks	93
7.8.2	Payments	93
7.8.2.1	Amazon Payments	93
7.8.2.2	Apple Pay	93
7.8.2.3	Google Wallet	93
7.8.2.4	PayPal	93
7.8.2.5	Square	93
7.9	Sales	94
7.10	Customer Relationship Management (CRM)	94
7.11	Business Intelligence	94
8	Data Science	95
8.1	datasciencemasters.org	95
8.2	Ten Simple Rules	96
8.2.1	#TenSimpleRules for Reproducible Computational Research	96
8.2.2	#TenSimpleRules for Creating a Good Data Management Plan	98
8.3	Data, Information, Knowledge, & Wisdom	99

8.3.1	Optimization	99
8.3.2	Smoothies	100
8.3.3	Body Temperature	100
8.4	Theory	101
8.4.1	Science	101
8.4.1.1	Cognitive Biases	101
8.4.1.2	Open Science	101
8.4.1.3	Scientific Method	102
8.4.1.3.1	Reproducibility	102
8.4.1.3.2	Systematic Review	102
8.4.1.3.3	Linked Reproducibility	102
8.4.2	Math	103
8.4.2.1	Math Courses	103
8.4.2.2	Project Euler	103
8.4.2.3	Rosalind	103
8.4.2.4	Mathematical Notation	104
8.4.2.4.1	LaTeX	104
8.4.2.4.2	MathJax	104
8.4.2.4.3	MathML	104
8.4.2.4.3.1	ASCIIMathML	104
8.4.2.5	Information Theory	105
8.4.2.6	Linear Algebra	105
8.4.2.7	Calculus	105
8.4.2.8	Statistics	105
8.4.2.8.1	Parametric Statistics	106
8.4.2.8.1.1	Regression Analysis	106
8.4.2.8.2	Nonparametric Statistics	106
8.4.2.8.2.1	Descriptive Statistics	106
8.4.2.8.2.2	Statistical Inference	106
8.4.2.8.3	Causality	106
8.4.3	Analysis	107
8.4.3.1	Learning	107
8.4.3.2	Data Mining	107
8.4.3.3	Data Dredging	107
8.4.3.4	Machine Learning	108
8.4.3.5	Deep Learning	108
8.4.4	Datasets	108
8.4.4.1	awesome-public-datasets	108
8.4.4.2	Awesome	108
8.5	Tools	109
8.5.1	ETL	109
8.5.2	Workflow	109
8.6	Techniques	109
8.6.1	Automated Workflows	109
8.6.2	5 Linked Open Data	110
8.6.3	Data Visualization	110
8.6.3.1	Visualizing Data Science	110
9	Data Visualization Tools	111
9.1	Matplotlib	111
9.1.1	Seaborn	112
9.2	Mayavi	112
9.3	Bokeh	112
9.4	VisPy	112

9.5	Vega	112
9.5.1	Vincent	112
9.6	Plotly	113
9.7	PyQtGraph	113
9.8	qgrid	113
9.9	D3.js	113
9.10	Three.js	113
9.11	Sigmajs	113
10	See Also	115
11	Units	117
11.1	Mass noun	117
11.2	RDF and Units	117
11.2.1	CSVW and Units	117
11.3	Tool Support for Units	117
11.4	Binary Prefixes	118
11.5	Metric System	118
11.5.1	SI Units	118
11.5.2	SI Base Units	119
11.5.3	SI Prefixes	119
11.5.4	SI Distance Units	120
11.5.4.1	metre	120
11.5.5	SI Volume Units	120
11.5.5.1	litre	120
11.5.6	SI Mass Units	121
11.5.6.1	kilogram	121
11.5.6.1.1	gram	121
11.5.7	SI Time Units	121
11.5.7.1	second	121
11.5.7.1.1	microsecond	121
11.5.7.1.2	millisecond	121
11.5.7.1.3	minute	122
11.5.7.1.4	hour	122
11.5.7.1.5	day	122
11.5.7.1.6	week	122
11.5.7.1.7	month	123
11.5.7.1.8	year	123
11.5.8	SI Frequency Units	123
11.5.8.1	hertz	123
11.5.9	SI Electric Current Units	123
11.5.9.1	ampere	123
11.5.9.2	volt	124
11.5.10	SI Power Units	124
11.5.10.1	joule	124
11.5.10.2	watt	124
11.5.11	SI Temperature Units	124
11.5.11.1	celsius	124
11.5.11.2	kelvin	124
11.5.12	SI Amount Units	125
11.5.12.1	mole	125
11.5.13	SI Luminous Intensity Units	125
11.5.13.1	candela	125
11.5.14	SI Data Units	125

11.5.14.1	bit	125
11.5.14.2	byte	125
11.6	Imperial units	126
11.6.1	Imperial Distance Units	127
11.6.1.1	inch	127
11.6.1.1.1	foot	127
11.6.1.1.2	yard	127
11.6.1.1.3	mile	127
11.6.2	Imperial Volume Units	127
11.6.2.1	teaspoon	127
11.6.2.2	dram	128
11.6.2.3	tablespoon	128
11.6.2.4	cup	128
11.6.2.5	fluid ounce	129
11.6.2.6	pint	129
11.6.2.7	quart	129
11.6.2.8	gallon	129
11.6.2.9	keg	129
11.6.3	Imperial Mass Units	130
11.6.3.1	gram	130
11.6.3.2	ounce	130
11.6.3.3	pound	130
11.6.3.4	ton	130
11.7	US customary units	130
11.8	Industry Units	131
11.8.1	Freight	131
11.8.2	Fuel	131
12	Knowledge Engineering	133
12.1	Symbols	133
12.1.1	Character encoding	134
12.1.1.1	Control Characters	134
12.1.1.1.1	Escape Sequences	135
12.1.1.2	ASCII	136
12.1.1.3	Unicode	136
12.1.1.3.1	UTF-8	137
12.2	Logic, Reasoning, and Inference	137
12.2.1	Logic	138
12.2.1.1	Set Theory	138
12.2.1.2	Boolean Algebra	138
12.2.1.3	Many-valued Logic	139
12.2.1.3.1	Three-valued Logic	139
12.2.1.3.2	Fuzzy Logic	139
12.2.1.3.3	Probabilistic Logic	139
12.2.1.4	Propositional Calculus	139
12.2.1.4.1	Modus ponens	139
12.2.1.5	Predicate Logic	140
12.2.1.5.1	Existential quantification	140
12.2.1.5.2	Universal quantification	140
12.2.1.6	Hoare Logic	140
12.2.1.7	First-order Logic	140
12.2.1.8	Description Logic	141
12.2.2	Reasoning	142
12.2.2.1	Inference	142

	12.2.2.1.1	Entailment	142
12.3		Data Engineering	142
	12.3.1	File Structures	146
	12.3.1.1	Git File Structures	146
	12.3.1.1.1	Git Object	147
	12.3.1.1.2	Git Reference	147
	12.3.1.1.3	Git Packfile	147
	12.3.1.1.3.1	bup	147
	12.3.1.2	Torrent file structure	147
12.3.2		File Locking	147
12.3.3		Data Structures	148
	12.3.3.1	Arrays	148
	12.3.3.2	Matrices	148
	12.3.3.3	Lists	149
	12.3.3.4	Graphs	149
	12.3.3.4.1	NetworkX	150
	12.3.3.4.2	DFS	150
	12.3.3.4.3	BFS	150
	12.3.3.4.4	Topological Sorting	151
	12.3.3.5	Trees	151
12.3.4		Compression Algorithms	152
	12.3.4.1	bzip2	152
	12.3.4.2	gzip	152
	12.3.4.3	tar	152
	12.3.4.4	zip	153
12.3.5		Hash Functions	153
	12.3.5.1	CRC	153
	12.3.5.2	MD5	153
	12.3.5.3	SHA	154
12.3.6		Filesystems	154
	12.3.6.1	RAID	154
	12.3.6.2	MBR	155
	12.3.6.3	GPT	155
	12.3.6.4	LVM	155
	12.3.6.5	btrfs	156
	12.3.6.6	ext	156
	12.3.6.7	FAT	156
	12.3.6.8	ISO9660	157
	12.3.6.9	HFS+	157
	12.3.6.10	NTFS	157
	12.3.6.11	FUSE	158
	12.3.6.11.1	SSHFS	158
12.3.7		Network Filesystems	159
	12.3.7.1	Ceph	159
	12.3.7.2	CIFS	159
	12.3.7.3	DDFS	159
	12.3.7.4	GlusterFS	159
	12.3.7.5	HDFS	160
	12.3.7.6	NFS	160
	12.3.7.7	S3	161
	12.3.7.8	Swift	161
	12.3.7.9	SMB	161
	12.3.7.10	WebDAV	161
12.3.8		Databases	161

12.3.8.1	Object Relational Mapping	162
12.3.8.2	Relation Algebra	162
12.3.8.3	Relational Algebra	162
12.3.8.4	Relational Databases	162
12.3.8.4.1	SQL	163
12.3.8.4.2	Drizzle	163
12.3.8.4.3	MySQL	164
12.3.8.4.4	PostgreSQL	164
12.3.8.4.5	SQLite	164
12.3.8.4.6	Virtuoso	165
12.3.8.5	NoSQL Databases	165
12.3.8.6	Graph Databases	165
12.3.8.6.1	Blazegraph	166
12.3.8.6.2	Blueprints	166
12.3.8.6.3	Gremlin	167
12.3.8.6.4	Neo4j	167
12.3.8.7	RDF Triplestores	168
12.3.8.8	Distributed Databases	168
12.3.8.8.1	Accumulo	168
12.3.8.8.2	BigTable	169
12.3.8.8.3	Apache Beam	169
12.3.8.8.4	Cassandra	169
12.3.8.8.5	Hadoop	170
12.3.8.8.6	HBase	170
12.3.8.8.7	Hive	170
12.3.8.8.8	Parquet	171
12.3.8.8.9	Presto	171
12.3.8.8.10	Spark	171
12.3.8.8.10.1	GraphX	172
12.3.9	Distributed Algorithms	172
12.3.9.1	Distributed Computing Problems	172
12.3.9.2	Non-blocking algorithm	173
12.3.9.3	DHT	173
12.3.9.4	MapReduce	173
12.3.9.5	Paxos	173
12.3.9.6	Raft	174
12.3.9.7	Bulk Synchronous Parallel	174
12.3.10	Distributed Computing Protocols	174
12.3.10.1	CORBA	175
12.3.10.2	Message Passing	176
12.3.10.3	ESB	176
12.3.10.4	MPI	176
12.3.10.5	XML-RPC	177
12.3.10.6	JSON-RPC	177
12.3.10.7	Avro	177
12.3.10.8	Protocol Buffers	178
12.3.10.9	Thrift	178
12.3.10.10	SOA	178
12.3.10.10.1	WS-*	178
12.3.10.10.2	WSDL	179
12.3.10.11	JSON-WSP	179
12.3.10.12	ROA	179
12.3.10.12.1	REST	179
12.3.10.13	WAMP	180

12.3.11	Data Grid	181
12.3.12	Search Engine Indexing	181
12.3.12.1	ElasticSearch	181
12.3.12.2	Haystack	182
12.3.12.3	Lucene	182
12.3.12.4	Nutch	182
12.3.12.5	Solr	182
12.3.12.6	Whoosh	183
12.3.12.7	Xapian	183
12.3.12.8	Information Retrieval	183
12.4	Time Standards	183
12.4.1	International Atomic Time (IAT)	183
12.4.2	Long Now Dates	183
12.4.3	Decimal Time	184
12.4.4	Unix Time	184
12.4.5	Year Zero	184
12.4.6	Astronomical year numbering	185
12.4.7	Before Present (BP)	185
12.4.8	Common Era (CE)	185
12.4.9	Time Zones	186
12.4.9.1	UTC	186
12.4.9.2	US Time Zones	187
12.4.9.2.1	US Daylight Saving Time	189
12.4.10	ISO8601	189
12.4.11	NTP	190
12.5	Linked Data	190
12.5.1	5 Linked Data	191
12.6	Semantic Web	191
12.7	Semantic Web Standards	191
12.8	Web Standards	192
12.8.1	Web Names	192
12.8.1.1	URL	192
12.8.1.2	URI	192
12.8.1.3	URN	192
12.8.2	IEC	192
12.8.3	IETF	193
12.8.4	ISO	193
12.8.5	OMG	193
12.8.6	W3C	193
12.8.7	HTTP	193
12.8.7.1	HTTP in RDF	194
12.8.7.2	HTTPS	194
12.8.7.3	HTTP STS	194
12.8.8	CSS	195
12.8.9	RTMP	195
12.8.10	WebSocket	195
12.8.11	WebRTC	196
12.8.12	HTTP/2	196
12.8.13	HTML	196
12.8.13.1	HTML4	196
12.8.13.2	XHTML	197
12.8.13.3	HTML5	197
12.8.14	XML	197
12.8.15	XSD	197

12.8.16	JSON	198
12.8.17	CSV	198
12.8.17.1	CSVW	198
12.8.18	RDF	199
12.8.18.1	RDF Interfaces	199
12.8.18.2	N-Triples	201
12.8.18.3	RDF/XML	201
12.8.18.4	TriX	201
12.8.18.5	N3	201
12.8.18.6	Turtle	202
12.8.18.7	TriG	202
12.8.18.8	RDFa	202
12.8.18.9	RDFa 1.1 Core Context	202
12.8.18.10	JSON-LD	203
12.8.19	RDFS	203
12.8.19.1	DCMI	204
12.8.19.2	EARL	204
12.8.19.3	RDF Data Cubes	205
12.8.19.4	SKOS	205
12.8.19.5	XKOS	205
12.8.19.6	FOAF	205
12.8.19.7	SHACL	206
12.8.19.8	SIOC	206
12.8.19.9	OA	206
12.8.20	Schema.org	207
12.8.20.1	Schema.org RDF	207
12.8.20.2	Schema.org TopBraid RDF	208
12.8.21	SPARQL	208
12.8.22	LDP	208
12.8.23	OWL	209
12.8.23.1	PROV	210
12.8.23.2	DBpedia	210
12.8.23.3	QUDT	212
12.8.23.4	QUDT Quantities	212
12.8.23.5	QUDT Units	213
12.8.23.6	Wikidata	215
12.9	Semantic Web Tools	215
12.9.1	CKAN	215
12.9.2	Protégé	216
12.9.3	RDFJS	216
12.9.4	RDFHDT	216
12.9.5	RDFLib	217
12.10	Semantic Web Schema Resources	217
12.10.1	prefix.cc	217
12.10.2	LOV	217
12.10.3	URIs for Units	218
12.10.4	LODCloud	218
13	LinkedReproducibility	219
13.1	StudyGraph: Document Nodes and Link Edges	219
13.2	StructuredPremises: Premises as structured data	221
13.3	LinkedMetaAnalyses	222
13.4	RDF Example	223
13.5	CSV, CSVW, and metadata rows	223

13.6	References	224
14	Research	225
14.1	Folders and Labels	225
14.2	Citation Metadata	225
14.3	Search engines	226
14.4	Research Tools	227
14.4.1	Mendeley	227
14.4.2	Zotero	227
14.4.3	CKAN	228
14.4.4	DSpace	228
14.4.5	Fedora Commons	228
14.4.6	Hydra	229
14.4.7	Blacklight	229
14.4.8	Islandora	229
14.4.9	OAI-PMH	229
15	Entrepreneurship	231
15.1	SBA	231
15.2	Planning	232
15.2.1	Plan for Failure	232
15.2.1.1	Contingency plan	232
15.2.1.2	Exit strategy	232
15.2.2	Plan for Success	233
15.2.2.1	Mission Statement	233
15.2.2.2	Project	233
15.2.2.3	Goal	233
15.2.2.4	Objective	233
15.2.2.5	Imagine Success	233
15.2.2.6	Start at the end	233
15.2.3	Profitability	233
15.3	USA Entrepreneurship	234
15.3.1	Types of business entity	234
15.4	Economics	236
16	Personal Finance	237
16.1	Dave Ramsey	237
16.2	Personal Budget	237
16.2.1	Personal Income	238
16.2.2	Occupational Outlook Handbook	238
16.2.3	Personal Expenses	238
16.2.4	Personal Debt	238
16.3	Interest	238
16.3.1	Compound interest	239
16.3.1.1	Doubling time	239
16.3.1.1.1	Rule of 72	239
16.4	Taxes	239
16.4.1	Income tax	239
16.4.2	Property tax	239
16.4.3	Capital gains tax	239
16.4.4	Gift tax	240
16.5	Charitable contributions	240
16.5.1	Charity Navigator	240
16.6	Retirement	241

16.6.1	Pension	241
16.6.1.1	Defined benefit pension plan	241
16.6.1.2	Social Security	241
16.6.2	401(k)	241
16.6.2.1	401(k) match	241
16.6.2.2	401(k) rollover	241
16.6.3	IRA	242
16.6.3.1	Roth IRA	242
16.7	Healthcare	242
16.7.1	Health insurance	242
16.7.2	Health Savings Account	242
16.7.3	Long Term Care	243
16.7.4	Advance Care Directive	243
16.8	Estate planning	243
16.8.1	Trust	243
16.8.2	Will and Testament	243
16.8.2.1	Estate tax	244
16.8.2.2	Inheritance tax	244
16.8.2.3	Intestacy	244
16.8.2.4	Digital Will	245
17	Investing	247
17.1	Basics	247
17.1.1	Cash flow	247
17.1.1.1	Revenue	247
17.1.2	Accounting Equation	248
17.1.2.1	Equity	248
17.1.2.1.1	Asset	248
17.1.2.1.2	Liability	248
17.1.3	Financial Statement	248
17.1.4	Capitalization table	249
17.1.5	Option	249
17.1.5.1	Stock Option	249
17.1.5.1.1	Employee Stock Option	249
17.1.6	SAFE	249
17.2	Investing Strategies	250
17.2.1	Investing Styles	250
17.2.2	Value Investing	250
17.2.3	Buy and Hold	250
17.2.4	Day trading	250
17.2.5	Algorithmic trading	250
17.2.5.1	High Frequency Trading	251
17.2.6	Stock Market Crash	251
17.2.7	Recession	251
17.2.8	Quantitative analyst	251
17.3	Tools	251
17.3.1	pandas	251
17.3.1.1	pandas-datareader	252
17.3.2	Quandl	252
17.3.2.1	quandl-python	252
17.3.3	QuantLib	253
17.3.3.1	pyql	253
17.3.4	Quantopian	253
17.3.4.1	Quantopian Open	254

17.3.4.2	Zipline	254
17.3.4.3	qgrid	254
17.3.5	See Also	254
18	Education Technology	255
18.1	STEM	255
18.2	Publishing	255
18.2.1	GitHub Pages	256
18.2.2	GitLab Pages	257
18.2.3	ReadTheDocs	257
18.2.4	Self Directed Learning	257
18.3	Online Courses	258
18.3.1	Class Central	258
18.3.2	Coursera	258
18.3.3	edX	259
18.3.3.1	Jupyter and edX	259
18.3.4	Udacity	259
18.4	Jupyter and Learning	259
18.4.1	Jupyter and Reproducibility	260
18.4.2	Jupyter and TDD	261
18.4.2.1	nbgrader	262
18.4.3	JupyterHub Servers	262
18.5	Knowledge Engineering	262
18.5.1	Linked Curricula Graphs	262
18.5.2	OpenBadges	263
18.5.3	OpenBadges Backpack	263
19	Educational Testing	265
19.1	Educational Objectives	265
19.2	Educational Testing Objectives	265
19.3	USA Testing Standards	265
19.3.1	USA K-12 Curriculum Standards	266
19.3.1.1	Common Core	266
19.3.1.2	High School Diploma	266
19.3.1.2.1	GED	266
19.3.1.3	College Credit Highschool Courses	266
19.3.1.3.1	AP	266
19.3.1.3.2	CLEP	267
19.3.1.3.3	IBDP	267
19.3.1.3.4	Dual Enrollment	267
19.3.2	USA State Testing Standards	267
19.3.3	USA Higher Education Testing	267
19.3.3.1	Undergraduate	267
19.3.3.1.1	SAT	267
19.3.3.1.2	ACT	268
19.3.3.1.3	ASVAB	268
19.3.3.2	Graduate	268
19.3.3.2.1	GRE	268
19.3.3.2.2	GMAT	269
19.3.3.2.3	LSAT	269
19.3.3.2.4	MCAT	269
19.3.3.3	Postgraduate	270
19.3.3.3.1	Dissertation / Thesis	270
19.3.3.3.2	Master's Degree	270

19.3.3.3.3	Doctorate	271
19.3.3.4	Honorary Degree	271
20	Art & Design	273
20.1	Media	273
20.2	Visual Composition	273
20.3	Graphic Design	273
20.3.1	Web Design	274
20.3.2	BMP	274
20.3.3	GIMP	274
20.3.4	GIF	274
20.3.5	ICO	275
20.3.6	Inkscape	275
20.3.7	JPEG	275
20.3.8	PNG	275
20.3.9	SVG	275
20.4	3D Modeling	276
20.4.1	Blender	276
20.4.2	NURBS	276
20.4.3	OpenGL	276
20.4.4	Rhinoceros 3D	276
20.4.5	SketchUp	277
20.4.6	WebGL	277
20.5	Mediated Reality	277
20.5.1	Mixed Reality	277
20.5.1.1	Augmented Reality	277
20.5.1.1.1	Virtual Reality	278
20.5.2	Mediated reality mediums	278
20.5.2.1	Google Cardboard	278
20.5.2.2	Google Glass	278
20.5.2.3	HTC Vive	278
20.5.2.4	Oculus VR	278
20.5.2.4.1	Oculus Rift	278
20.5.2.4.2	Samsung Gear VR	279
20.5.2.5	Windows Holographic	279
20.5.2.5.1	Microsoft HoloLens	279
20.6	Photography	279
20.7	Videography	279
20.8	Video Production	280
20.8.1	Cinelerra	280
20.8.2	CinePaint	280
20.8.3	FFmpeg	280
20.8.4	FLV	281
20.8.5	HandBrake	281
20.8.6	H.264	281
20.8.7	Lightworks	281
20.8.8	LiVES	282
20.8.9	MPEG	282
20.9	Screenwriting	283
20.9.1	Fountain.io	283
20.10	Songwriting	283
20.10.1	Poetry	283
20.11	Scorewriting	283
20.11.1	LilyPond	284

20.11.2	MuseScore	284
20.11.3	Musical Notation	284
20.11.3.1	MusicXML	285
20.11.3.2	Tablature	285
20.11.4	Rosegarden	285
20.11.5	TablEdit	285
20.12	Musical Composition	286
20.12.1	Audio Synthesis	286
20.12.2	ChucK	286
20.12.3	Csound	287
20.12.4	Hookpad	287
20.12.5	Hydrogen	287
20.12.6	MIDI	288
20.12.7	STK	288
20.12.8	VST	288
20.13	Sound Production	288
20.13.1	Acoustical Engineering	289
20.13.2	DJ	289
20.13.3	AAC	289
20.13.4	Ardour	289
20.13.5	Audacity	289
20.13.6	FLAC	290
20.13.7	JACK	290
20.13.8	LADSPA	291
20.13.9	Mixxx	291
20.13.10	MP3	291
20.13.11	PulseAudio	291
20.13.12	WAV	293
20.14	Audio/Visual Production	293
20.14.1	VJ	293
20.14.2	Advanced Visualization Studio	293
20.14.3	Geiss	294
20.14.4	libvisual	294
20.14.5	MilkDrop	294
20.14.5.1	MilkDrop Presets	295
20.14.6	ProjectM	295
20.14.7	Resolume	295
20.14.8	vvvv	295
20.15	Web Production	296
20.15.1	M3U	296
20.15.2	PLS	296
20.15.3	Popcorn Maker	296
20.15.4	Stream Annotation	296
20.15.4.1	OpenAnnotation	297
20.15.5	XSPF	297
20.15.5.1	JSPF	298
20.16	Web Distribution	298
20.16.1	CDN	298
20.16.2	BitTorrent	299
20.16.2.1	debtorrent	300
20.16.2.2	webtorrent	300
20.16.3	Named Data Networking	301
20.16.4	Steam	301
20.16.4.1	Kerbal Space Program	301

20.16.5	Tube Sites	301
20.16.6	HTTP Live Streaming	302
20.16.7	MPEG-DASH	302
20.16.8	nginx-rtmp-module	303
20.17	Video Chat	303
20.18	Media Tools	303
20.18.1	DreamStudio	304
20.18.2	Ubuntu Studio	304
20.18.3	VLC	304
20.19	Home A/V	305
20.19.1	Standards	305
20.19.1.1	DIAL	305
20.19.1.2	DLNA	305
20.19.1.3	Google Cast SDK	305
20.19.1.4	SSDP	306
20.19.1.5	UPnP	306
20.19.1.6	Zeroconf	306
20.19.2	10-foot interfaces	306
20.19.2.1	Gaming Consoles	306
20.19.2.2	Standalone Media Devices	307
20.19.2.2.1	Android TV	307
20.19.2.2.2	Chromecast	307
20.19.2.2.2.1	Chromecast Audio	307
20.19.2.2.3	Apple TV	308
20.19.2.2.4	Kodi	308
20.19.2.2.5	MythTV	308
20.19.2.2.6	Plex	309
20.19.2.2.7	Roku	309
21	UI / UX Design	311
21.1	Human-Computer Interaction (HCI)	311
21.2	Usability	311
21.3	Experience Design (XD)	312
21.4	User Experience (UX)	312
21.5	User Interface (UI)	312
21.6	Information Design	312
22	Team Building	313
22.1	Personal Development	313
22.1.1	Golden Rule	313
22.1.2	Emotional Intelligence	314
22.2	Collaboration Engineering	314
22.2.1	Seven Layer Model of Collaboration	314
22.2.2	Six Patterns of Collaboration	315
22.2.3	ThinkLets	315
22.3	Collaborative Software	315
22.3.1	The same page	316
22.3.2	Group Support Systems	316
22.3.3	Text Templates	317
22.4	Collaboration Checklist	317
23	Collaboration Plan	319
23.1	Sharing Files	319
23.2	Mailing List	319

23.3	Project Site	320
23.4	Re: Document-based collaboration	320
24	Credits	323
24.1	WRD R&D	323
24.2	Lead	323
24.3	Contributors	323
25	License	325
26	Tools	327
27	wrdrd	329
27.1	wrdrd	329
27.1.1	Installation	329
27.1.2	Usage	329
27.1.3	domaintool	329
27.1.4	crawl	329
27.1.5	stripsinglehtml	330
27.2	wrdrd.tools package	330
27.2.1	Submodules	330
27.2.2	wrdrd.tools.crawl module	330
27.2.3	wrdrd.tools.domain module	330
27.2.4	wrdrd.tools.stripsinglehtml module	333
27.3	Credits	333
27.3.1	WRD R&D	333
27.3.2	Lead	333
27.3.3	Contributors	333
27.4	History	333
27.4.1	0.1.0 (2014-1-11)	333
27.4.2	0.2.0 (2014-10-14)	333
27.5	License	334
28	Indices and tables	335
	Python Module Index	337

Docs: <https://wrdrd.github.io/docs/>
Src: <https://github.com/wrdrd/docs>

CHAPTER 1

WRD R&D Documentation

Homepage: <https://wrdrd.github.io/docs/>
Source: git <https://github.com/wrdrd/docs>
Twitter: <https://twitter.com/wrdrd>

The latest HTML version of WRD R&D Documentation is at <https://wrdrd.github.io/docs/>.

The Sphinx ReStructuredText sources of WRD R&D Documentation are at: <https://github.com/wrdrd/docs>.

The WRD R&D Documentation includes the <https://github.com/westurner/tools> Tools Sphinx documentation set as a Git subrepository as well.

<https://wrdrd.github.io/docs/>

https://en.wikipedia.org/wiki/Open_source

https://en.wikipedia.org/wiki/Open-source_software

https://en.wikipedia.org/wiki/Open-source_software_security#The_debate

https://en.wikipedia.org/wiki/Comparison_of_open-source_software_hosting_facilities

https://en.wikipedia.org/wiki/Network_effect

https://en.wikipedia.org/wiki/Open-source_movement

Open Source Licensing

https://en.wikipedia.org/wiki/Open-source_license

https://en.wikipedia.org/wiki/Comparison_of_free_and_open-source_software_licenses

<http://opensource.org/licenses>

- <http://choosealicense.com/licenses/>
- <https://github.com/github/choosealicense.com>
- <https://developer.github.com/v3/licenses/>
- Q: What is the default license of [public GitHub repositories]?

A: All Rights Reserved; though free public repositories are for open source projects (which must be forkable and changeable).

When you create a new repository, you have the option to select an Open Source license (and a `.gitignore`) to initially commit to the new repository.

<https://stackoverflow.com/questions/4007674/whats-the-default-license-of-code-published-at-github>

- Q: Do you have to include a license notice with every source file?

A: No. A LICENSE (.txt, .md (Markdown), .rst (ReStructuredText), or similar is sufficient to indicate that the copyright is anything other than All Rights Reserved, by default.

<https://programmers.stackexchange.com/questions/125836/do-you-have-to-include-a-license-notice-with-every-source-file/125881#125881?newreg=32f66dddfdc04c03975d9f39a2f3a918>

Open Source Initiative (OSI)

Wikipedia: https://en.wikipedia.org/wiki/Open_Source_Initiative

Homepage: <http://opensource.org/>

- https://en.wikipedia.org/wiki/Open_Source_Initiative#See_also

<http://opensource.org/licenses>

Open Source Governance

Business

<https://en.wikipedia.org/wiki/Business>

https://en.wikipedia.org/wiki/Small_business

<https://en.wikipedia.org/wiki/Enterprise>

Solving

Solving a:

- need
- challenge
- problem
- gap
- remaining piece

Books

- The Personal MBA: Master The Art of Business

<https://personalmba.com/>

<https://personalmba.com/best-business-books>

- <https://www.quora.com/What-books-should-entrepreneurs-read>

- https://en.wikipedia.org/wiki/The_Five_Dysfunctions_of_a_Team
- https://en.wikipedia.org/wiki/Getting_Things_Done

See also:

- *Art & Design > Screenwriting, Web Production, Web Distribution*
- *Education Technology > Publishing*
- *Team Building*

Strategy

<https://en.wikipedia.org/wiki/Strategy>

<https://en.wikipedia.org/wiki/Goal>

https://en.wikipedia.org/wiki/Strategic_planning

https://en.wikipedia.org/wiki/Strategic_thinking

https://en.wikipedia.org/wiki/Strategic_management

https://en.wikipedia.org/wiki/Management_by_objectives

[https://en.wikipedia.org/wiki/Tactic_\(method\)](https://en.wikipedia.org/wiki/Tactic_(method))

https://en.wikipedia.org/wiki/Business_process

Business Modeling

https://en.wikipedia.org/wiki/Business_model

https://en.wikipedia.org/wiki/Cash_flow

<https://en.wikipedia.org/wiki/Risk>

- **Business Model examples:** <https://gist.github.com/ndarville/4295324>
- https://en.wikipedia.org/wiki/List_of_the_largest_software_companies
- https://en.wikipedia.org/wiki/Business_models_for_open-source_software

ROI

https://en.wikipedia.org/wiki/Return_on_investment

How do we justify this expense in terms of Return on Investment?

How do we quantify organic / word-of-mouth marketing?

In terms of sales, what's a reasonable performance goal?

In terms of market share, what's a reasonable performance goal?

Are there benchmarks for how / what competitors are doing in this space?

What are our desired outcomes from this experiment?

What do we want to happen when our message reaches people?

See: *Information Systems > ROI*

Business Plans

https://en.wikipedia.org/wiki/Business_plan

- https://en.wikipedia.org/wiki/Business_plan#Content
- <https://www.sba.gov/writing-business-plan>
- https://en.wikipedia.org/wiki/Open_business#Main_ideas

“How to make your business plan stand out?”

- Data (see: *Data Science*)
- Projections: Pessimistic, Realistic, Optimistic
- Version Number: (see <http://semver.org>)
- Collaborative Document: (it will change)
 - Wiki
 - Live Doc
 - Publish to HTML and PDF:
 - * <https://github.com/westurner/wiki/blob/master/Makefile> (Make, Sphinx)
 - * <https://westurner.github.io/pages/resume> (ReStructuredText)

Marketing

<https://en.wikipedia.org/wiki/Marketing>

https://en.wikipedia.org/wiki/Marketing_plan

https://en.wikipedia.org/wiki/Online_marketing

Relationships

https://en.wikipedia.org/wiki/Customer_relationship_management

https://en.wikipedia.org/wiki/Business_relationship_management

https://en.wikipedia.org/wiki/Enterprise_relationship_management

https://en.wikipedia.org/wiki/Interpersonal_relationship

https://en.wikipedia.org/wiki/Tuckman's_stages_of_group_development

Interactions: Cold, Warm, and Hot

See: *Information Systems > Customer Relationship Management (CRM)*

See: *Team Building*

Marketing Mix

https://en.wikipedia.org/wiki/Marketing_mix

The Four Ps

https://en.wikipedia.org/wiki/Marketing_mix#McCarthy.27s_four_Ps

- Product
- Price
- Promotion
- Place

Sales

<https://en.wikipedia.org/wiki/Sales>

https://en.wikipedia.org/wiki/Sales_process_engineering

https://en.wikipedia.org/wiki/Profit-based_sales_targets

See: *Information Systems* > *Sales*

See: *Information Systems* > *CRM*

Metrics

https://en.wikipedia.org/wiki/Performance_metric

https://en.wikipedia.org/wiki/Performance_indicator

https://en.wikipedia.org/wiki/Critical_success_factor

See: *Data Science*

Conversion

https://en.wikipedia.org/wiki/Conversion_marketing

- https://en.wikipedia.org/wiki/Conversion_marketing#Conversion_rate
- https://en.wikipedia.org/wiki/Conversion_marketing#Common_Conversion_Marketing_Services

https://en.wikipedia.org/wiki/Conversion_rate_optimization

- https://en.wikipedia.org/wiki/Conversion_funnel
- https://en.wikipedia.org/wiki/Purchase_funnel

Web Analytics

https://en.wikipedia.org/wiki/Web_analytics

- [https://en.wikipedia.org/wiki/Impression_\(online_media\)](https://en.wikipedia.org/wiki/Impression_(online_media))
- https://en.wikipedia.org/wiki/Conversion_marketing#Conversion_rate
- Total Time Reading (TTR)
- [] Get Analytics tracking id(s)
- [] Add Analytics JS to page
- [] Configure factors variables conditions

Brand

<https://en.wikipedia.org/wiki/Brand>

- https://en.wikipedia.org/wiki/Brand#Branding_strategies

Concept

- Audience
 - Who are you trying to reach?
- Actions
 - What do you want them to do?
- Emotions
 - How do you want them to feel?

Copy

[https://en.wikipedia.org/wiki/Copy_\(written\)](https://en.wikipedia.org/wiki/Copy_(written))

<https://en.wikipedia.org/wiki/Copywriting>

https://en.wikipedia.org/wiki/Copy_editing

<https://en.wikipedia.org/wiki/Proofreading>

https://en.wikipedia.org/wiki/Copy_testing

- Keywords and phrases
- *Concept*

See also: *Art & Design > Screenwriting*

Graphic Design

https://en.wikipedia.org/wiki/Graphic_design

See: *Art & Design > Graphic Design*

Color Scheme

https://en.wikipedia.org/wiki/Color_scheme

- https://en.wikipedia.org/wiki/Color_chart
- Web: https://en.wikipedia.org/wiki/Web_colors
 - [] Style guide: What are our current *Brand* hexadecimal color codes?
- Print, Manufacturing: <https://en.wikipedia.org/wiki/Pantone>

Types

https://en.wikipedia.org/wiki/Typeface#Style_of_typefaces

https://en.wikipedia.org/wiki/Web_typography

<http://www.google.com/fonts>

- Web-compatible font names: serif, sans-serif, monospace

Logo

https://en.wikipedia.org/wiki/Logo#Internet-compatible_logos

Logo Graphic image

Logotype Stylized business name

Wordmark See: *logotype*

Favicon

<https://en.wikipedia.org/wiki/Favicon>

- *SVG* -> *ICO*, *PNG*
- [] PNG: 16x16, 32x32, 64x64, 128x128, 256x256, 512x512
- [] Apple Touch: 57x57, 72x72, 114x114, 144x144
- Transparent backgrounds work well

Social Media Images

Facebook

- Profile: 160x160
- Profile: 50x50
- Cover: 1702x630 (resized to 851x315)

Twitter

- Profile: 500x500 (max.)
- Profile: 73x73

- Profile: 48x48
- Profile: 24x24
- Header: 1252x626 (min. visible: 520x260)

LinkedIn

- Cover: 646x220
- Standard Logo: 100x60
- Square Logo: 50x50
- Careers Cover: 974x238
- Product Image: 100x80
- Profile: 450x450
- Profile: 200x200
- Profile: 65x65

YouTube

- Channel Icon: 800x800
- Channel Icon: 90x90
- Channel Art (TV): 2120x1192
- Channel Art (desktop): 1060x175
- Channel Art (tablet): 768x175
- Channel Art (mobile): 640x175

Google+

- Profile: 250x250
- Cover: 2120x1192

Pinterest

- Profile: 600x600
- Profile: 165x165

Sources:

- Chrome DevTools (Inspect Element)
- http://www.prosar.com/inbound_marketing_blog/bid/181457/Cheat-Sheet-Image-Sizes-for-Twitter-Facebook-Pinterest-Google-
- <http://thefinancialbrand.com/30777/facebook-twitter-youtube-linkedin-profile-image-sizes/>
- http://help.linkedin.com/app/answers/detail/a_id/32617

Photography Checklist

- *Photography* Guidelines
 - Lighting: Event Time of day, sources, colors, shine
 - Perspective: Closeups, Angles
 - Balance: exposure, color balance, contrast, brightness, gamma filters

- [] Location: Building, Storefront, Hallway
- [] People: Stock photos, smiles, grimaces (“what’s (s)he saying?”)
- [] Things: Products, Services, Outcomes
- [] Source Rights
- [] Offsite Backups (CD/DVD, USB Drive)
- [] Photo Gallery; Hosting
 - [] Thumbnails
 - [] Next/Previous
 - [] Multitouch Zoom (“pinch to zoom”)
 - [] Album Browser
 - [] Carousel API

Layout Graphics

See: *Web Development > Web Design > Web Layout*

- [] Acquire *Web Content* from Photographer, Logo Designer, Web Developer/Maintainer
 - *SVG* vector images are often more rescalable and reusable
 - Which font is the / best matches the *wordmark*?
- Legacy websites tend to have embedded image layouts (because style)
 - [] Contact Original Designer / Copyright Owner
 - [] Crop images (e.g. *Logo*, *wordmark*) from image layout
 - [] Match *Types* and *Color Scheme*

Social Media

https://en.wikipedia.org/wiki/Social_media

https://en.wikipedia.org/wiki/Social_technology

https://en.wikipedia.org/wiki/Social_networking_service

https://en.wikipedia.org/wiki/List_of_social_networking_websites

“I want to favorite/like/share/upvote this but it’s not I can’t just copy the URL I want to click”

Twitter

Wikipedia: <https://en.wikipedia.org/wiki/Twitter>

Homepage: <https://twitter.com>

Homepage: <https://twtr.com>

Source: <https://github.com/twitter>

Twitter: <https://twitter.com/twitter>

Twitter: <https://twitter.com/gov>

Twitter: <https://twitter.com/TwitterIR>

Twitter: <https://twitter.com/TwitterSmallBiz>

Twitter: <https://twitter.com/support>

Twitter: <https://twitter.com/TwitterDev>

Twitter: <https://twitter.com/TwitterEng>

Twitter: <https://twitter.com/TwitterAPI>

Twitter: <https://twitter.com/TwitterOSS>

Twitter: <https://twitter.com/TwitterData>

Twitter: <https://twitter.com/design>

Twitter: <https://twitter.com/TwitterMedia>

Twitter: <https://twitter.com/TwitterStage>

Twitter: <https://twitter.com/TwitterMusic>

Twitter: <https://twitter.com/TwitterMovies>

Twitter: <https://twitter.com/TwitterSports>

Twitter: <https://twitter.com/TwitterFood>

Twitter: <https://twitter.com/TwitterBooks>

Twitter: <https://twitter.com/Nonprofits>

Twitter: <https://twitter.com/TwitterAds>

Twitter: <https://twitter.com/TwitterForNews>

Twitter: <https://twitter.com/TwitterMoments>

Docs: <https://support.twitter.com/>

Docs: <https://dev.twitter.com/>

Docs: <https://dev.twitter.com/cards/overview>

- Are you tweetable?
- Can I (send you a) tweet you sometime?
- What's your Twitter [(optional) username / handle / *URL*]?
- When you add @username (with the @ *at* symbol, the username is then linked and added to a notification list.
 - Tweets that start with @username show in the “Tweets & replies” Twitter User Profile view.
 - Tweets that start with .@username show in the “Tweets” Twitter User Profile view; may be linked replies; and still do send a notification to @username and any other @-addressed Tweeters.
- When you add #example (with the # *hashtag* symbol) before a word – like in IRC clients and old BBS systems – the hashtag becomes a link to a current search for other tweets containing that hashtag and/or similar terms (after the 140 character limit).
 - (e.g. regex #\w+), except for Unicode; and Right-To-Left languages.
 - <https://github.com/twitter/twitter-text-js>
- #webdev: * *HTML* templates: Add Twitter cards *RDFa* attributes and vocabulary markup
- Support: Can we enqueue these for followup?
 -
 -
 - *Zapier: Twitter <-> <API>*
 - * <https://zapier.com/zapbook/twitter/>
 - * New tweet <-> New sheet row

- * New tweet -> New support ticket / issue / case
- * New tweet <-> New webchat (e.g. Slack, Gitter,) message
- * {...}

Zapier

Homepage: <https://zapier.com/>

Source: <https://github.com/zapier>

Twitter: <https://twitter.com/zapier>

Twitter: <https://twitter.com/zapierstatus>

Docs: <https://zapier.com/zapbook/>

Docs: <https://zapier.com/zapbook/use-cases/>

Docs: <https://zapier.com/status/>

Docs: <https://zapier.com/help/>

Docs: <https://zapier.com/developer/>

Docs: <https://zapier.com/developer/documentation/v2/>

Docs: <https://zapier.com/developer/documentation/v2/scripting/>

Docs: <https://zapier.com/developer/documentation/v2/reference/>

Docs: <https://zapier.com/engineering/>

Docs: <https://zapier.com/learn/the-ultimate-guide-to-remote-working/>

Zapier (z' API-er, ~"happier") is a software-development-free *SaaS* web service for integrating 600+ great web APIS through Zaps: *triggers* (when this) and *actions* (do this).

- Users can find and select Zaps from the Zapbook (App Directory).
- Users can CRUD (create, read, update (edit), delete) Zaps.
- Zaps run in worker instances with free and paid time and memory limits.
- Users can review Zap logs and data.

See also:

- *Given-When-Then*

IFTTT

Homepage: <https://ifttt.com/>

Src: <https://github.com/ifttt>

See also:

- *Given-When-Then*

OpenGraph

Wikipedia: https://en.wikipedia.org/wiki/Facebook_Platform#Open_Graph_protocol

Homepage: <http://ogp.me/>

OpenGraph is an *RDFa* standard for structured data.

- When you share a link, Facebook reads the OpenGraph *RDFa* tags from the linked *HTML* for e.g. name, description, image.

See:

- *Web Development Checklist*
- *Knowledge Engineering > Linked Data*
- *Schema.org* – <https://schema.org/Thing>

Location Based Services

https://en.wikipedia.org/wiki/Location-based_service

- https://en.wikipedia.org/wiki/Location-based_service#Location_Based_Marketing_Best_Practices
- Google Maps
 - Directions Link
 - Static Map / Directions Images
 - <https://developers.google.com/maps/documentation/imageapis/>
 - * Map Image
 - * Street View Image
 - Maps Widget
 - <https://developers.google.com/maps/documentation/embed/start>
 - Check the directions
 - <https://www.google.com/mapmaker/>
 - “Get your business on Google”
 - * <https://www.google.com/business/>
- Bing Maps
 - Directions Link
 - Static Map / Directions Images
 - <https://www.bing.com/maps/embed/Customize.aspx>
 - * Map Image
 - Maps Widget
 - <https://www.bing.com/maps/embed/Customize.aspx>
 - Check the directions
 - <https://www.bing.com/maps/> > “Feedback”
 - “Add your business to Bing in 3 easy steps”
 - * <https://www.bingplaces.com/>
- Facebook
 - Check in to the page with location services on

- [] Check in to the page with location services off
- Foursquare
 - [] Badge into the mayorship

Restaurant

- *Web Content*
 - *Photography Checklist*: Foodie photos are normally close-ups at an angle
 - * Top-down / bird's eye photos are not as appealing
- Online Reviews
 - <http://schema.org/review>
 - UrbanSpoon
 - * Photos
 - Yelp
 - * Photos
 - Zagat

Voice Searches to Test

- [] “Directions to _____ [in <city>, <state>]”
 - *Schema.org*: <https://schema.org/location>
- [] “What time does _____ open/close?”
 - *Schema.org*: <https://schema.org/openingHours>
- [] “Where can I find _____ in <city>, <state>?”

See: *Web Development Checklist*

Wikipedia: https://en.wikipedia.org/wiki/Web_development

Web Glossary

Web Page A *web page* is a single (“static”) *HTML* page.

- Sometimes, when people say *web page* they really mean the whole *website* (as composed by e.g. a *web application* backed by a *database* or a static page generator with support for managing collections of interlinked *HTML* pages (e.g. Sphinx, Tinkerer).

Wikipedia: https://en.wikipedia.org/wiki/Web_page

Website

Web site A website is a set of interlinked *HTML* pages. Also used to refer to a web page.

Wikipedia: <https://en.wikipedia.org/wiki/Website>

Web Application A web application is a software application running on a server which returns responses to specific requests and may interact with a database, email services, etc.

- Like all *systems*, a web application works with variable data inputs and outputs.

Wikipedia: https://en.wikipedia.org/wiki/Web_application

Docs: *Information Systems > Cloud Application Layers*

Web Hosting Service A web hosting service offers virtual or physical server resource space for a web page, web site, or web application.

- Shared, Dedicated, Virtual Private, Cloud (Virtual Instance)
- Managed (24/7 Supported // n hours a month), Unmanaged (file a support request and wait)

Wikipedia: https://en.wikipedia.org/wiki/Web_hosting_service

Docs: *Clouds*

Docs: *Web Hosting*

MAC address A MAC (*Media Access Control*) address is a 48-bit identifier.

- *NIC* cards have a *MAC address*.
- *Wireless* cards a *MAC address*.
- Many *Wireless* Access Points (*AP*) MAY all have the same text SSID but different *MAC addresses* (e.g. for *wireless mesh networking* or *WDS*)
- *DHCP IP address* leases are tied to a *MAC address*.
- A *MAC address* identifies a node to the nearest packet routing link segment (e.g. AP, Switch, Hub, Bridge).
- Some *IPv6 addresses* contain the link *Mac address*.
- *UUID* version 1 128-bit identifiers contain a link *MAC address* and the date/time. (Python `import uuid; print(uuid.uuid1())`)

Wikipedia: https://en.wikipedia.org/wiki/MAC_address

ARP Address Resolution Protocol

Wikipedia: https://en.wikipedia.org/wiki/Address_Resolution_Protocol

TCP Transmission Control Protocol

Wikipedia: https://en.wikipedia.org/wiki/Transmission_Control_Protocol

IP Internet Protocol

Wikipedia: https://en.wikipedia.org/wiki/Internet_Protocol

IP Address A *IP* identifier number identifying a particular network entity (e.g. 127.0.0.1)

- We are running out of 32-bit *IPv4* addresses (127.0.0.1), and are now moving toward 128-bit *IPv6* addresses (0000:0000:0000:0000:0000:0000:0000:0001, ::1)

- Certain IP addresses are locally-routable (e.g. 192.168.0.1 within a home LAN) while others are globally-routable (e.g. 8.8.8.8)

Wikipedia: https://en.wikipedia.org/wiki/IP_address

IPv4 IPv4 (*IP* version 4) is a *web standard* protocol defined by *IETF*.

Wikipedia: <https://en.wikipedia.org/wiki/IPv4>

Docs: [IPv4 Address](#)

IPv4 Address *IPv4* addresses are 32-bit *IP Address* identifiers.

Standard: <https://tools.ietf.org/html/rfc1918>

```
## Local IPv4 Addresses
127.0.0.1
10.0.0.0/8
172.16.0.0/12
192.168.0.0/16

## Global IPv4 Addresses
8.8.8.8
```

IPv6 IPv6 (*IP* version 6) is a *web standard* protocol defined by *IETF*.

Wikipedia: <https://en.wikipedia.org/wiki/IPv6>

IPv6 Address *IPv6* addresses are 128-bit *IP Address* identifiers.

Wikipedia: https://en.wikipedia.org/wiki/IPv6_address

Standard: <https://tools.ietf.org/html/rfc4291>

Docs: https://en.wikipedia.org/wiki/IPv6_address#IPv6_address_scopes

Docs:

IPv6 Address Examples:

```
## Local IPv4 Addresses
0000:0000:0000:0000:0000:0000:0000:0001 == ::1 # 127.0.0.1
0001:0000:0000:0000:0000:0000:0000:0001 == 1::1

::/128 # unspecified (~IPv4 0.0.0.0/32)
::1/128 # localhost (~IPv4 127.0.0.0/24)
::/0 # unicast default route (~IPv4 0.0.0.0/0)
::ffff:0:0/96 # IPv6-mapped IPv4
```

```
:::96          # IPV4 compatible IPv6 addresses (deprecated)
2002::/16     # 6to4
2001::/32     # teredo
fc00::/7      # unique local address
fe80::/10 #MAC # link-local address (~IPv4 169.254.0.0/16)
fec0::/10     # site-local address (deprecated)
3ffe::/16     # 6bone (returned)

## Global IPv6 Addresses

# 8.8.8.8
```

DHCP DHCP (*Dynamic Host Configuration Protocol*) is a standard for acquiring an *IP address*, *DNS*, and *NTP* settings.

Wikipedia: https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol

Domain Name A human-readable textual name for a network entity (e.g. `example.org`)

Wikipedia: https://en.wikipedia.org/wiki/Domain_name

DNS Domain Name System. Converts a *domain name* (e.g. `localhost` or `wrdrd.github.io`) into an *IP address* (e.g. `127.0.0.1` (IPv4) or `::1` (IPv6)).

- Initial DNS hosting costs are often covered by Web Hosts.
- There are DNS record types for different types of services.
- Surfing to a website in a browser may utilize A, AAAA, and/or CNAME records to lookup the *IP address* of the web server (or least busy load balancer).
- Sending an email utilizes an MX record to lookup the IP address and sender information for the mail host.
- Updates to DNS settings can take as long as 86400 seconds (one day) to propagate, depending upon the DNS TTL.

Wikipedia: https://en.wikipedia.org/wiki/Domain_Name_System

Docs: *DNS Configuration*

Docs: *wrdrd.tools.domain*

URL A URL (*Uniform Resource Locator*) is a string of characters that identify a resource location.

Where `host` is an IP address, hostname, or domain name, a URL is of the form:

```
scheme://host:port/p/a/t/h

https://wrdrd.github.io/docs/ # https, wrdrd.github.io, port 443, /docs/
```

- With an *http* scheme, the default port is 80.
- With an *https* scheme, the default port is 443.

Wikipedia: https://en.wikipedia.org/wiki/Uniform_resource_locator

URI A URI (*Uniform Resource Identifier*) is a string of characters that identify a resource.

```
scheme://host:port/p/a/t/h?query#fragment
https://wrdrd.github.io/docs/#wrdrd
```

- *URLs* are URIs.
- *URNs* are URIs.

Wikipedia: https://en.wikipedia.org/wiki/Uniform_resource_identifier

URN A URN (*Uniform Resource Name*) is a string of characters that identify a named resource *in a namespace*.

```
urn:namespace:key
urn:isbn:0-486-27557-4
urn:uuid:6e8bc430-9c3a-11d9-9669-0800200c9a66
```

Wikipedia: https://en.wikipedia.org/wiki/Uniform_resource_name

Magnet URI A Magnet *URI* is a *URN* containing an key to retrieve from a network (such as a *DHT*).

- *Web browsers* can be configured to open Magnet URIs with other programs.

Wikipedia: https://en.wikipedia.org/wiki/Magnet_URI_scheme

UUID A UUID (*Universally Unique Identifier*) is a 128- *bit* identifier for a resource.

IETF RFC 4122 defines 5 different algorithms for generating *UUID*:

- **UUID 1** A UUID 1 identifier contains a *mac address* and a *datetime* with 100- *nano-second* resolution.
- **UUID 2** A UUID 2 identifier contains a *mac address* and a POSIX UID or GID.
- **UUID 3** A UUID 3 identifier contains an *MD5* hash of a *URI*, *URN*, or *URL*.
- **UUID 4** A UUID 4 identifier contains a random identifier as determined by the configured source of random.
- **UUID 5** A UUID 5 identifier contains an *SHA-1* hash of a *URI*, *URN*, or *URL*.

Wikipedia: https://en.wikipedia.org/wiki/Universally_unique_identifier

Standard: <https://tools.ietf.org/html/rfc4122>

Web Browser A software program which visually renders resources identified by a URL and interprets scripts.

Examples: Internet Explorer, Mozilla Firefox, Google Chrome

- All web browsers support *HTML* over *HTTP*.
- Many web browsers support *HTTPS* and/or *HTTP STS*.
- Some web browsers support *WebSocket*.
- Some web browsers support *WebRTC*.
- Many web browsers support image formats like *GIF*, *JPEG*, *PNG*, and *SVG* Scalable Vector Graphics.
- Many web browsers support Javascript scripts.
- All web browsers work with a *DOM* (Document Object Model) which is parsed from *HTTP* and transformed by Javascript.
- A number of example web browser extensions: Browser Extensions

Wikipedia: https://en.wikipedia.org/wiki/Web_browser

Docs: Tools > Browsers

Docs: *WebSec*

DOM Document Object Model. Can be thought of as a layout outline of the objects in a particular document (e.g. text, shapes, images, videos).

Different web browsers interpret the DOM differently, depending on Web Standards and individual implementations.

Wikipedia: https://en.wikipedia.org/wiki/Document_Object_Model

Docs: Web Design

Web Standard An agreed-upon standard specification for web things like data interchange, structure, and presentation.

Wikipedia: https://en.wikipedia.org/wiki/Web_standards

Docs: *Web Standards*

Docs: *Semantic Web Standards*

Open Web Standards Open Web Standards are *Open Source Web Standards* (e.g. *HTTP*, *HTML*, *XHTML*, *HTML5*, *CSS*, Javascript, *SVG*, *RDF*)

Web Content

Media Resources: Copy, Text, Photos, Images, Videos (things with *HTTP* URLs)

See *Web Standards*, *Art & Design*

Web Design

Wikipedia: https://en.wikipedia.org/wiki/Web_design

CSS

Wikipedia: https://en.wikipedia.org/wiki/Cascading_Style_Sheets

Docs: [CSS](#)

CSS (*Cascading Style Sheets*) define the presentational aspects of *HTML* and a number of mobile and desktop web frameworks.

- CSS is designed to ensure separation of data and presentation. With javascript, the separation is then data, code, and presentation.

Web Layout

A *web layout* is a box-model composition of *DOM* objects, their styles, and their behaviors at various screen sizes and resolutions.

Different browsers implement the *DOM*, *HTML*, *CSS*, and Javascript differently. It is necessary to test a web layout in the browsers which are utilized by the target audience.

In general, a simpler page renders faster and more consistently.

Some users may be browsing without Javascript (either because their very classic web browser doesn't support it, or, optionally, because of security concerns introduced by active page scripts). Because of this, it's usually best to not rely upon Javascript for page layout and instead work with pure-CSS implementations.

CSS framework developers specialize in developing CSS grids and layouts which work across browsers, devices, and various screen sizes.

- https://en.wikipedia.org/wiki/Page_layout
- https://en.wikipedia.org/wiki/Web_design#Page_layout
- https://en.wikipedia.org/wiki/CSS_frameworks
- https://en.wikipedia.org/wiki/Responsive_Web_Design
- https://en.wikipedia.org/wiki/List_of_displays_by_pixel_density

Image Based Layouts

At first glance, it may seem that an image-based layout with fixed dimensions (as might be developed in a traditional graphic design program) would be simpler and easier; however:

- an 800px wide image layout is hardly usable on a mobile device
- search engines and screen readers are unable to read text embedded within images; necessitating `alt=` attributes on `` tags and `title=` attributes on `<a>` tags
- when scaled (by zooming in), raster images like JPEG, PNG, and GIF look blocky and pixelated

Practically, it is not possible to develop a responsive web layout which supports diverse screen sizes and resolutions with traditional graphic design tools. It is far more consistent and reproducible to start with an HTML web page and a CSS framework and then develop a template from there.

Screen Captures

There are many tools and services for collecting screen captures (or screen shots) of web layouts.

Features to look for:

- Capturing the visible area of the page
- Capturing the whole page
- Setting the browser resolution

Ways to collect screen captures and movies at various points in a testing workflow:

- Browser: Web Browser testing tools (e.g. Javascript)
- Browser: browser extensions
- Web Service: multi-platform browser testing grid services
- Build Script: record the [X] buffer with the test sequence
- Local Recording: *FFmpeg*, *VLC*, CamStudio

See: *Video Production*

Bootstrap

Wikipedia: [https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))

Homepage: <https://getbootstrap.com/>

Source: git <https://github.com/twbs/bootstrap>

- What is Bootstrap?
 - A responsive HTML and CSS (LESS, Sass SCSS (4.0)) Framework.
 - [https://en.wikipedia.org/wiki/LESS_\(stylesheet_language\)](https://en.wikipedia.org/wiki/LESS_(stylesheet_language))
 - [https://en.wikipedia.org/wiki/Sass_\(stylesheet_language\)](https://en.wikipedia.org/wiki/Sass_(stylesheet_language))

- Bootstrap *Web Site* Styles / Themes / Templates

<https://expo.getbootstrap.com/resources/>

- Custom:
 - * <https://getbootstrap.com/customize/#less-variables>
 - * <http://bootply.com/>
- Templates:
 - * <https://themes.getbootstrap.com/> (Updated, Free, Official)
 - <https://themes.getbootstrap.com/pages/our-license>
 - * <https://expo.getbootstrap.com/resources/#themes>
 - * <https://bootswatch.com/> (Free)

- * <http://www.themesforbootstrap.com/>
- * <https://wrapbootstrap.com/>
- * <https://wrapbootstrap.com/theme/deusone-responsive-one-page-template-WB0271X52>
- * <http://themeforest.net/search?utf8=%E2%9C%93&term=bootstrap>
- Examples:
 - * <https://expo.getbootstrap.com/>
 - * <http://builtwithbootstrap.com/>
 - * <http://v4-alpha.getbootstrap.com/examples/>

Web Development Checklist

A checklist for building a modern website with structured data; for search, social web, sharing.

See also: <http://webdevchecklist.com/>

- Pick a CSS framework
- Create page layout template
 - Create or acquire static template
 - * Helps if it already includes a CSS framework
 - Create or acquire dynamic template
- Create static HTML page from layout template
- Port content from existing site
 - Add HTML formatting
 - Add CSS #id deep link anchors and classes
- Add structured data markup to page
 - <http://schema.org/docs/full.html>
 - See: *Knowledge Engineering, Semantic Web Standards, Schema.org*
 - Add standard header tags
 - * meta tags: description
 - * link rel="canonical"
 - * lang="en"
 - Add OpenGraph meta markup
 - * <http://ogp.me/>
 - * og:title
 - * og:type
 - * og:image (:width, :height, :type)
 - * og:url
- Section: Navbar

- [] Choose top-level links
- [] Indicate current location
- [] Section: Above the fold
 - `schema:ImageObject`
 - `schema:VideoObject`
 - `schema:MusicVideoObject`
 - Text
 - HTML/CSS/JS
- [] Add an `<h1>` tag with a page title
- [] Section: About
 - [] Add textual description
 - [] Organization (`schema:Organization`)
 - [] Business (`schema:Organization > schema:LocalBusiness > {...}`)
- [] Section: Products / Services
 - [] Acquire product/menu/service offering information
 - * [] Products (`schema:Product, schema:ProductModel`)
 - * [] Services (`schema:Service < schema:Intangible`)
 - [] Format product/menu/service offering information as HTML + RDFa
- [] Section: Media / In the news
 - [] Research media profile
 - * [] Articles `schema:Article > schema:NewsArticle`
 - [] Acquire news media assets
 - * [] Media Objects (`schema:MediaObject`)
- [] Section: Contact
 - [] Email
 - [] Name, Address, Telephone (`schema:LocalBusiness > schema:Organization > schema:Place`)
 - [] Locations (`schema:LocalBusiness`)
 - * [] Embed map thumbnail/widget
 - * [] Link to Directions
 - `schema:Organization`
 - * name
 - * url
 - * address `<schema:PostalAddress>`
 - * hasMap (map) URL
 - * Directions

- * telephone
- * faxNumber
- * email
- * description
- * logo
- * image
- * sameAs (~= URL)
- * legalName
- * founder
- * foundingDate
- * taxID (TIN)
- * memberOf
- schema:LocalBusiness < schema:Organization
 - * name
 - * url
 - * address <schema:PostalAddress>
 - * hasMap (map) URL
 - * **Directions**
 - * telephone
 - * faxNumber
 - * email
 - * image (s)
 - * branchOf <schema:Organization>
 - * openingHours
 - * currenciesAccepted
 - * paymentAccepted
 - * priceRange
 - * schema:FoodEstablishment < schema:LocalBusiness
 - acceptsReservations Yes/No/URL
 - menu text/URL
 - servesCuisine text
- [] Social Media
 - * [] Google+
 - * [] Twitter
 - * [] Facebook
 - * [] LinkedIn

* [] [...]

- [] Section: Footer
 - [] Copleft: © <year> <business name>
 - [] <location>
 - [] Feedback
 - [] Terms
 - [] Privacy
- [] Section: Post-load JS scripts
 - [] JS libraries (*CDN*, *cdnjs*, *jQuery*, *Bootstrap*, *underscore*, *Backbone*, *Angular*, *React*)
 - [] JS Analytics loaders (*Data Science* > *repro:ObservationalStudy*)
 - [] JS Optimization loaders (*Machine Learning* > *repro:ControlledTrial*)

Hosting / DNS

DNS Configuration

Wikipedia: https://en.wikipedia.org/wiki/Domain_Name_System

Standards: https://en.wikipedia.org/wiki/Domain_Name_System#RFC_documents

Standards: https://en.wikipedia.org/wiki/List_of_DNS_record_types

Standards: https://en.wikipedia.org/wiki/Domain_Name_System_Security_Extensions

DNS Domain Name Information (A, AAAA, CNAME, TXT, MX, SRV)

```
DOMAIN="<domainname>"  
IP=$(nslookup $DOMAIN)
```

- Date of Registration / Expiration
- Registrant (Name, Address, Email)
 - Privacy / WhoisGuard
- DNS Registration Service Provider
- Linux/OSX DNS Commands:

```
nslookup DOMAIN  
dig $DOMAIN  
dig +qr any $DOMAIN  
dig -t mx $DOMAIN  
whois $DOMAIN  
whois $DOMAIN | egrep 'Registrar|Date|Domain Status|Registrant|Admin'
```

- Online Whois Tools
 - <https://whois.domaintools.com/protect\T1\textdollarDOMAIN>

See: *wrdrd.tools.domain*

Web Hosting

Wikipedia: https://en.wikipedia.org/wiki/Web_hosting_service

Web Hosting Information

- Reverse IP (How many sites are hosted from the same *IP address*?)
 - <http://reverseip.domaintools.com/search/?q=\protect\T1\textdollarIP>

See: *Information Systems > Clouds*

WebSec

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Web_security_exploits

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Internet_security

WebSec (*web security*) is *Information Security* for web applications.

Security at W3C :

- <http://www.w3.org/Security/>
- http://www.w3.org/Security/wiki/Main_Page
- <http://www.w3.org/2011/webappsec/>
- *Web Standards*

Responsible Disclosure

Wikipedia: https://en.wikipedia.org/wiki/Responsible_disclosure

- https://www.owasp.org/index.php/Vulnerability_Disclosure_Cheat_Sheet

Bug Bounty Program

Wikipedia: https://en.wikipedia.org/wiki/Bug_bounty_program

- Hacker One:
 - Homepage: <https://hackerone.com/>
 - <https://hackerone.com/twitter>
- Professional integrity:
<https://en.wikipedia.org/wiki/Integrity>
- *Open Source*

SANS/CIS Consensus Audit Guidelines:

Wikipedia: https://en.wikipedia.org/wiki/Consensus_audit_guidelines

Wikipedia: https://en.wikipedia.org/wiki/The_CIS_Critical_Security_Controls_for_Effective_Cyber_Defense

Wikipedia: https://en.wikipedia.org/wiki/CSC_Version_6.0

- “SANS Top 20”: <https://www.sans.org/critical-security-controls/>
- “SANS Top 25”: <https://cwe.mitre.org/top25/>
- CIS:
 - <https://www.cisecurity.org/critical-controls.cfm>
 - <https://www.cisecurity.org/critical-controls/reports/>
 - PDF for email.

CWE

CWE (*Common Weakness Enumeration*)

Homepage: <https://cwe.mitre.org/>

- <https://cwe.mitre.org/top25/>
- <https://cwe.mitre.org/top25/#CWE-89>
- <https://cwe.mitre.org/data/definitions/89.html>

OWASP

Homepage: <https://www.owasp.org/>

- https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
- https://www.owasp.org/index.php/OWASP_Proactive_Controls
- https://www.owasp.org/index.php/Top_10_2013-Table_of_Contents
- https://www.owasp.org/index.php/Web_Standards_and_Specifications

WebSec Forums

Internet Forums for WebSec (“Web Security”)

- *Responsible Disclosure* / Vulnerability Disclosure
- <https://www.cvedetails.com/>
- IRC, Slack, Mattermost
- <https://www.reddit.com/r/websec>
- <https://www.reddit.com/r/netsec>

- <https://twitter.com/search?q=websec>
- <https://twitter.com/search?q=netsec>
- <https://twitter.com/search?q=netsec>
- <https://twitter.com/search?q=infosec>

HTTP

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Hypertext_Transfer_Protocol

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Hypertext_Transfer_Protocol_headers

See:

- Knowledge Engineering > *HTTP*
- *HTTPS, HTTPS*

HTTPS

Wikipedia: <https://en.wikipedia.org/wiki/HTTPS>

See:

- Knowledge Engineering > *HTTPS*
- *HTTP, HTTP*

HTTP STS

Wikipedia: https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security

HTTP STS (*HTTP Strict Transport Security*) is a standardized extension for notifying browsers that all requests should be made over *HTTPS* indefinitely or for a specified time period.

See also:

- Knowledge Engineering > *HTTP STS*
- [https everywhere](#)

ACME Protocol

Wikipedia: https://en.wikipedia.org/wiki/Automated_Certificate_Management_Environment

Src: <https://github.com/ietf-wg-acme/acme/>

Spec: <https://tools.ietf.org/html/draft-ietf-acme-acme>

Spec: <https://ietf-wg-acme.github.io/acme/draft-ietf-acme-acme.html>

- ACME (Automated Certificate Management Environment) is a protocol for Web PKI with X.509 Certificates and *HTTPS* (TLS/SL).

- ACME Spec Introduction: <https://tools.ietf.org/html/draft-ietf-acme-acme#section-1>
- *Letsencrypt* implements the ACME spec.

Letsencrypt

Wikipedia: https://en.wikipedia.org/wiki/Let%27s_Encrypt

Homepage: <https://letsencrypt.org/>

Src: <https://github.com/letsencrypt>

Docs: <https://letsencrypt.readthedocs.io/en/latest/>

Docs: <https://letsencrypt.readthedocs.io/en/latest/using.html#getting-certificates-and-choosing-plugins>

Docs: <https://letsencrypt.readthedocs.io/en/latest/using.html#third-party-plugins>

- “A free, automated, and open certificate authority.”
- Certs are valid for 90 days
- Use the staging server to avoid the certificate roll rate limit
- There are many ACME clients (that work with letsencrypt):
<https://letsencrypt.org/docs/client-options/>
 - Certbot is an official, packaged Python client.

Web Development Requirements

- [] *Software Development > Project Management*
- [] *Team Building > Collaboration Checklist*
- [] *Collaboration Plan*
- [] *Photography Checklist*
- [] *Social Media Images*
- [] *Web Hosting*
- [] *Web Development Checklist*

User Stories

Wikipedia: https://en.wikipedia.org/wiki/User_story

User Stories are an *Agile technique* for capturing structured requires on *cards* (or as *issues* in e.g. GitHub).

- User Stories are, in general, less complex than *Use Cases* which are often more highly specified (in terms of e.g. UML diagrams).
- User Stories can be grouped in **epics**. An **epic** story is a long and arduous journey; often with multiple parts.
- User Stories can be *estimated* and assigned arbitrary but relatively relevant point values with e.g. *Planning Poker* and/or a *Kanban* web application.

See also:

- *Software Development > Agile > User Story*
- *Team Building > Six Patterns of Collaboration*

https://en.wikipedia.org/wiki/Software_development

https://en.wikipedia.org/wiki/Software_development_process

https://en.wikipedia.org/wiki/Category:Software_design_patterns

- [https://en.wikipedia.org/wiki/Architectural_pattern_\(computer_science\)](https://en.wikipedia.org/wiki/Architectural_pattern_(computer_science))
- <http://martinfowler.com/eaaCatalog/>

https://en.wikipedia.org/wiki/Software_engineering

Computer Science

https://en.wikipedia.org/wiki/Computer_science#Applied_computer_science

https://en.wikipedia.org/wiki/Outline_of_computer_science

https://en.wikipedia.org/wiki/Category:Computer_science

Topics:

- [https://en.wikipedia.org/wiki/Abstraction_\(computer_science\)](https://en.wikipedia.org/wiki/Abstraction_(computer_science))
- <https://en.wikipedia.org/wiki/Category:Algorithms>
- *Computer Engineering*
- *Data Structures*
- *Operating Systems*
- *Software Development (applied CS)*
- *Data Science (applied CS)*
- *Knowledge Engineering (applied CS)*
- *Science (applied CS with/for/on behalf of a domain)*

Computer Science Curricula

Curricula

- <https://en.wikipedia.org/wiki/Curriculum>
- https://en.wikipedia.org/wiki/Open-source_curriculum
- https://en.wikipedia.org/wiki/Open-source_curriculum#Resources

Criteria

- <https://en.wikipedia.org/wiki/Criterion>
- [https://en.wikipedia.org/wiki/Criterion_\(disambiguation\)#Science_and_mathematics](https://en.wikipedia.org/wiki/Criterion_(disambiguation)#Science_and_mathematics)
- https://en.wikipedia.org/wiki/Information_criterion
- “Common Criteria” – *Information Security* platform specification
- “Common Core” – US education standard: https://en.wikipedia.org/wiki/Common_Core_State_Standards_Initiative
- *ROI, ROI*

Rubric

- [https://en.wikipedia.org/wiki/Rubric_\(academic\)](https://en.wikipedia.org/wiki/Rubric_(academic))
- A rubric can be expressed with a *matrix data structure*.
- A rubric can be expressed as an *RDF graph* in e.g. *RDFa* or *JSON-LD*.

Computer Science criteria, guidelines, and curricula:

- ACM Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science (December 20, 2013)

<https://www.acm.org/education/CS2013-final-report.pdf>

- AL – Algorithms and Complexity (p.55)
- AR – Architecture and Organization (p.62)
- CN – Computational Science (p.68)
- DS – Discrete Structures (p.76)
- GV – Graphics and Visualization (p.82)
- HCI – Human-Computer Interaction (p.89)
- IAS – Information Assurance and Security (p.97)
- IM – Information Management (p.112)
- IS – Intelligent Systems (p.121)
- NC – Networking and Communication (p.130)
- OS – Operating Systems (p.135)
- PBD – Platform-Based Development (p.142)
- PL – Programming Languages (p.155)
- SDF – Software Development Fundamentals (p.167)
- SE – Software Engineering (p.172)

- SF – Systems Fundamentals (p.186)
- SP – Social Issues and Professional Practice (p.192)
- ACM Computer Science Curriculum 2008: An Interim Revision of CS 2001
<http://www.acm.org/education/curricula/ComputerScience2008.pdf>
- Programmer Competency Matrix
<http://sijinjoseph.com/programmer-competency-matrix/>
<https://competency-checklist.appspot.com/>
<https://github.com/hltbra/programmer-competency-checklist>
 - computer science
 - * data structures
 - * algorithms
 - * systems programming
 - software engineering
 - * source code and version control
 - * build automation
 - * automated testing
 - programming
 - * problem decomposition
 - * systems decomposition
 - * communication
 - * code organization within a file
 - * source tree organization
 - * code readability
 - * defensive coding
 - * error handling
 - * IDE
 - * API
 - * frameworks
 - * requirements
 - * scripting
 - * database
 - knowledge
 - * tool knowledge
 - * languages exposed to
 - * codebase knowledge
 - * knowledge of upcoming technologies

- * platform internals
- * books
- * blogs
- experience
 - * languages with professional experience
 - * platforms with professional experience
 - * years of professional experience
 - * domain knowledge

Computer Science Courses

- *Education Technology > Online Courses*
- <https://www.khanacademy.org/cs>
- <https://www.class-central.com/subject/cs>

Computer Science Exercises

Exercises, Problems

- http://rosettacode.org/wiki/Category:Programming_Tasks
- *Project Euler*
- *Rosalind*

Code Golf

Wikipedia: https://en.wikipedia.org/wiki/Code_golf

- https://en.wikipedia.org/wiki/Code_golf#Tournaments_by_language

Project Management

Wikipedia: https://en.wikipedia.org/wiki/Project_management

- <https://en.wikipedia.org/wiki/DMAIC>
- <https://en.wikipedia.org/wiki/Kaizen>
- https://en.wikipedia.org/wiki/Systems_development_life_cycle
- https://en.wikipedia.org/wiki/Software_project_management
- https://en.wikipedia.org/wiki/Software_development_process

Waterfall

Wikipedia: https://en.wikipedia.org/wiki/Waterfall_model

Everyone Codes

Q. Everyone? A. Everyone but

- <https://code.org/>

Agile

Wikipedia: https://en.wikipedia.org/wiki/Agile_software_development

- https://en.wikipedia.org/wiki/Lean_software_development
- <http://epf.eclipse.org/wikis/epfpractices/>
- <http://agilemodeling.com/essays/bestPractices.htm>

Stand Up Meeting

Wikipedia: https://en.wikipedia.org/wiki/Stand-up_meeting

- [https://en.wikipedia.org/wiki/Scrum_\(software_development\)#Daily_scrum_meeting](https://en.wikipedia.org/wiki/Scrum_(software_development)#Daily_scrum_meeting)

Note: Meeting participants need not actually stand; as in a *Digital Stand Up Meeting*.

Three Questions

Wikipedia: https://en.wikipedia.org/wiki/Stand-up_meeting#Three_Questions

1. What did I accomplish yesterday?
2. What will I do today?
3. What obstacles are impeding my progress?

Digital Stand Up Meeting

A Digital *Stand Up Meeting* is a meeting conducted through digital means of communication; rather than through traditional in-person communication.

Pros:

- **Links:** Realtime collaborative documents (*live docs*), team chat logs, and issue/ticket comment threads have URIs, are easier to remember, and more to the point.
- Easier to prepare for

- More time to read, think, and respond
- More comfortable
- Easier to get up to date with

Cons:

- It's not everyone in the same room
- “Shouldn't this all be in the issue tracker?”
 - Mailing Lists, Forums, *SIOC* <- Links to source, docs, issues
 - Bugs, User Stories -> Issue Tracker
 - Daily Stand Up Meeting log / **minutes** ->
 - *

Digital Stand Up Meeting Process Checklist:

- Create a collaboratively edited document (“live doc”, “doc”)
 - Realtime collaborative editors: Etherpad Lite, Google Docs
 - An example of *Three Questions* for stand up meetings (in Markdown, with *URIs*, for GitHub, BitBucket):

```
# team-name/project-name Daily Stand Up Log

## 2015-01-01

### @username1
1. yesterday: #1, #2, http://
2. today: #2
3. obstacles: -

### @username2
1. yesterday: #3, #2, https://
2. today: #2
3. obstacles: -
```

- Share the link to the live doc with the team
 - Email, Text Chat
 - Meeting/event invitations (iCal)
- Chat (Text, Voice, Video)
 - Text
 - * Slack
 - * Mattermost
 - * HipChat
 - * Gitter
 - * XMPP
 - * IRC
 - Voice/Video (cross-platform): Google Hangouts, Skype, Cisco WebEx, *WebRTC*
 - * Live streaming

- YouTube
- Twitter
- Facebook

See: *Team Building (The same page, Collaborative Software)*

Extreme Programming

Wikipedia: https://en.wikipedia.org/wiki/Extreme_programming

EPFWiki: <http://epf.eclipse.org/wikis/xp/>

Unified Process

Wikipedia: https://en.wikipedia.org/wiki/Unified_Process

EPFWiki: <http://epf.eclipse.org/wikis/openup/>

- “Rational Unified Process” (RUP)
- OpenUP is an *Open Source* Unified Process

Scrum

Wikipedia: [https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development))

EPFWiki: <http://epf.eclipse.org/wikis/scrum/>

- The Daily Scrum (EPF Wiki Entry)
http://epf.eclipse.org/wikis/scrum/Scrum/tasks/the_daily_scrum_C8FB7908.html

Effort Estimation

Wikipedia: https://en.wikipedia.org/wiki/Software_development_effort_estimation

Planning Poker

Wikipedia: https://en.wikipedia.org/wiki/Planning_poker

Kanban

Wikipedia: <https://en.wikipedia.org/wiki/Kanban>

Wikipedia: [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development))

- GitHub and Kanban:

- <https://waffle.io>
- <https://taiga.io> (<https://github.com/taigaio>)

Requirements Management

Wikipedia: https://en.wikipedia.org/wiki/Requirements_management

Requirements Modeling

Wikipedia: https://en.wikipedia.org/wiki/Requirements_analysis

Agile Modeling

Wikipedia: https://en.wikipedia.org/wiki/Agile_modeling

Docs:

- “Agile Modeling Best Practices”
<http://agilemodeling.com/essays/bestPractices.htm>
 - **Model storming:** <http://agilemodeling.com/essays/modelStorming.htm>
 - JIT: Just-in-time
 - BRUF: “Big Requirements Up Front”
 - JBGE: “Just Barely Good Enough” (*/anticipation*)
 - **Executable Specifications:** <http://agilemodeling.com/essays/executableSpecifications.htm>
 - * TST: *Test Driven Development* and *Behavior Driven Development* tests (*Test Automation*)
 - * BLD: Software Packaging
 - * BLD: Configuration Management (*Infrastructure as Code*)
 - * BLD: *Continuous Integration (one-click build)*
 - * BLD: *Continuous Deployment (one-click deploy)*

See also:

- *Scrum*, Extreme Programming, Unified Process
- *Collaborative Software*
- *Six Patterns of Collaboration*

Test Driven Development

Wikipedia https://en.wikipedia.org/wiki/Test-driven_development

- <https://westurner.github.io/wiki/awesome-python-testing>
- <https://westurner.github.io/wiki/awesome-python-testing#workflow-evolution>
- <https://westurner.github.io/wiki/awesome-python-testing#test-driven-development-tdd>

Behavior Driven Development

Wikipedia: https://en.wikipedia.org/wiki/Behavior-driven_development

- <https://westurner.github.io/wiki/awesome-python-testing#behavior-driven-development-bdd>

Given-When-Then

Wikipedia: <https://en.wikipedia.org/wiki/Given-When-Then>

- Given (context)
- When (event)
- Then (expect, should, assert)

Given When Then	Given	When	Then
<i>Hoare Logic</i>	precondition P	command C	postcondition Q
Terms	Input: State, Context (,RDF), Database	Input: Parametrized Event, Message	Output Conditions, Rules, Expressions, Assertions, Expectations
HTTP 1.1	app state (etc, fs, db)	request (URI(URN URL))?attrs=values	response ({"json": True})
Channels	app state (etc, fs, db)	event(message_in)	conditions(message_out)
Vows	test suite, @Vows.Batch, Vows.Context, topic ()	vow - def should_quack (self, topic) :	vow - “ expect(topic.quack()).to_equal(value)“

User Story

Wikipedia: https://en.wikipedia.org/wiki/User_story

A user story is a couple of simple sentences expressing desired competencies, features, or behaviors of a system.

Ideally, user stories can simply be transformed to e.g. *TDD* and *BDD* tests.

Example user stories:

- “[role] can [action [objects]] [in order to / so that [...]]”

- “Users can login and post Things with a description and a url”
- “Users can login and post Things with a description and a url in order to have something with a URL to share”
- “Users can share shared Things with a URL”
- User story formats

https://en.wikipedia.org/wiki/User_story#Format

Card, Conversation, Confirmation

Wikipedia: https://en.wikipedia.org/wiki/User_story#History

Docs: <http://ronjeffries.com/xprog/articles/expcardconversationconfirmation/>

The Three C’s of *User Story Requirements Modeling* are **Card**, **Conversation**, and **Confirmation**.

- Card
 - Issue name and description (or `description__markdown_github`).
 - *Codelabels*
- Conversation
 - Issue comment thread or **threads** (with full URLs)
 - Pull Request comment thread or **threads** (with full URLs)
 - *Codelabels*
- Confirmation
 - Issue `status`: open / closed
 - Issue Labels: { duplicate, wontfix }
 - Issue labels: { ready, in progress, } TODO
 - * <https://waffle.io/> uses ready, in progress labels to build an interactive *Kanban* board with columns from github issue labels.

Use Case

Wikipedia: https://en.wikipedia.org/wiki/Use_case

A use case describes actions, actors, and goals.

- UML defines a formal model for *Use cases* and use case diagrams.
 - Use case diagrams may involve **stick figures** as symbols for **actors**.
- In practice, the use case is a more formal requirement model than the *User Story*.
 - In terms of process workflow, a *Use Cases* could be derived from zero or more *User Stories*.
 - There are many domain modeling tools for e.g. UML; each with various levels of support for *round-trip* between modeling tool and e.g. code that you add after transforming to which is sometimes referred to as *stub* code.

Requirements Traceability

Wikipedia: https://en.wikipedia.org/wiki/Requirements_traceability

- *URLs*
 - issue numbers (GitHub, BitBucket,):
 - * ad-hoc (the numbers probably won't be sequential)
 - * issue #1: Mission, Project list
 - * issue #2: Project
 - * issue #3: Goal
 - * issue #4: an Objective for goal #3
 - With a fixed, preallocated numbering system, this would be e.g. `project:3.1`; however, issue ticket numbers are sequential by order of creation
 - the numbers would then appear somewhat arbitrary
 - there would be no need to renumber things
 - #perma-link document *URL* fragments
 - * #mission-statement
 - * #project-<name> / #p-<name>
 - * #goal-<name> / #g-<name>
 - * #objective-<name> / #o-<name>
 - * implementations:
 - sphinx restructuredtext (with Ctrl-F'able explicit links):
 - <https://docs.python.org/devguide/documenting.html#sections>

```

.. _mission:

#####
Mission
#####

.. _project-name:

*****
Project: Name
*****

.. _goal-name:

Goal: Name
=====

.. _objective-name:

Objective: Name
-----

```

```
* [ ] Task: Name
.. _task-name:
Task: Name
^^^^^^^^^^^^^^
```

- **markdown:**

```
#
Mission:

## Project: Name

## Goal: Name

### Objective: Name

* [ ] Task: Name

#### Task: Name
```

- see also: #pyglobalgoals

- * <https://github.com/schemaorg/schemaorg/issues/1127>

- [] schema:Project
 - [] schema:Goal
 - [] schema:Objective
 - [] schema:Task

- **Codelabels:**

```
MISSION
GOAL
OBJECTIVE

IDEA: idea

STRY: user story -> TST, ENH, BUG, RLS ; -> CASE ; + REQ
CASE: use case -> TST, ENH, BUG, RLS ; -> STRY ; + REQ
REQ: requirement -> TST ; + REQ

SPRINT: sprint == { REQ, CASE, STRY, TST, BUG, ENH, DOC, TST, RLS }

TASK
```

Work Breakdown Structure

Wikipedia: https://en.wikipedia.org/wiki/Work_breakdown_structure

Mission

Wikipedia: https://en.wikipedia.org/wiki/Mission_statement

Project

Wikipedia: <https://en.wikipedia.org/wiki/Project>

- [https://en.wikipedia.org/wiki/Scope_\(project_management\)](https://en.wikipedia.org/wiki/Scope_(project_management))

Goal

Wikipedia: <https://en.wikipedia.org/wiki/Goal>

- https://en.wikipedia.org/wiki/Goal_setting

SMART

Wikipedia: https://en.wikipedia.org/wiki/SMART_criteria

- Specific
- Measurable
- Attainable
- Relevant
- Time-bound

Objective

Wikipedia: <https://en.wikipedia.org/wiki/OKR>

- https://en.wikipedia.org/wiki/Management_by_objectives
- *SMART*

Checkboxes

Wikipedia: <https://en.wikipedia.org/wiki/Checkbox>

Checkboxes can be expressed in plaintext with square brackets surrounding a space character, a state character, or a completion character (e.g. lowercase x or uppercase X).

A Markdown list of items with checkboxes and *Codelabels*:

```
- [x] ENH: one one
- [ ] ENH: two two
  - [x] TST: two.one
  - [ ] DOC: two.two \* two
- [ ] RLS: ``three`` (also `three`)
```

A ReStructuredText list of items with checkboxes and *Codelabels*:

```
* [x] ENH: one one
* [ ] ENH: two two

* [x] TST: two.one
* [ ] DOC: two.two \* two

* [ ] RLS: ``three``
```

TODO

```
* [ ] Unchecked # github
* [x] Checked   # github
* [X] Checked   # github

- [X (YYYY-MM-DD HH:MM:SS)]
- [-]
- [o]
- [O]

- [[ ]]           # text
- [[x]]          # text
- [[o]]          # text
- \\[ ]          # text
- \\[-]          # text
- \\[o]          # text
- \\[x]          # text
- \\[X]          # text
```

Codelabels

Codelabels (*code labels*) are three-letter codes with which commit messages can be prefixed.

CODE	Label	color name	background	text
BLD	build	light green	<i>#bfe5bf</i>	<i>#2a332a</i>
BUG	bug	red	<i>#fc2929</i>	<i>#ffffff</i> (<i>github default</i>)
CLN	cleanup	light yellow	<i>#fef2c0</i>	<i>#333026</i>
DOC	documentation	light blue	<i>#c7def8</i>	<i>#282d33</i>
ENH	enhancement	blue	<i>#84b6eb</i>	<i>#1c2733</i> (<i>github default</i>)
ETC	config			
PRF	performance	deep purple	<i>#5319e7</i>	<i>#ffffff</i>
REF	refactor	dark green	<i>#009800</i>	<i>#ffffff</i>
RLS	release	dark blue	<i>#0052cc</i>	<i>#ffffff</i>
SEC	security	orange	<i>#eb6420</i>	<i>#ffffff</i>
TST	test	light purple	<i>#d4c5f9</i>	<i>#2b2833</i>
UBY	usability	light pink	<i>#f7c6c7</i>	<i>#332829</i>
DAT	data			

```

SCH  schema

REQ  requirement
ANN  announcement

# Workflow Labels (e.g. for waffle.io kanban board columns)
ready      dark sea green #006b75    #ffffff
in progress yellow        #fbca04    #332900

# GitHub Labels
duplicate  darker gray    #cccccc    #333333 (github default)
help wanted green          #159818    #ffffff (github default)
invalid    light gray    #e6e6e6    #333333 (github default)
question   fuschia       #cc317c    #ffffff (github default)
wontfix    white        #ffffff    #333333 (github default)

Note: All of these color codes (except for fuschia)
are drawn from the default GitHub palette.

Note: There are 23 labels listed here.

```

Note: For examples with color swatches in alphabetical order, see <https://github.com/westurner/dotfiles/labels>

Codelabel Syntax

```

BLD: Makefile: default -> test
BUG: setup.py: fix console_entrypoints
CLN: .gitignore: add "'pattern'
DOC: index.rst: add readme.rst to TOC
ENH: app/app.py: print 'Hello World' (closes #1)
PRF,REF: app/app.py: factor out of a loop
REF: app/app.py: extract into class and @staticmethods
RLS: setup.py, VERSION.txt, app/app.py: app v0.0.1
SEC: app/app.py: pass lists/tuples to sarge.run (#2)
TST: Extend tests for #1
UBY: app/app.py: update _('strings')

DAT: data/filename.csvw.jsonld: added, updated
SCH: data/context.jsonld: added, updated

```

- COMMA, DELIMITED, SET: of prefix labels
- THR - Three-characters (code labels)

Code labels are helpful for:

- DOC: working with issue tracking tagging and labeling systems
- DOC: aggregating changes into release logs (HISTORY)
- DOC: correlating changes with requirements (https://en.wikipedia.org/wiki/Requirements_traceability) [*Requirements Traceability*]

Note: These code labels are adapted and extended from a previous version of the `pandas/CONTRIBUTING.md`

Technical Debt

Wikipedia: https://en.wikipedia.org/wiki/Technical_debt

- https://en.wikipedia.org/wiki/Technical_debt#Causes
- *Refactoring* can reduce *Technical Debt*.

Refactoring

Wikipedia: https://en.wikipedia.org/wiki/Code_refactoring

- *Test Driven Development* and *Continuous Integration* are key to “Fearless Refactoring”.
- *Refactoring* can reduce *Technical Debt*.
- While syntactical eloquence and/or elegance in one or more languages can be learned through *Code Golf*, practical *Refactoring* is more about structural, architectural decisions which anticipate change.

Continuous Delivery

Wikipedia: https://en.wikipedia.org/wiki/Continuous_delivery

- https://en.wikipedia.org/wiki/Application_lifecycle_management
- US Digital Services Playbook
<https://playbook.cio.gov/>
 - <https://westurner.github.io/opengov/us/us-digital-services-playbook>

Revision Control

Wikipedia: https://en.wikipedia.org/wiki/Revision_control

Docs: Tools > Version Control Systems

Revision control, or *version control*, is a change management strategy for a project repository.

- There is a better way than emailing which version of what the attachment was named before the re: subject line changed.
 - folders
 - file versions
 - usernames, change logs, signatures
- A *Revision Control System* (RCS) or *Version Control System* (VCS) is a software solution for handling change management with one or more project repositories.
- Centralized *Revision Control* Systems (RCS, VCS):
 - CVS, SVN

- *Distributed Revision Control* Systems (DVCS):
 - Git, Mercurial, Bazaar

Distributed Version Control

Wikipedia: https://en.wikipedia.org/wiki/Distributed_revision_control

Distributed Revision Control Systems (“DVCS”) are version control systems designed for backup redundancy and collaboration.

- <http://betterexplained.com/articles/intro-to-distributed-version-control-illustrated/>
- Git, Mercurial, Bazaar
- vcs (Git, HubFlow, Mercurial)

Bisection

Wikipedia: [https://en.wikipedia.org/wiki/Bisection_\(software_engineering\)](https://en.wikipedia.org/wiki/Bisection_(software_engineering))

Bisection is a method for determining which change causes a fault (or a specific test to change from passing to failing or vice-versa).

Many bisection algorithms take a start and end (“between here and here”) and do a binary search (“this half or that half”); checking out each revision and running a script that should return 0 for OK, or non-zero.

Code bisection with git:

- <https://www.kernel.org/pub/software/scm/git/docs/git-bisect.html>
- <https://www.kernel.org/pub/software/scm/git/docs/git-blame.html>
- <http://git-scm.com/book/en/Git-Tools-Debugging-with-Git>

Code bisection with mercurial:

- <http://www.selenic.com/mercurial/hg.1.html#bisect>
- <http://www.selenic.com/mercurial/hg.1.html#annotate>
- <http://hgbook.red-bean.com/read/finding-and-fixing-mistakes.html#sec:undo:bisect>

Branching & Merging

Wikipedia: [https://en.wikipedia.org/wiki/Branching_\(revision_control\)](https://en.wikipedia.org/wiki/Branching_(revision_control))

Wikipedia: [https://en.wikipedia.org/wiki/Merge_\(revision_control\)](https://en.wikipedia.org/wiki/Merge_(revision_control))

- <http://www.infoq.com/articles/agile-version-control>
- HubFlow – a Git release workflow for GitHub and *Pull Requests* with always-stable (“firm”) master and develop branches, and hotfix/, feature/, and release/ branches; consistently woven.

Test Automation

Wikipedia: https://en.wikipedia.org/wiki/Test_automation

- *Test Driven Development* – write tests for bugs, use cases, user stories, features, workflows, changes.
- *Continuous Integration* – always run the tests when checking code into *Revision Control* and *before releasing*.
- *Continuous Delivery* – test the application delivery and deployment (script a complete system install e.g. with virtualization like Docker).

Continuous Integration

Wikipedia: https://en.wikipedia.org/wiki/Continuous_integration

- *Test Automation*
- Configuration Management
- #DevOps

Continuous Deployment

Wikipedia: https://en.wikipedia.org/wiki/Continuous_deployment

- Configuration Management
- Virtualization
- #DevOps

Computer Engineering

Wikipedia: https://en.wikipedia.org/wiki/Computer_engineering

Computer engineering combines, traditionally, electrical engineering and *Computer Science*.

- Computers are made of components with inputs, outputs, and logic tables.
- Computer programs may or may not be verified using formal methods.
- https://en.wikipedia.org/wiki/List_of_emerging_technologies#Electronics
- https://en.wikipedia.org/wiki/List_of_emerging_technologies#IT_and_communications

Monitor

- Serial
- CRT
- LCD
- OLED
- Quantum Dots
- [] list of emerging technologies # ~display

todo: grep for todo,fixme,[]

Northbridge

Wikipedia: [https://en.wikipedia.org/wiki/Northbridge_\(computing\)](https://en.wikipedia.org/wiki/Northbridge_(computing))

- Many/most desktops, laptops, and servers have a northbridge which connects many core system components.
- A *CPU* is connected to a northbridge.
- A *CPU* may be connected to a northbridge through a motherboard *processor socket*.
- A northbridge connects to *RAM*.
- A northbridge connects to a *System Bus* (e.g. *PCI-e*)
- A southbridge connects to a *Northbridge*.

Processor

CPU

Wikipedia: https://en.wikipedia.org/wiki/Central_processing_unit

A CPU (*Central Processing Unit*) — or **processor** — does variable logic according to a set of gates and input instructions.

- A computer has one or more CPUs.
- A CPU has one or more cores.
- A CPU has one or more internal (*L1, on-die*) or external (*L2*) caches.
- A CPU cache buffers addressed memory for CPU instructions.
- A CPU supports an instruction set.
- A CPU may be programmable with microcode.
- A CPU evaluates a stream of instructions (e.g. a *Turing Machine*).

ARM

Wikipedia: https://en.wikipedia.org/wiki/ARM_architecture

- Routers
- Smartphones
- Notebooks
 - Chromebooks
- Routers
- Servers

PPC

Wikipedia: <https://en.wikipedia.org/wiki/PowerPC>

- https://en.wikipedia.org/wiki/PowerPC#Operating_systems
- https://en.wikipedia.org/wiki/Apple's_transition_to_Intel_processors (2006)

x86

<https://en.wikipedia.org/wiki/X86>

x86 is 32-bit *CPU* architecture.

- A **PC** by a number of definitions
 - A PC is a computer with a *CPU* that runs *x86* or *x86_64* instructions.
 - A PC is a computer running windows
 - * e.g. “Mac or PC? PC or Mac?”; where Mac = OSX on *x86_64* and PC = Windows on *x86* or *x86_64*; and Linux runs on anything.

x86_64

Wikipedia: <https://en.wikipedia.org/wiki/X86-64>

x86_64 (*x64*, *AMD64*) is a 64-bit *CPU* architecture.

- amd64 is *x86_64*.
- In 2015, most new computers are either *x86_64* or *ARM*.

Memory

Wikipedia: https://en.wikipedia.org/wiki/Computer_memory

- *RAM* – live working area (“desktop”, “workspace”)
- *Persistent Storage* – Hard Drive, CD/DVD, USB drive, SSD (“file cabinet”)

RAM

Wikipedia: https://en.wikipedia.org/wiki/Random-access_memory

RAM (*Random Access Memory*) is a category of volatile storage technologies which require voltage to remain applied in order to maintain state.

- RAM is hundreds of times faster than many/most other *Persistent Storage* methods.
- It takes seconds for the voltage from RAM to discharge.
- A *cold boot* or *cold reboot* is when the RAM gets a few seconds (sometimes 30 or more) to discharge. (“**the magic touch**”)

System Bus

[https://en.wikipedia.org/wiki/Bus_\(computing\)](https://en.wikipedia.org/wiki/Bus_(computing))

ISA

Wikipedia: https://en.wikipedia.org/wiki/Industry_Standard_Architecture

PCI

Wikipedia: https://en.wikipedia.org/wiki/Conventional_PCI

PCMCIA

Wikipedia: https://en.wikipedia.org/wiki/PC_Card

PCMCIA is a standard for smaller-form-factor expansion cards.

- PCMCIA is now known as “PC Card” and “CardBus”.
- Some laptops have PCMCIA slots.
- There are PCI to PCMCIA adapter cards.

AGP

Wikipedia: https://en.wikipedia.org/wiki/Accelerated_Graphics_Port

ExpressCard

Wikipedia: <https://en.wikipedia.org/wiki/ExpressCard>

- Some laptops have ExpressCard slots.
- ExpressCard supersedes the *PCMCIA* (PC Card, CardBus) standards for smaller-form-factor expansion cards.

PCI-e

Wikipedia: https://en.wikipedia.org/wiki/PCI_Express

Video Card

Wikipedia: https://en.wikipedia.org/wiki/Video_card

A video card connects a *System Bus* with a monitor through one or more display connectors and does computer graphics processing.

- A computer may have zero or more video cards.

- A video card contains a *GPU*.
- Many/most video cards connect to the *System Bus* (e.g. *AGP*, *PCI-e*).
- Some video cards connect over *USB* (e.g. for adding monitors to a notebook).

Multiheaded Display

Docs: <https://wiki.archlinux.org/index.php/Multihead>

Docs: <https://wiki.ubuntu.com/X/Config/Multihead/>

Q: Why would I need more than one monitor?

A: To multitask.

A: For presentations. For presentations with multiple projectors and something like *Resolume*.

Per *Video Card* multi-headed display configurations:

- Synchronized, Mirroring — Same output on multiple monitors
- Teamed, Seamed, Tiled — One framebuffer which spans multiple monitors
- Multi-seat — separate display/keyboard/mouse (“multi-user console server”, *thin client server*, LTSP)

Challenges

- Window placement
 - Split screen
 - Tabbed
 - Stacked
 - Floating
- Keyboard shortcuts
 - move window_x to monitor 2
 - move window_x and ancillary floating windows to desktop 6
 - set monitor 2 to desktop 6
 - move this whole workspace to the monitor on the side
 - show window_x in fullscreen across other windows
 - toggle floating and always-on-top for window_x

See: i3wm

GPU

Wikipedia: https://en.wikipedia.org/wiki/Graphics_processing_unit

A GPU (*Graphics Processing Unit*) does variable logic according to a set of gates and input instructions; with massive parallelism.

- A computer has zero or more GPUs.
- A GPU may have internal *RAM*; or may use a portion of the system ram (as with many notebooks)

- A GPU is designed for parallel computation.
- GPUs were designed, in particular, for realtime *3D Modeling* (vertex pixel shading, physics)
- GPUs have driven demand for faster *system bus standards*.
- GPU-accelerated code is code that has been adapted for or just runs faster on a GPU.
- Many GPUs support GPU-accelerated *OpenGL*.
- Some GPUs support *CUDA*.
- Some GPUs support *PhysX*.

CUDA

Wikipedia: <https://en.wikipedia.org/wiki/CUDA>

Homepage: http://www.nvidia.com/object/cuda_home_new.html

CUDA (*Compute Unified Device Architecture*) is an API for *GPUs*.

CUDA-accelerated libraries for *Data Science*, *Machine Learning*, *Deep Learning*:

- Pylearn2 (Theano), Numba
- Torch
- cuBLAS (BLAS + CUDA = faster than Intel MKL)

<https://developer.nvidia.com/cuBLAS>

PhysX

Wikipedia: <https://en.wikipedia.org/wiki/PhysX>

Homepage: <https://developer.nvidia.com/gameworks-physx-overview>

PhysX is a realtime physics engine for *GPUs* by Nvidia.

VGA

Wikipedia: https://en.wikipedia.org/wiki/Video_graphics_array

Wikipedia: https://en.wikipedia.org/wiki/VGA_connector

VGA (*Video Graphics Array*) is a video display interface.

- VGA connectors are often *blue*.
- VGA connectors are 15-pin and trapezoidal
- VGA predates *DVI* and *HDMI*
- There are *VGA* to *DVI* and/or *HDMI* adapters.

DVI

Wikipedia: https://en.wikipedia.org/wiki/Digital_Visual_Interface

Wikipedia: https://en.wikipedia.org/wiki/Digital_Visual_Interface#Connector

DVI is a video display interface.

- DVI connectors are often *white*.
- There are a number of different DVI connectors; as well as Mini-DVI and Micro-DVI connectors.

HDMI

Wikipedia: <https://en.wikipedia.org/wiki/HDMI>

Wikipedia: <https://en.wikipedia.org/wiki/HDMI#Connectors>

HDMI (*High-Definition Multimedia Interface*) is an audio/visual interface standard.

- HDMI cables carry audio and video over the same cable.
- There are 5 types of HDMI connectors:
 - Type A – classic 19-pin HDMI
 - Type B – 29-pin HDMI
 - Type C – “Mini HDMI” (mini- HDMI Type A)
 - Type D – “Micro HDMI” (looks like micro-*USB*)
 - Type E – automotive locking tab
- HDMI can also carry *Ethernet* signals.

Mini HDMI

Mini HDMI is an *HDMI* Type C (“Mini HDMI Type A”) connector.

- Mini HDMI connectors are often found on older mobile devices.
- *Micro HDMI* supersedes Mini HDMI.

Note: An adapter is required to connect *Mini HDMI* and/or *Micro HDMI* connectors to e.g. a standard HDMI Type A connector on a TV.

Micro HDMI

Micro HDMI is an *HDMI* Type E connector.

- Micro HDMI connectors are often found on newer mobile devices.
- *Micro HDMI* supersedes *Mini HDMI*.

Note: An adapter is required to connect *Mini HDMI* and/or *Micro HDMI* connectors to e.g. a standard HDMI Type A connector on a TV.

Data Device Bus

[https://en.wikipedia.org/wiki/Bus_\(computing\)](https://en.wikipedia.org/wiki/Bus_(computing))

USB

Wikipedia: <https://en.wikipedia.org/wiki/USB>

Wikipedia: https://en.wikipedia.org/wiki/USB_2.0

Wikipedia: https://en.wikipedia.org/wiki/USB_3.0

Wikipedia: https://en.wikipedia.org/wiki/USB_Type-C

USB (*Universal Serial Bus*) is a group of standards for device interaction and one-way and two-way power and data transfer.

USB bus speeds:

- USB – 12 mbps
- USB 2.0 – 480 Mbps
- USB 3.0 – 5000 Mbps (5 Gbps) (5 *Gigabit*)
- USB 3.1 – 10000 Mbps (10 Gbps) (10 *Gigabit*)
- *USB Type-C* (USB 3.1; 10 GBps)

USB connectors:

- USB Type A – classic rectangular USB with pins on one side
- USB Type B – square USB (e.g. some printers)
- Mini-USB – now deprecated (see: Micro-USB)
- USB Mini-A – (deprecated)
- USB Mini-B – (deprecated)
- Micro-USB – industry standard OMTP (2007), ITU (2009), EU (2010)
 - USB Micro-A – rectangular
 - USB Micro-B – trapezoidal
 - USB Micro-AB – supports both Micro-A and Micro-B
 - USB OTG (*on-the-go*) – (mobile) support for charging and hub
 - USB 3.0 Micro-B – Micro-USB-B + 5 pins (USB Micro-A cables work with USB 3.0 Micro-B connectors, but USB 3.0 Micro-B cables do not work with USB Micro-B connectors)
- USB 3.0 Type A – classic rectangular USB with pins on one side (works with USB Type A)
- USB 3.0 Type B – USB 3.0 Type A + extra block of pins on the top

- Each USB device can draw 5 v 500 mA of current (2.5 watts).
 - *USB Type-C* devices support 5 v 1.5 A, 3.0 A + 900 mA (e.g. for charging and powering one laptop or mobile device from another).

USB Hub

A *USB* Hub is an n-way splitter with two or more USB connectors.

- A *powered USB Hub* is a USB Hub which must be connected to an external power source; and can charge many devices
 - *USB Type-C* essentially functions as a powered USB Hub (in either direction, as power is available)

USB Keyboard

USB Mouse

USB Type-C

Wikipedia: https://en.wikipedia.org/wiki/USB_Type-C

- Operating Systems: Windows 10, OSX 10, Android M
- Adapters: DisplayPort, Thunderbolt, MHL
- USB Type-C as the primary charging interface:
 - Post-2015 MacBooks, Chromebook Pixel 2+
- Vendors with portable storage drives with USB Type-C connectors (2015):
 - LaCie, SanDisk

SATA

Wikipedia: https://en.wikipedia.org/wiki/Serial_ATA

Wikipedia: https://en.wikipedia.org/wiki/Serial_ATA#eSATA

SATA (*Serial ATA*) is a data device bus standard.

- SATA (1.5, 3.0, 6.0, 16 Gbit/s) is faster than all current *USB* standards (USB 2.0, USB 3.0. USB 3.1 (*USB Type-C*)), *IDE*, and *ATA (PATA)*
- *eSATA* is SATA for external drives.

eSATA

Wikipedia: https://en.wikipedia.org/wiki/Serial_ATA#eSATA

eSATA (*External SATA*) works with (powered, unpowered, portable) external drives.

mSATA

Wikipedia: https://en.wikipedia.org/wiki/Serial_ATA#Mini-SATA.28mSATA.29

mSATA (Mini-SATA) is a smaller *eSATA* connector.

eSATAp

Wikipedia: <https://en.wikipedia.org/wiki/ESATAp>

eSATAp (Power eSATA) is a *eSATA* connector which also includes power and supports *USB*.

An eSATAp port combines the four pins of the USB 2.0 (or earlier) port, the seven pins of the eSATA port, and optionally two 12 V power pins

- Without eSATAp, *eSATA* devices require an additional power source.

SCSI

Wikipedia: <https://en.wikipedia.org/wiki/SCSI>

DevPrefix: `/dev/sg` (CD/DVD)

DevPrefix: `/dev/sd` (harddrive, USB)

DevFS: `/dev/scsi`

SCSI (*Small Computer System Interface*) is a set of standards for device interaction and data interchange.

- Drives faster than 7200 RPM are often either *SCSI* or, now, *SATA* drives
- Some CD/DVD devices are *SCSI* devices
- *SATA* and *eSATA* devices register as SCSI devices with newer Linux kernels and distributions.

IDE

Wikipedia: https://en.wikipedia.org/wiki/Parallel_ATA#IDE_and_ATA-1

IDE (*Integrated Drive Electronics*) is a 40-pin cable connector and drive interface standard which predates (and is now part of) the *Parallel ATA* standards.

- Newer drives have *USB*, *SATA*, or *eSATA* connectors (which, like IDE, all also require the drive to handle its own storage logic)

PATA

PATA, ATA (*Parallel ATA (AT Attachment)*) is a 40-pin drive interface standard based on *IDE*, ATA, and ATAPI.

- *SATA* (*Serial ATA*) is derived from *PATA* (*Parallel ATA*).
- Newer drives have *USB*, *SATA*, or *eSATA* connectors

Persistent Storage

Slowest -> Fastest:

- Punch cards
- Tape drives
- Disk drives (*floppy*: 8", 5.25", 3.5")
- *Disc Drives* (*CD*, *MiniDisc*, *DVD*, *Blu-ray*, [3D-] optical storage)
- *Hard Drives* (*HD*)
- *SSDs*

Hard Drives

- 5400 RPM – notebook (energy savings; see also *SSD*)
- 7200 RPM – desktop, notebook
- 10000 RPM – high end *SCSI* drives

SSD

Wikipedia: https://en.wikipedia.org/wiki/Solid-state_drive

An *SSD* (*Solid-State Drive*) is a binary data storage device based on an integrated circuit that does not require voltage to be applied to maintain state.

- *SSDs* are faster and (currently) more expensive than *Hard Drives*.
- *SSDs* are more energy efficient than *Hard Drives*.
- Notebooks and netbooks may include or be upgraded with an *SSD*.
- Servers benefit from *SSDs* for caching, fast reads, and fast writes.

Disc Drives

- *CD*
- *DVD*
- *Blu-ray*
- [3D-] optical storage

See: *Data Engineering*

Network Interfaces

RS-232

Wikipedia: <https://en.wikipedia.org/wiki/RS-232>

An RS-232 *serial* or *console* port supports a DB-25 (25-pin) trapezoidal connector (most properly with stabilizing thumb screws).

- Systems with no/broken/unsupported graphics cards can often be booted and interacted with over an RS-232 serial port (given a serial console/monitor to attach)
- There are *USB* to *RS-232* cables
- There are *RS-232* over IP solutions.
- Many full-system virtualization tools support a serial / RS-232 console (for logging, debugging, emergencies).
- Some e.g. wireless routers have *RS-232* ports for flash recovery.
- Many enterprise grade switches have *RS-232* ports to which a console may be attached.

Gigabit

A gigabit is 1000 Mbps (1000 megabits per second).

- *1000BASE-T*, *1000BASE-X*, *10GBASE-T*, and *40GBASE-T* can all handle *Gigabit Ethernet* speeds (or greater)
- Wireless routers before *802.11ac* are not fast enough to handle gigabit speeds.
- Wireless routers with *802.11ac* and up (TODO) handle (shared) gigabit speeds.

NIC

A NIC (*Network Interface Card*) is a card that plugs into a *System Bus* which has one or more connectors for e.g. [wired *Gigabit Ethernet* Patch Cable].

8P8C

Wikipedia: https://en.wikipedia.org/wiki/Modular_connector#8P8C

An 8P8C modular connector is an **8-pin plug** (usually with a clip for cable safety); often for [*Ethernet*].

- Some 8PMC modular connectors include a [flexible] coat/shield/tail to help with pin/connector strain and cabling safety.
- *Ethernet* (*CAT-5*, *CAT-6*) cables have 8PMC connectors
- *HDMI* over *Ethernet* (*CAT-5*, *CAT-6*) cables have *8P8C* connectors
- *RS-232* cables have *8P8C* connectors

TIA

Wikipedia: https://en.wikipedia.org/wiki/Telecommunications_Industry_Association

Homepage: <https://www.tiaonline.org/>

Homepage: <http://www.tia-942.org/>

Standards: <https://www.tiaonline.org/standards/>

TIA (*Telecommunications Industry Association*) is a standards group for many types of wireless and wireline networking; including specs for *Ethernet*-capable cabling (*CAT-5*, *CAT-5e*, *CAT-6*, *CAT-6A*).

- List of *TIA* standards: <http://www.tiaonline.org/standards/>

T568A and T568B

Wikipedia: https://en.wikipedia.org/wiki/TIA/EIA-568#T568A_and_T568B_termination

TIA EIA-568 specifies the **T568A** and **T568BB** connector pin sequence ('pinouts*') for 4-pair (8-wire) cabling (e.g. usually with an 8P8C [RJ45 [*Ethernet*]] connector).

- An *Ethernet Crossover Cable* has one **T568A** and one **T568BB** on each end, respectively.
- An *Ethernet* patch cable has one of either *T568A and T568B* or *T568A and T568B* connectors on each end.

CAT-5

Wikipedia: https://en.wikipedia.org/wiki/Category_5_cable

A CAT-5 (*Category 5*) cable is an *Ethernet* cable.

- CAT-5 can carry *10BASE-T*, *100BASE-TX*, or *1000BASE-T*
- *CAT-5e* is the newer CAT-5 standard.

CAT-5e

Wikipedia: https://en.wikipedia.org/wiki/Category_5_cable#Category_5_vs._5e

- CAT-5e (*Category 5e*) cable is an *Ethernet* cable with more stringent *crosstalk* (~'interference') specs than a *CAT-5* cable.
- CAT-5e can carry *10BASE-T*, *100BASE-TX*, or *1000BASE-T*
- <https://en.wikipedia.org/wiki/Crosstalk>
- There is no CAT-6e, but there is a *CAT-6A*.

CAT-6

Wikipedia: https://en.wikipedia.org/wiki/Category_6_cable

A CAT-6 (*Category 6*) cable is an *Ethernet* cable

- CAT-6 can carry *10BASE-T*, *100BASE-TX*, *1000BASE-T*, *10GBASE-T*,

CAT-6A

Wikipedia: https://en.wikipedia.org/wiki/Category_6_cable#Category_6A

A CAT-6A (*Category 6A*) cable is an *Ethernet* cable

- CAT-6A can carry *10BASE-T*, *100BASE-TX*, *1000BASE-T*, *10GBASE-T*,

Ethernet

Wikipedia: https://en.wikipedia.org/wiki/Ethernet_over_twisted_pair

Wikipedia: https://en.wikipedia.org/wiki/Fast_Ethernet

- Ethernet cables are limited to 100 metres (328 ft) in length – https://en.wikipedia.org/wiki/Fast_Ethernet#Copper
- An ethernet cable installer uses a *crimper tool* to *crimp* [8PMC] connectors (“*terminators*”) to the ends of an ethernet patch cable or an *Ethernet Crossover Cable*.

Ethernet Crossover Cable

Wikipedia: https://en.wikipedia.org/wiki/Ethernet_crossover_cable

An ethernet crossover cable has *the correct pins* swapped on one or both s of a cable (*T568A and T568B* <-> *T568A and T568B*).

Because the only difference between the *T568A and T568B* and *T568A and T568B* pin/pair assignments are that pairs 2 and 3 are swapped, a crossover cable may be envisioned as a cable with one modular connector following *T568A and T568B* and the other *T568A and T568B* (see TIA/EIA-568 Wiring). Such a cable will work for *10BASE-T* or *100BASE-TX*.

—https://en.wikipedia.org/wiki/Ethernet_crossover_cable#Overview

Warning: *Ethernet Crossover Cable* != *Ethernet* Patch Cable

- Many/most(?) newer devices are auto-sensing; and so won’t short out, or maybe work at all if a crossover cable is connected to anything other than:
 - between two *NICs*
 - between router/switch ‘bridge’ or ‘uplink’ ports (with instructions designated by the manufacturer in a manual often also available online).

Patch Cable

Wikipedia: https://en.wikipedia.org/wiki/Patch_cable

A “Patch Cable” is a cable for connecting two devices (with two different or the same connectors on each end).

- e.g. an *Ethernet* patch cable
- e.g. an *Fiber* patch cable

10BASE-T

Wikipedia: https://en.wikipedia.org/wiki/Ethernet_over_twisted_pair#Cabling

10Base-T is a 10 Mbps *Ethernet* standard.

- 100BASE-TX ultimately supersedes *10BASE-T*

100BASE-TX

Wikipedia: https://en.wikipedia.org/wiki/Fast_Ethernet#100BASE-TX

100BASE-TX is a 100 Mbps *Ethernet* standard.

- 100BASE-T is backward-compatible with *10BASE-T* (some cards will say 10/100)
- 100BASE-TX ultimately supersedes *10BASE-T*

Gigabit Ethernet

- Copper Gigabit Ethernet: *1000BASE-T*, *10GBASE-T*, *40GBASE-T*
- Fiber Gigabit Ethernet : *1000BASE-X*, [...]

1000BASE-T

Wikipedia: https://en.wikipedia.org/wiki/Gigabit_Ethernet#1000BASE-T

1000BASE-T is a 1000 Mbps (1 Gbps; 1 *Gigabit Ethernet*) *Ethernet* standard.

- 100BASE-T is backward-compatible with 10BASE-T (some cards will say 10/100, or 10/100/1000)

1000BASE-X

Wikipedia: https://en.wikipedia.org/wiki/Gigabit_Ethernet#1000BASE-X

1000BASE-X is a 1000 Mbps (1 Gbps; 1 *Gigabit Ethernet*) *Fiber Ethernet* standard.

10GBASE-T

Wikipedia: https://en.wikipedia.org/wiki/10_Gigabit_Ethernet#10GBASE-T

10GBASE-T is a 10000 Mbps (10 Gbps) copper *Ethernet* standard.

40GBASE-T

Wikipedia: https://en.wikipedia.org/wiki/100_Gigabit_Ethernet#40GBASE-T

40GBASE-T is a 40000 Mbps (40 Gbps) copper *Ethernet* standard with *ISO* Cat.8 cables.

Wireless

Wikipedia: <https://en.wikipedia.org/wiki/Wireless>

- Wireless / Mobile / Radiofrequency
- Wi-Fi == *802.11* wireless networking

802.11

Wikipedia: https://en.wikipedia.org/wiki/IEEE_802.11

IEEE 802.11 is a group of standards for wireless networking.

IEEE 802.11 is a set of media access control (MAC) and physical layer (PHY) specifications for implementing wireless local area network (WLAN) computer communication in the 2.4, 3.6, 5, and 60 GHz frequency bands.

802.11 “WiFi” standards:

IEEE Standard	Frequency Band[s]	Bandwidth (bits per second)
802.11b	2.4 GHz	11 Mbps
802.11g	2.4 GHz	54 Mbps
802.11a	5 GHz	54 Mbps
802.11n	2.4 GHz, 5 GHz	600 Mbps (MIMO)
802.11ac	2.4 GHz, 5 GHz	1300 Mbps (MIMO)
802.11ad	60 GHz	7000 Mbps (“WiGig”)
802.11ax	2.4 GHz, 5 GHz	~4x 802.11ac [Draft]
802.11ay	60 GHz	100000 Mbps (100 Gbps) [Draft]

MIMO Multiple-Input Multiple-Output 802.11 devices aggregate multi-link capacity (and so operate with multi-band presence).

Other IEEE 802.11 standards:

- 802.11s – wireless mesh networking
-

Note: Microwave ovens and cordless telephones, for example, operate in the 2.4 GHz (and 900 MHz) bands; which may cause interference in the already-crowded 2.4 GHz frequencies; which is why 802.11a, 802.11n, 802.11ac all [also] support the 5 GHz (*hertz*) range.

Wi-Fi Direct

Wikipedia: https://en.wikipedia.org/wiki/Wi-Fi_Direct

Wi-Fi Direct is an alternative to Bluetooth with relatively seamless wireless device pairing/configuration.

Wireless mesh network

https://en.wikipedia.org/wiki/Wireless_mesh_network

Wireless mesh networks route connections between nodes.

- Wireless mesh networks do not require APs.
- Wireless mesh networks are designed to be resilient to and tolerant of network failure.
- Wireless mesh networks require ingress and egress points in order to route with the wider internet.

Mesh Wireless Approaches:

- OLPC (*One Laptop Per Child*) laptops support IEEE 802.11s mesh networking with standard 802.11b/g wireless cards.

Wikipedia: https://en.wikipedia.org/wiki/One_Laptop_per_Child#Technology

Wikipedia: https://en.wikipedia.org/wiki/OLPC_XO#Networking

Docs: http://wiki.laptop.org/go/Mesh_Network_Details

– http://wiki.laptop.org/go/XO-4_Touch

- Redstone Technologies LLC (*Gigabit* wireless mesh networks)
 - <http://redstone.us.com/simplified-wireless-architecture/>
 - <http://redstone.us.com/8-major-breakthroughs/>

Mobile Broadband

https://en.wikipedia.org/wiki/Mobile_broadband

3G

Wikipedia: <https://en.wikipedia.org/wiki/3G>

4G

Wikipedia: <https://en.wikipedia.org/wiki/4G>

- *LTE* is a *4G* wireless standard.

5G

Wikipedia: <https://en.wikipedia.org/wiki/5G>

CDMA

Wikipedia: https://en.wikipedia.org/wiki/Code_division_multiple_access

- W-CDMA (*WCDMA*) is a *3G* wireless standard.

GSM

Wikipedia: <https://en.wikipedia.org/wiki/GSM>

LTE

Wikipedia: [https://en.wikipedia.org/wiki/LTE_\(telecommunication\)](https://en.wikipedia.org/wiki/LTE_(telecommunication))

- Artemis pCell pWave LTE: <http://www.artemis.com/pcell>

Fiber

https://en.wikipedia.org/wiki/Fiber-optic_communication

https://en.wikipedia.org/wiki/Optical_fiber

Optical fiber can carry photons near the speed of light.

Fiber to the x

Wikipedia: https://en.wikipedia.org/wiki/Fiber_to_the_x

The various fiber deployment strategies are somewhat undescriptively all called “fiber to the x (*FTTX*)”.

- Fiber to the neighborhood
- Fiber to the premises
- Fiber to the home

Fiber is the way forward in *wireline* broadband networks:

- https://en.wikipedia.org/wiki/Fiber_to_the_premises_by_country
- https://en.wikipedia.org/wiki/Fiber_to_the_premises_in_the_United_States

Fibre Channel

Wikipedia: https://en.wikipedia.org/wiki/Fibre_Channel

Fibre channel is an optical fiber networking technology.

Network Glossary

SOHO Small home office

Modem Modulator/demodulator. Transforms signals between *Ethernet*, *Ethernet* over *USB*, *Coaxial Cable (Co-ax)*, *Copper Phone*, *DSL Twisted Copper Pair(s)*, *High Frequency Radio Mobile Broadband (cell-modem)*, *SDR (Software Defined Radio)*, *Fiber to the x*.

ARP ARP (*Address Resolution Protocol*) is an *IETF* standard for linking *IP* addresses with *Ethernet* addresses.

Standard: <https://tools.ietf.org/html/rfc826>

```
arp -a # list arp entries [BSD style]
arp -e # list arp entries [Linux tabular format]
arp -n # do not resolve DNS addresses (display IPs instead)
```

DHCP DHCP (*Dynamic Host Configuration Protocol*) is an internet standard for acquiring IP address leases and e.g. DNS resolver configuration settings.

Most SOHO routers include a DHCP server.

Some ISPs offer ‘dynamic’ IPs: an IP lease is acquired and/or released for a given MAC address, and may change as often as the ISP DHCP configuration specifies.

Some ISPs offer ‘static’ IPs: through DHCP and/or manual configuration of static routes.

If a DHCP lease expires while e.g. downloading or streaming, the connections are only dropped if the address is also removed from the interface (e.g. by dhclient upon notification of the DHCP lease expiration). One workaround for this is to preemptively renew the DHCP lease (as dhclient usually does).

To troubleshoot an ISP connection by removing the router and directly attaching a device with a firewall that starts at boot time, it’s often necessary to ‘clone’ the *MAC address* (for a static IP) or release the DHCP lease and/or power-cycle the modem.

Wikipedia: https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol

Gateway A gateway provides a connection to another *IP* network.

A routing table may include a default gateway *network route* and zero or more additional routes defined by system configuration and/or *DHCP*

Network Interface A network interface is a software identifier for a “port” on a *NIC*.

```
## List interfaces
ifconfig # UNIX, BSD, Linux
ipconfig # Windows
ip l #link # Linux
ip -s link # Linux
ip a #addr # Linux

## Bring an interface up or down
ifconfig eth0 up 127.2.2.2 netmask 255.255.255.0 broadcast 127.2.2.255
ifconfig eth0 down
ip l set dev eth0 up
ip l set dev eth0 down
```

- There may be one or more network interfaces registered to a *NIC*:

```
ip addr add 127.1.2.3 dev lo netmask 255.255.255.0
```

NIC A *NIC* (*Network Interface Card*), or *Network Card*, is a physical adapter for connecting to a network (generally with a physical *Ethernet* (or *Fiber*) cable).

- A *Wireless NIC* is also called a *wireless card* or *wireless adapter*.

Docs: *NIC*

Network Route Packets are routed based upon network routing tables: rules for which *network interface(s)* and/or link(s)/tunnel(s) to send packets destined for anywhere (default gateway) or a specific network subnet (identified by e.g. a /24 prefix or a 255.255.255.0 subnet mask).

```
netstat -rn # BSD, OSX, Linux
route -n    # Linux # Flags="G"
route add gw 192.168.1.1 dev eth0
ip r #route # Linux      # "default via"
ip r add default gw 10.2.2.253
ip r add 10.2.3.0/24 via 10.2.2.253 dev eth0

# routes are usually managed by the OS.
# to reset them all and reboot all interfaces with downtime (!):
# /etc/init.d/networking restart # SysV Ubuntu, Debian
# service network restart      # systemd {...}

# restart interface eth0
# #(and re-run dhcplient for an IP lease from DHCP
# # (e.g. from a router that assigns local IPs like 192.168.0.101))
# ifdown eth0; ifup eth0      # Linux [man ifdown ifup]
```

Hub A network hub connects (or *bridges*) multiple network interfaces to a shared bus; where all machines get a copy of all packets as repeated by the hub.

- Normally, a *NIC* (device, driver) is configured to select only packets destined for local addresses.
- (In *promiscuous mode*, an interface receives all packets regardless of the packet protocol destination field. (Useful for debugging e.g. *ARP*)).

Bridge A network bridge is a *hub*; often a software defined bridge to which *network interfaces* can be *added/attached* and *deleted/removed*.

In Linux, network bridges are configured with `brctl`:

```
brctl
brctl show

# create a bridge which will last until reboot
brctl addbr br0
brctl addif br0 eth0
brctl addif br0 eth1
brctl show br0
brctl showmacs br0
```

Switch A network switch routes packets to specific ports based upon an *ARP* table.

- *SOHO Routers* are often configured as a switch (relay packets); with a fallback or configuration setting to *hub* mode (repeat all packets).

Router A *SOHO Router* is usually configured as a *Gateway* and a *Switch* (with at least one port allocated to a different network connection).

- An actual (“trunk”, “internet”, “backbone”) router is usually configured as a *Router*, with routing table advertisement protocols like RIP, BGP, and **IPv6 radvd**; in order to optimize throughput and minimize latency.
- *SOHO Routers* are often configured as a switch (relay packets); with a fallback or configuration setting to *hub* mode (repeat all packets).

AP A *Wireless* access point (*AP*) is a wireless *NIC* with one or more antennas and a power source; which often also acts as a router and/or a wireless mesh router.

Note: Cool *AP* Ideas (cost, maintenance, spacing, redundancy, resiliency):

- [] Wifi Streetlamp posts
 - [] Wifi Parking meters
 - [] Solar wireless APs (shaped as flowers for cost and water/cleaning/maintenance)
 - [] MEMS wireless APs (see also: Bluetooth BLE)
-

WiFi WiFi is a name for a set of *IEEE 802.11 Wireless* standards.

Wikipedia: https://en.wikipedia.org/wiki/Information_systems

- https://en.wikipedia.org/wiki/Information_systems#Types_of_information_systems
- https://en.wikipedia.org/wiki/List_of_collaborative_software

See also:

- *Team Building > Collaborative Software*

Information Systems Glossary

System An interconnected graph of nodes linked with edges through which flow produces predictable and unpredictable behaviors.

<https://en.wikipedia.org/wiki/System>

Infrastructure Core *system* components.

Information Not noise.

See: *Data Science > Information Theory, Knowledge Engineering*

Information System A *system* with *information* inputs and outputs designed to solve a problem.

https://en.wikipedia.org/wiki/Information_system

Information Technology Hardware and software *information systems* for solving business problems.

https://en.wikipedia.org/wiki/Information_technology

Application An application of technology for a particular domain.

App See *Application*

Operating System Software for running *applications* and working with devices.

https://en.wikipedia.org/wiki/Operating_system

API Advanced Programming Interface. A specification for inputs and outputs provided by a system component.

An API specification is specified, versioned, updated, and deprecated.

A reference implementation implements a model copy of an API.

If a component utilizes an API it is said to be written against, with, or on top of said API.

An API abstracts a lower-level in order to provide a higher-level API.

One primary advantage of an API is that, ideally, system components are loosely coupled and thus interchangeable and incrementally upgradeable.

https://en.wikipedia.org/wiki/Application_programming_interface

https://en.wikipedia.org/wiki/Loose_coupling

[https://en.wikipedia.org/wiki/Abstraction_\(computer_science\)](https://en.wikipedia.org/wiki/Abstraction_(computer_science))

Operating System API An *API* provided by one or more *Operating Systems*.

Examples:

- <https://en.wikipedia.org/wiki/POSIX>
- https://en.wikipedia.org/wiki/Windows_API

Language API An *API* provided by or for one or more programming languages through a *standard library* or a third-party component.

Examples:

- <https://en.wikipedia.org/wiki/WebGL>

https://en.wikipedia.org/wiki/Public_interface

[https://en.wikipedia.org/wiki/Protocol_\(object-oriented_programming\)](https://en.wikipedia.org/wiki/Protocol_(object-oriented_programming))

https://en.wikipedia.org/wiki/Standard_library

Web API An *API* for interacting with local (browser) or remote (e.g. *HTTP*) components.

One primary advantage of a Web API is that *downstream components* do not need to know anything about the underlying *Operating System APIs* and *Language APIs*.

https://en.wikipedia.org/wiki/Web_API

Web Service A *Web API* with a formal specification.

Generally provided over *HTTP*, traditionally (as in the *enterprisey* WS-* standards) with *XML*, but, more recently, with *JSON*.

https://en.wikipedia.org/wiki/Web_service

Webhook A webhook is an HTTP message for an event.

web services and *applications* can send webhooks when e.g. a change occurs.

For example, when GitHub, ReadTheDocs, and Travis-CI are configured (with webhooks) a change pushed to a GitHub repository branch or pull request enqueues a build at e.g. ReadTheDocs and/or Travis-CI (and GitHub can show the build status as a linked icon).

In terms of *web services*, a webhook is usually just a *JSON* POST to a *URL*; with an access token.

RESTful API A RESTful API (*REST API*) is a *Web API* that abides by best-practice guidelines for interacting with *resources* through standard *HTTP* methods like PUT/POST, GET, POST/PUT, DELETE.

Many web developers prefer RESTful APIs because the standard methods and error messages specified by *HTTP* are already implemented by existing, well-tested libraries available for most languages.

https://en.wikipedia.org/wiki/Representational_state_transfer

https://en.wikipedia.org/wiki/Create,_read,_update_and_delete

Criteria

- *ROI*
- *Need / Want*
- *Maintainability*

seeAlso:

- *Computer Science Curricula* (Criteria)
- *Information Security* (Risk)

ROI

The investment should be justified by gains in productivity & efficiency.

Productivity & Efficiency:

```
output / input
returns / costs
```

Units and ratios:

- Hours

- Dollars
- *Units: e.g. QUDT URIs (RDF)*

```
widgets / hours ----*----> widgets/hour
```

```
** curves
```

```
reduction in {risk, error, loss} / time spent on {workflow, automation}
```

seeAlso:

- *Business > ROI*
- <https://en.wikipedia.org/wiki/Productivity>
- <https://en.wikipedia.org/wiki/Efficiency>

Need / Want

Communication and collaboration are essential to success.

Maintainability

Choosing Components

- In n-years, will I be able to find someone who can maintain this?
 - Locally? Globally?
 - Job listing keyword search [rough approximation]
 - Job board search
- In n-years, will there still be a community supporting these components?
- Is it open source? How do we find/pay/train someone to understand how it works?
- Is there a *non-profit* software foundation behind this component?

Clouds

Wikipedia: https://en.wikipedia.org/wiki/Cloud_computing

https://en.wikipedia.org/wiki/Cloud_computing#Service_models

Cloud Questions

- Why would I want to run my business “in the cloud”?
- What are our core competencies?
- Do we need/want to run this all ourselves?
- Would it be more safe/secure to outsource the management of these business systems?

Cloud Application Layers

- *Applications* Layer
 - <https://github.com/wrdrd/wrd/blob/master/app.yaml> (AppEngine/AppScale)
- Platform Layer
 - *PaaS* APIs (*AppEngine*, *AppScale*, *Heroku*)
 - *Databases* (*Relational Databases*, *SQL*, *NoSQL Databases*, *Graph Databases*, *RDF Triplestores*, *SPARQL*)
 - Authz: Authentication, Authorization
 - * Servers: OpenLDAP, 389, FreeIPA, ActiveDirectory, Samba4
 - * Standards: WebID, OAuth, SASL, SAML, LDAP
 - * Infrastructure Layer: OpenStack Keystone
 - Email
 - * AppEngine/AppScale Email API
 - Logging / Monitoring
 - * AppEngine/AppScale Logging
 - * Sentry, Raven (POST JSON)
 - * FireLogger
 - Message Queue (Task Queue, Channels, *WebSockets*, AMQP, Worker Pools)
 - * Celery: <https://celery.readthedocs.io/en/latest/getting-started/brokers/>
 - * AppEngine/AppScale Tasks API
 - * Mesos schedulers (Aurora, Chronos, Marathon)
 - Search Indexing (*JSON*, *json-ld*-, *JSON-LD*)
 - * *ElasticSearch (Lucene)*, *Solr*
 - * AppEngine/AppScale Search API
 - Big Data storage and data local iterators:
 - * *MapReduce*, *Bulk Synchronous Parallel*
 - * *HDFS*, *DDFS*
 - * *Hadoop*, *Spark*, *Cassandra*, *Accumulo*
 - * Mesos
 - * Ibis
 - HTTP Object Storage
 - * libcloud (<https://libcloud.readthedocs.io/en/latest/storage/>)
 - * Amazon *Amazon AWS S3*
 - * OpenStack Swift (*Swift*, *S3* APIs)
 - * Linux: *Ceph* RADOS Gateway (*Swift*, *S3* APIs)
 - * Linux, OSX, POSIX: *GlusterFS* (*Swift*, *S3* APIs)

- Infrastructure Layer (“provisioning”)
 - Resource Pool (“instances” of {virtual machines, containers, task workers})
 - * Cobbler (libvirt virtualization)
 - * Docker Swarm
 - * Kubernetes Docker pods
 - * Kubernetes-Mesos: Mesos schedulers and Kubernetes Docker pods
 - * Mesos schedulers (Aurora, Chronos, Marathon)
 - * OpenStack
 - Inventory / Monitoring
 - * #ConfigMgmt: Packages
 - * #ConfigMgmt: Ansible, Puppet, Salt, configuration management
 - * #ConfigMgmt #Instances: Cobbler Web Dashboard
 - * #ConfigMgmt #Instances: OpenStack Horizon Dashboard
 - * #Inventory: Cobbler JSON, Salt Grains, Puppet Facts, osquery SQL tables
 - * #Monitoring: #ConfigMgmt events
 - * #Monitoring: Nagios, Shinken, Icinga
 - * #Monitoring: Monitd, Collectd, RRD
 - * #Logging: Syslog, LogStash, Heka (logtailers)
 - * #Logviz: Kibana, Grafana, Graphite, RRD (*Data Visualization Tools*)
 - * #PRF: <http://www.brendangregg.com/linuxperf.html>
 - * Docs: <https://www.opsschool.org/en/latest/>
 - Physical and Virtual Servers (“servers”, “racks”, “machines”)
 - * #ConfigMgmt: *Configuration Management* (“infrastructure as code”, shell-escaping, task queues / *ESBs*, *Distributed Computing Protocols*, #Logging, worker pools)
 - * Virtualization (full / para-virtualization)
 - * Operating Systems (“OS”)
 - * Manual Testing – PEBKAM (*Between Keyboard and Monitor*)
 - * *Computer Engineering* (“computer”, #power AC/DC, BIOS, *NIC*, *Ethernet*, USB, RS232 Serial Console, KVM switch)
 - Physical and Software Networks
 - * Internet connections
 - * VLANs, IPv6 6to4 tunnels
 - * SDN — Software Defined Networking
 - OpenStack Neutron
 - Docker networking
 - Docker Swarm networking
 - Vagrant networking

- Virtualbox networking
- * *DNS* (BIND, dnsmasq, Cobbler, OpenStack, Vagrant, Docker)
- Physical and Virtual Storage
 - * Virtual storage:
 - Central file storage (*NFS, CIFS, WebDAV*)
 - distributed file storage: (*Ceph, GlusterFS*)
 - persistent block storage: (*Amazon AWS S3, OpenStack Cinder*)
 - on-disk filesystems: (*ext4, LVM, btrfs, NTFS, FAT, HFS+*)
 - * Physical storage
 - SAN – Storage Area Network (*iSCSI, Fibre Channel, persistent block storage*)
 - NAS – Network Attached Storage (*FreeNAS, Synology, Network Filesystems, iSCSI*)
 - *Data Device Bus: (USB, SATA, SCSI, IDE, ATA Hard Drives)*
 - * Power, HVAC, Fire Suppression
 - #HVAC: Cooling (Active / Passive)
 - #HVAC: Heating (Active / Passive)
 - #Fire: Sprinklers
 - #Fire: Fire Extinguishers
 - Class A Fire Extinguisher – silver – water
 - Class C Fire Extinguisher – red – chemicals (e.g. CO2)
 - Do not pour water on an electrical fire until you have switched off the power source at the circuit breaker box.
 - Novels/films: 20,000 Leagues Under the Sea, Sphere
 - **How to Extinguish a Fire at the Initial Stages:** <http://www.wikihow.com/Extinguish-a-Fire-at-the-Initial-Stages>
 - Extinguishing **Electrial** Fires: **Red Class C** Extinguisher
 - Extinguishing **Liquid/Oil** Fires
 - Extinguishing **Organic** Fires
 - #Power: Power supplies (AC → DC conversion)
 - #Power: Batteries (AC → DC → Batteries → AC/DC)
 - #Power: Generators
 - #Power: Generator fuel for the generators
 - #Power: Renewable energy, Clean Energy, Sustainable Energy

SaaS

Wikipedia: https://en.wikipedia.org/wiki/Software_as_a_service

Wikipedia: https://en.wikipedia.org/wiki/Application_service_provider

SaaS (*Software-as-a-Service*) is a service provision, application lifecycle, and recurring billing model for providing hosted applications.

Examples of SaaS:

- *ReadTheDocs* is a *SaaS* which can integrate with GitHub (also a *SaaS* offering) through *Webhooks*:
- Travis-CI is a *SaaS Continuous Integration* service which pulls and builds from a GitHub repository upon receipt of a *Webhook*, that is free for *Open Source* projects
https://en.wikipedia.org/wiki/Travis_CI
- Google Gmail is a *SaaS* webmail service:
<https://en.wikipedia.org/wiki/Gmail>
- Many *Customer Relationship Management (CRM)* software applications are offered as *SaaS* subscription services
- See: *Business Modeling*
SaaS is distinct from e.g. subscription software licensing; because *SaaS* applications are usually *hosted* by the service provider

PaaS

Wikipedia: https://en.wikipedia.org/wiki/Platform_as_a_service

PaaS (*Platform-as-a-Service*) platforms offer platform APIs on top of which applications can be developed and marginally scaled if designed and developed for concurrency and asynchronicity.

Examples of PaaS Platforms:

- *AppEngine*
- *AppScale*
- *Deis*
- <https://github.com/progrium/dokku> – dokku is an extremely minimal (no firewall, etc.) “Docker powered mini-Heroku in around 100 lines of Bash” (see also: bashreduce)
 - <https://news.ycombinator.com/item?id=9054503> – The original dokku developer now works with *Deis*
 - <https://github.com/dokku-alt/dokku-alt> – dokku-alt is a fork of dokku
- <https://github.com/flynn/flynn>
- *Heroku*

AppEngine

Wikipedia: https://en.wikipedia.org/wiki/Google_App_Engine

Homepage: <https://cloud.google.com/appengine/>

Twitter: <https://twitter.com/googlecloud>

Project: <https://code.google.com/p/googleappengine/>

Source: svn <http://googleappengine.googlecode.com/svn/trunk/python> appengine

Source: svn <http://googleappengine.googlecode.com/svn/trunk/java> appenginejava

Source: hg <https://code.google.com/p/appengine-go/>

Source: git <https://github.com/GoogleCloudPlatform/appengine-php>
 Docs: <https://developers.google.com/appengine/>
 Docs: <https://cloud.google.com/appengine/docs>
 Docs: <https://cloud.google.com/appengine/docs/python/>
 Docs: <https://cloud.google.com/appengine/docs/java/>
 Docs: <https://cloud.google.com/appengine/docs/go>
 Docs: <https://cloud.google.com/appengine/docs/php>
 Docs: <https://code.google.com/p/googleappengine/issues/list>
 Docs: <https://cloud.google.com/nodejs/>
 Docs: <https://cloud.google.com/docs/>

Google AppEngine is an *PaaS* platform for developing and scaling web applications written in Python, Java, Go, and *PHP*.

- AppEngine applications can interface with Google Cloud Platform *Web APIs*

AppScale

Wikipedia: <https://en.wikipedia.org/wiki/AppScale>
 Homepage: <http://www.appscale.com/>
 Twitter: <https://twitter.com/appscalecloud>
 Download: <https://github.com/AppScale/appscale/releases>
 Source: git <https://github.com/AppScale/appscale>
 Source: <https://github.com/AppScale/appscale/tree/master/AppDB/cassandra>
 Docs: <https://github.com/appscale/appscale/wiki>
 Docs: <https://github.com/AppScale/appscale/wiki/Installing-AppScale-from-source-on-GitHub>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-Google-Compute-Engine>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-Amazon-EC2>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-Eucalyptus>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-Eucalyptus>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-VirtualBox>
 Docs: <https://github.com/AppScale/appscale/wiki/AppScale-on-KVM>
 Docs: <https://github.com/AppScale/appscale/wiki/Virtualized-Cluster>
 Docs: <https://github.com/AppScale/appscale/wiki/Autoscaling-Triggers>
 Docs: <https://github.com/AppScale/appscale/wiki/Adding-Support-for-a-New-Database>
 Docs: <https://github.com/AppScale/appscale/wiki/Search-API-in-AppScale>
 Docs: <https://github.com/AppScale/appscale/wiki/Logging-in-AppScale>
 Docs: <https://github.com/AppScale/appscale/wiki/Managing-Users>

AppScale is a completely *Open Source PaaS* platform for developing and scaling web applications written in Python, Java, Go, and *PHP*.

- AppScale Python apps deploy applications from `app.yaml` YAML files; just like *AppEngine*
- AppScale development supporters include Google and NSF: <http://googlecloudplatform.blogspot.com/2015/05/AppScale-and-App-Engine-Work-Together-to-Provide-Infrastructure-Flexibility.html>
- AppScale applications can interface with AppScale implementations of *AppEngine* Google Cloud Platform *Web APIs*

Table 7.1: adapted from <https://github.com/AppScale/appscale/wiki/How-AppScale-implements-the-Google-App-Engine-APIs>

<i>AppEngine</i> API	<i>AppScale</i> implementation
Datastore	AppDB { Cassandra, Thrift, Protocol Buffers }
Memcache	memcached
URL Fetch	urllib2
Blobstore API	custom server built on Tornado
XMPP	ejabberd
Channel API	ejabberd and strophejs
Mail	sendmail
Images	Python Imaging Library (PIL)
Task Queue	RabbitMQ
Cron	Vixie Cron
Search	SOLR
CloudSQL	MySQL
Users	AppScale Dashboard

Deis

Homepage: <http://deis.io/>

Source: git <https://github.com/deis/deis>

Source: <https://github.com/deis/deis/blob/master/Makefile>

Source: <https://github.com/deis/deis/blob/master/Vagrantfile>

Docs: <http://docs.deis.io/en/latest/>

Docs: <http://docs.deis.io/en/latest/toctree/#toctree>

Docs: http://docs.deis.io/en/latest/installing_deis/

Docs: http://docs.deis.io/en/latest/installing_deis/aws/

Docs: http://docs.deis.io/en/latest/installing_deis/baremetal/

Docs: http://docs.deis.io/en/latest/installing_deis/gce/

Docs: http://docs.deis.io/en/latest/installing_deis/openstack/

Docs: http://docs.deis.io/en/latest/installing_deis/rackspace/

Docs: http://docs.deis.io/en/latest/installing_deis/vagrant/

Docs: http://docs.deis.io/en/latest/understanding_deis/concepts/

Docs: http://docs.deis.io/en/latest/understanding_deis/architecture/

Docs: http://docs.deis.io/en/latest/understanding_deis/components/

Docs: http://docs.deis.io/en/latest/using_deis/deploy-application/

Docs: http://docs.deis.io/en/latest/using_deis/using-buildpacks/

Deis is an *Open Source PaaS* platform built on Docker and CoreOS written in Python and Go.

- Apps are deployed to Deis with git (`git push`) or the `deis` CLI client.
- configuration management is useful but not necessary for provisioning Deis (e.g. creating and managing custom deis images and containers with extra libraries and configuration).
- Deis builds with Make, Docker Dockerfiles, and CoreOS.
 - <https://github.com/deis/deis/blob/master/controller/Dockerfile>
 - <https://github.com/deis/deis/blob/master/controller/requirements.txt>

- <https://github.com/deis/deis/blob/master/database/Dockerfile>
- <https://github.com/deis/deis/blob/master/store/Makefile>
- <https://github.com/deis/deis/tree/master/tests>
- Deis can work with the Linux Ceph filesystem.
 - Deis supports *Heroku* Buildpacks: [#included-buildpacks](http://docs.deis.io/en/latest/using_deis/using-buildpacks/#included-buildpacks)
 - * Ruby, Node.js, java, Gradle, Grails, Play, Python, PHP, Clojure, Scala, Go
 - * buildpacks are composable: <https://github.com/heroku/heroku-buildpack-multi>
- Deis can scale to n instance of containers per process (e.g. web):

```
deis scale web=3
```

See also: *Heroku*, Kubernetes, Kubernetes-Mesos

Heroku

Wikipedia: <https://en.wikipedia.org/wiki/Heroku>

Homepage: <https://www.heroku.com/>

Twitter: <https://twitter.com/heroku>

Source: <https://github.com/heroku>

Docs: <https://devcenter.heroku.com/articles/git>

Heroku is a *PaaS* Platform.

Deis supports *Heroku* Buildpacks.

IaaS

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Cloud_infrastructure

Infrastructure-as-a-Service providers provide a bit more than regular hosting services in that they offer something like virtual datacenter resource pools: servers, networks, and redundant storage systems on top of which IT systems can be developed, tested, and deployed.

Examples of IaaS:

- *Amazon AWS*
- *Google Cloud*
- *Rackspace Cloud*
- libcloud implements a Python *language api* over very many *IaaS* and *PaaS* clouds: https://libcloud.readthedocs.io/en/latest/supported_providers.html

Amazon AWS

Wikipedia: https://en.wikipedia.org/wiki/Amazon_Web_Services
Wikipedia: https://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud
Wikipedia: https://en.wikipedia.org/wiki/Amazon_Elastic_Block_Store
Wikipedia: https://en.wikipedia.org/wiki/Amazon_S3
Wikipedia: https://en.wikipedia.org/wiki/Amazon_Relational_Database_Service
Wikipedia: https://en.wikipedia.org/wiki/Amazon_CloudFront
Homepage: <https://aws.amazon.com/>
Twitter: <https://twitter.com/awscloud>
Docs: <https://aws.amazon.com/products/>
Docs: <https://aws.amazon.com/ec2/pricing/>
Docs: <https://aws.amazon.com/ebs/pricing/>
Docs: <https://aws.amazon.com/s3/pricing/>
Docs: <https://aws.amazon.com/rds/pricing/>
Docs: <https://aws.amazon.com/cloudfront/pricing/>
Docs: <https://aws.amazon.com/cloudformation/>

- EC2 – Elastic Compute Cloud (CPU/GPU/RAM instances)
- EBS – Elastic Block Store (persistent block storage)
- S3 – Simple Storage Service (HTTP object storage)
- SQS – Simple Queue Server
- CloudFormation – EC2 [auto-]scaling
- CloudFront – CDN
- RDS: Managed MySQL, Oracle, SQL Server, PostgreSQL
- DynamoDB: *NoSQL Databases* supercolumn cloud datastore

Python and AWS

- boto, libcloud

Google Cloud

Wikipedia: https://en.wikipedia.org/wiki/Google_Cloud_Platform
Wikipedia: https://en.wikipedia.org/wiki/Google_Compute_Engine
Wikipedia: https://en.wikipedia.org/wiki/Google_Cloud_Messaging
Homepage: <https://cloud.google.com/>
Twitter: <https://twitter.com/googlecloud>
Docs: <https://cloud.google.com/products/>
Docs: <https://cloud.google.com/compute/>
Docs: <https://cloud.google.com/container-engine/>
Docs: <https://cloud.google.com/dns/>
Docs: <https://cloud.google.com/datastore/>
Docs: <https://cloud.google.com/storage/>
Docs: <https://cloud.google.com/sql/>
Docs: <https://cloud.google.com/bigquery/>

Docs: <https://cloud.google.com/dataflow/>

Google Cloud Platform (GCP) is an *IaaS* cloud platform.

- *AppEngine* – *PaaS*
- Compute Engine (GCE) – *IaaS*
 - KVM, SDN
- Container Engine (GCE) – *IaaS*
 - kubernetes, docker
- CloudSQL (MySQL)

Python and Google Cloud

- libcloud
- <https://cloud.google.com/compute/docs/tutorials/python-guide>
- <https://github.com/westurner/dotfiles/blob/master/etc/bash/08-bashrc.gcloud.sh>

Rackspace Cloud

Wikipedia: https://en.wikipedia.org/wiki/Rackspace_Cloud

Homepage: <https://www.rackspace.com/cloud>

Twitter: <https://twitter.com/Rackspace>

Docs: <http://docs.rackspace.com/>

Rackspace Cloud is an *IaaS* cloud platform built with OpenStack.

- Rackspace CloudFiles is now OpenStack Swift.
- Rackspace Cloud is powered by OpenStack.
- Cloud Servers – OpenStack Compute
- Cloud Load Balancers
- Cloud DNS
- Cloud Networks
- Cloud Block Storage – OpenStack Cinder
- Cloud Files – OpenStack Swift
- CDN
- Cloud Databases (MySQL)
- Cloud Big Data (Hadoop)
- Cloud Queues
- Rackspace Auto Scale
- Rackspace Private Cloud v4, v9, v10 (OpenStack)

Python and Rackspace Cloud

- libcloud

- <https://developer.rackspace.com/sdks/python/>
- <https://github.com/rackspace/pyrax>

Configuration Management

Wikipedia: https://en.wikipedia.org/wiki/Configuration_management#Software

Wikipedia: https://en.wikipedia.org/wiki/Software_configuration_management

https://en.wikipedia.org/wiki/Comparison_of_open-source_configuration_management_software

See: Tools > Configuration Managment

Information Security

Wikipedia: https://en.wikipedia.org/wiki/Information_security

Managing risk and uncertainty.

Standards:

- https://en.wikipedia.org/wiki/Statement_on_Auditing_Standards_No._70:_Service_Organizations
- https://en.wikipedia.org/wiki/Evaluation_Assurance_Level
- https://en.wikipedia.org/wiki/Cloud_computing_security

See: *WebSec*

Confidentiality

Wikipedia: <https://en.wikipedia.org/wiki/Confidentiality>

Integrity

Wikipedia: https://en.wikipedia.org/wiki/Data_integrity

Availability

Wikipedia: <https://en.wikipedia.org/wiki/Availability>

https://en.wikipedia.org/wiki/Service-level_agreement

Business Continuity

Wikipedia: https://en.wikipedia.org/wiki/Business_continuity

https://en.wikipedia.org/wiki/Business_continuity_planning

https://en.wikipedia.org/wiki/Disaster_recovery

See: *Information Security (Availability)*

Backups

Wikipedia: <https://en.wikipedia.org/wiki/Backup>

Reliability

Wikipedia: https://en.wikipedia.org/wiki/Reliability_engineering

[https://en.wikipedia.org/wiki/Redundancy_\(engineering\)](https://en.wikipedia.org/wiki/Redundancy_(engineering))

Scenarios

https://en.wikipedia.org/wiki/Scenario_planning

Business Systems

https://en.wikipedia.org/wiki/Online_office_suite

https://en.wikipedia.org/wiki/Comparison_of_office_suites

Google Apps

- \$5/user/month // \$50/user/year
- gmail (e.g. `username@example.org`)
 - can add aliases (e.g. `webmaster@example.org` -> `username@example.org`)
 - can setup forwarding (e.g. `username@example.org` -> `username@gmail.com`)
- mail, contacts, chat, calendar, drive, docs, sheets, slides, groups, sites
 - drive: online storage
 - sites: (e.g. `employees.example.org`)
 - * update through web interface
 - * file sharing (employee handbook pdf)
 - groups: [employee] emailing list

- * don't need accounts for every employee, they can use their own email addresses (everything gets relayed)
- * basically like archived emails with always on reply-all

- <http://google.com/a>
- <http://learn.googleapps.com/>
- <https://www.google.com/enterprise/marketplace/>

Handling Money

https://en.wikipedia.org/wiki/Transaction_processing

https://en.wikipedia.org/wiki/Payment_Card_Industry_Data_Security_Standard

Accounting

Wikipedia: <https://en.wikipedia.org/wiki/Accounting>

https://en.wikipedia.org/wiki/Double-entry_bookkeeping_system

<https://en.wikipedia.org/wiki/Accountant>

https://en.wikipedia.org/wiki/Financial_statement

https://en.wikipedia.org/wiki/Business_valuation

GNUCash

Wikipedia: <https://en.wikipedia.org/wiki/GnuCash>

- Free and Open Source
- Personal Accounting
- Small Business Accounting

Quicken

Wikipedia: <https://en.wikipedia.org/wiki/Quicken>

- Personal Accounting
- USA Version
- International Versions

QuickBooks

Wikipedia: <https://en.wikipedia.org/wiki/QuickBooks>

- Small Business Accounting
- Square integrates with QuickBooks
- <http://www.google.com/enterprise/marketplace/search?query=quickbooks>

Payments

Amazon Payments

Wikipedia: https://en.wikipedia.org/wiki/Amazon_Payments

Apple Pay

Wikipedia: https://en.wikipedia.org/wiki/Apple_Pay

Google Wallet

Wikipedia: https://en.wikipedia.org/wiki/Google_Wallet

- https://en.wikipedia.org/wiki/Google_Checkout

PayPal

Wikipedia: <https://en.wikipedia.org/wiki/PayPal>

Square

Wikipedia: https://en.wikipedia.org/wiki/Square,_Inc.

<https://squareup.com/>

- Square Reader (plugs into headphone jack)
- Square Stand (point of sale)
- Square Register (iOS & Android app)
- Square Market (online store)
- Square Appointments (online scheduling)
- Square Feedback (customer feedback)

- Square Analytics (sales reporting)
- Square Capital (business funding)
- Square Invoices (online invoicing)
- Square Payroll (employee payroll)

Sales

- https://en.wikipedia.org/wiki/Sales_process_engineering
- https://en.wikipedia.org/wiki/Group_information_management
- https://en.wikipedia.org/wiki/Personal_information_management
- https://en.wikipedia.org/wiki/Sales_pipeline
- https://en.wikipedia.org/wiki/Sales_intelligence
- https://en.wikipedia.org/wiki/Sales_force_management_system

See: *CRM*

Customer Relationship Management (CRM)

https://en.wikipedia.org/wiki/Customer_relationship_management

- https://en.wikipedia.org/wiki/Contact_list
- https://en.wikipedia.org/wiki/Address_book
- https://en.wikipedia.org/wiki/Contact_manager
- https://en.wikipedia.org/wiki/Opt-in_email
- https://en.wikipedia.org/wiki/Mailing_list
- https://en.wikipedia.org/wiki/Customer_service#See_also
- https://en.wikipedia.org/wiki/Comparison_of_CRM_systems
- https://en.wikipedia.org/wiki/Customer_intelligence
- https://en.wikipedia.org/wiki/Customer_experience

Business Intelligence

https://en.wikipedia.org/wiki/Business_intelligence

See: *Data Science*

See: *Knowledge Engineering*

https://en.wikipedia.org/wiki/Data_science

- *Science*
 - Open Science
 - Scientific Method
- *Reproducibility*
 - *Ten Simple Rules*
 - *Linked Reproducibility*

Open Source Tools:

- SageMathCloud (SageMath)
- Jupyter Notebook
 - Jupyter Docker Stacks (Conda)
 - Jupyter Extensions
 - *Jupyter and Reproducibility*

datasciencemasters.org

“The Open Source Data Science Masters”

<http://datasciencemasters.org/>

Ten Simple Rules

Homepage: <http://collections.plos.org/ten-simple-rules>

Hashtag: #TenSimpleRules

Twitter: <https://twitter.com/hashtag/TenSimpleRules?src=hash>

#TenSimpleRules for Reproducible Computational Research

“Ten Simple Rules for Reproducible Computational Research”

<http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1003285>

DOI: 10.1371/journal.pcbi.1003285 Featured in PLOS Collections

1. For Every Result, Keep Track of How It Was Produced
 2. Avoid Manual Data Manipulation Steps
 3. Archive the Exact Versions of All External Programs Used
 4. Version Control All Custom Scripts
 5. Record All Intermediate Results, When Possible in Standardized Formats
 6. For Analyses That Include Randomness, Note Underlying Random Seeds
 7. Always Store Raw Data behind Plots
 8. Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected
 9. Connect Textual Statements to Underlying Results
 10. Provide Public Access to Scripts, Runs, and Results
1. For Every Result, Keep Track of How It Was Produced
 - *RDF, JSON-LD* (e.g. *W3C PROV*)
 - Workflow
 - *Knowledge Engineering > Linked Data*
 2. Avoid Manual Data Manipulation Steps
 - Workflow
 - *Continuous Delivery*
 - *Test Automation* (e.g. *Test Driven Development*)
 3. Archive the Exact Versions of All External Programs Used
 - *Jupyter and Reproducibility* (%version_information, %watermark) (should be “Reproducibility and Jupyter Notebook”)
 4. Version Control All Custom Scripts
 - *Revision Control* (e.g. *Distributed Version Control*)
 5. Record All Intermediate Results, When Possible in Standardized Formats
 - *Linked Data* (e.g. *5 Linked Open Data*)

6. For Analyses That Include Randomness, Note Underlying Random Seeds

Python random functions:

```
print(os.environ['PYTHONHASHSEED'])
RANDOMSEED = 1 # /dev/[x]random

import random
random.seed(RANDOMSEED)

import numpy as np
np.random.seed(RANDOMSEED) # Seed
print(np.random.get_state()) # State
np.random.rand(4, 2) # (rows, cols, [...])
np.random.randn(4, 2) # "standard normal" distribution
```

- <http://docs.scipy.org/doc/numpy/reference/routines.random.html#distributions>
-

Python hash randomization and algorithmic determinism:

```
python -R
https://docs.python.org/3/using/cmdline.html#cmdoption-R
PYTHONHASHSEED
https://docs.python.org/3/using/cmdline.html#envvar-PYTHONHASHSEED
```

7. Always Store Raw Data behind Plots

- Or, “Generate all plots from [source-controlled] [transforms-of] raw data”
- ./data
- ./tests/data
- ./nb/data (./notebooks)
- *Data Visualization, Data Visualization Tools*

8. Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected

- *pandas*:
 - <http://pandas.pydata.org/pandas-docs/stable/reshaping.html#reshaping-by-stacking-and-unstacking>
 - <http://pandas.pydata.org/pandas-docs/stable/reshaping.html#combining-with-stats-and-groupby>
- *Schema.org*: <https://schema.org/docs/full.html>
- *SKOS*:

```
http://www.w3.org/TR/skos-reference/
http://www.w3.org/TR/skos-reference/skos.html
```

```
skos:narrower,          skos:narrowerTransitive,          skos:broader          ,
skos:broaderTransitive, [...]
```

- *XKOS*: “An SKOS extension for representing statistical classifications”
<http://rdf-vocabulary.ddialliance.org/xkos.html>
- *RDF Data Cubes*: “The RDF Data Cube Vocabulary”
qb:DataSet, qb:Dimension, qb:ObservationGroup, qb:Slice, [...]
<http://www.w3.org/TR/vocab-data-cube/>

9. Connect Textual Statements to Underlying Results

- *Linked Data: URIs, URLs, #uri-fragments*
- *Turtle / TriG: <> (this document, this named graph)*
- *ReStructuredText*
 - <http://sphinx-doc.org/rest.html#footnotes> #citations #substitutions
 - <https://github.com/yoloseem/awesome-sphinxdoc>
- *Linked Reproducibility: URIs, URLs, #uri-fragments*

10. Provide Public Access to Scripts, Runs, and Results

- *Jupyter and Reproducibility*
- https://en.wikipedia.org/wiki/Comparison_of_source_code_hosting_facilities
 - GitHub: Git
 - BitBucket: Hg, Git
- https://en.wikipedia.org/wiki/Digital_object_identifier
 - <https://en.wikipedia.org/wiki/Figshare>
 - <https://en.wikipedia.org/wiki/Zenodo>
- *Linked Reproducibility*
- *Education Technology > Publishing*

#TenSimpleRules for Creating a Good Data Management Plan

“Ten Simple Rules for Creating a Good Data Management Plan”

<http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004525>

DOI: 10.1371/journal.pcbi.1004525

1. Determine the Research Sponsor Requirements
2. Identify the Data to Be Collected
3. Define How the Data Will Be Organized
4. Explain How the Data Will Be Documented
5. Describe How Data Quality Will Be Assured
6. Present a Sound Data Storage and Preservation Strategy
7. Define the Project’s Data Policies
8. Describe How the Data Will Be Disseminated
9. Assign Roles and Responsibilities

10. Prepare a Realistic Budget

<http://journals.plos.org/plosone/s/data-availability>

> PLOS journals require authors to make all data underlying the findings described in their manuscript fully available without restriction, with rare exception.

Data, Information, Knowledge, & Wisdom

<https://en.wikipedia.org/wiki/Data>

<https://en.wikipedia.org/wiki/Information>

<https://en.wikipedia.org/wiki/Knowledge> (see: *Knowledge Engineering*)

<https://en.wikipedia.org/wiki/Wisdom>

```
# Lead -> Gold
```

- Data is information
- Information is data
- Raw data is not knowledge
- Wisdom compares knowledges

Optimization

https://en.wikipedia.org/wiki/Mathematical_optimization

Find local and global optima (maxima and minima) within an n-dimensional field which may be limited by resource constraints.

```
# Global optima of a 1-dimensional list
points = [10, 20, 100, 20, 10]
global_max, global_min = max(points), min(points)
assert global_max == 100
assert global_min == 10

# Local optima of a 1-dimensional list
sample = points[:1]
local_max, local_min = max(sample), min(sample)
assert local_max == 20
assert local_min == 10

# A 2-dimensional list ...
points = [(-0.5, 0),
          (0, 0.5),
          (0.5, 0),
          (0, -0.5)]
```

- [https://en.wikipedia.org/wiki/Optimization_\(disambiguation\)](https://en.wikipedia.org/wiki/Optimization_(disambiguation))
- <https://en.wikipedia.org/wiki/Metaheuristic>
 - https://en.wikipedia.org/wiki/Receiver_operating_characteristic
 - <http://rayli.net/blog/data/top-10-data-mining-algorithms-in-plain-english/>

- http://scikit-learn.org/stable/tutorial/machine_learning_map/
- https://en.wikipedia.org/wiki/Firefly_algorithm

Smoothies

Data

Inputs, Outputs

Revenue:

```
2014-01-01 1200 CDT $80
2014-01-01 1210 CDT $100
2014-01-01 1500 CDT $20
```

Expenses:

```
2014-01-01 wages $256 ($8/hr * 8hrs * 4 people)
2014-01-01 utilities $100
```

Information

Aggregations, Tendencies

Revenue (gross):

```
2014-01-01 total: $200
```

Expenses:

```
2014-01-01 total: $356
```

Net:

```
2013-01-01 net: -$200
2014-01-01 net: -$156
```

On Mondays, we usually (on (simple) average) make about \$500.

Knowledge

- Positive net revenue is good.
- One customer is worth the world to us.

Wisdom

We could save money by not being open on New Years Day, but, our loyal customers would not be happy about that.

Body Temperature

Data

```
time, body temp, outdoor temp, indoors/outdoors
time, exercise type, intensity, duration
```

Information

Daily temperature variance is about n degrees

Knowledge

- Walking outside when it is warm increases body temperature
- Walking outside when it is cold decreases body temperature
- Exercise increases body temperature

Wisdom

If it's 1745, and body temperature is n degrees above baseline, I'm probably walking outside and it is hot out.

Theory**Science**

<https://en.wikipedia.org/wiki/Science>

https://en.wikipedia.org/wiki/Outline_of_science

<https://en.wikipedia.org/wiki/Category:Science>

Cognitive Biases

https://en.wikipedia.org/wiki/Cognitive_bias

https://en.wikipedia.org/wiki/Heuristics_in_judgment_and_decision-making

https://en.wikipedia.org/wiki/List_of_cognitive_biases

- https://en.wikipedia.org/wiki/Confirmation_bias
- https://en.wikipedia.org/wiki/Post_hoc_ergo_propter_hoc
- https://en.wikipedia.org/wiki/Logical_fallacies#See_also
- https://en.wikipedia.org/wiki/List_of_fallacies
- https://en.wikipedia.org/wiki/Controlling_for_a_variable
 - “distance walked per day”
 - “sports played” (sport, years)

https://en.wikipedia.org/wiki/Critical_thinking

Open Science

https://en.wikipedia.org/wiki/Open_science

- https://en.wikipedia.org/wiki/Open_source
- https://en.wikipedia.org/wiki/Open_standard (*Web Standards, Semantic Web Standards*)
- https://en.wikipedia.org/wiki/Open_data

https://en.wikipedia.org/wiki/Peer_review

- <https://en.wikipedia.org/wiki/Repeatability>

- <https://en.wikipedia.org/wiki/Reproducibility>
- *Reproducibility*

Scientific Method

https://en.wikipedia.org/wiki/Scientific_method

<https://en.wikipedia.org/wiki/Argument>

https://en.wikipedia.org/wiki/Empirical_evidence

<https://en.wikipedia.org/wiki/Hypothesis>

- https://en.wikipedia.org/wiki/Statistical_hypothesis_testing
- https://en.wikipedia.org/wiki/Null_hypothesis
- https://en.wikipedia.org/wiki/Alternative_hypothesis
- https://en.wikipedia.org/wiki/Dependent_and_independent_variables

Reproducibility

https://en.wikipedia.org/wiki/Design_of_experiments

- https://en.wikipedia.org/wiki/Design_of_experiments#Discussion_topics_when_setting_up_an_experimental_design
- <https://en.wikipedia.org/wiki/Repeatability>
- <https://en.wikipedia.org/wiki/Reproducibility>

See: *Jupyter and Reproducibility*

Systematic Review

<https://en.wikipedia.org/wiki/Meta-analysis>

https://en.wikipedia.org/wiki/Systematic_review

Linked Reproducibility

Hashtag: #LinkedReproducibility

Twitter: <https://twitter.com/hashtag/LinkedReproducibility>

Wrdrrdocs: *LinkedReproducibility*

Note: This heading is now merged into a separate page: *LinkedReproducibility*

Math

<https://en.wikipedia.org/wiki/Mathematics>

https://en.wikipedia.org/wiki/Outline_of_mathematics

https://en.wikipedia.org/wiki/Mathematics_education#Methods

- <http://www.iflscience.com/brain/math-gifs-will-help-you-understand-these-concepts-better-your-teacher-ever-did>

Math Courses

- <https://www.khanacademy.org/math/arithmetics>
- <https://www.khanacademy.org/math/pre-algebra>
- <https://www.khanacademy.org/math/algebra-basics>
- <https://www.khanacademy.org/math/algebra>
- <https://www.khanacademy.org/math/basic-geo>
- <https://www.khanacademy.org/math/geometry>
- <https://www.khanacademy.org/math/algebra2>
- <https://www.khanacademy.org/math/trigonometry>
- <https://www.khanacademy.org/math/probability>
- *Linear Algebra*
- *Calculus*
- *Information Theory*
- “Mathematics for Computer Science” (CC-BY-SA 3.0)
https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-spring-2015/readings/MIT6_042JS15_textbook.pdf
- <https://www.khanacademy.org/math/recreational-math>
- <https://www.khanacademy.org/math/competition-math>
- <https://www.class-central.com/subject/math>
- https://en.wikipedia.org/wiki/Kaggle#How_Kaggle_competitions_work

Project Euler

https://en.wikipedia.org/wiki/Project_Euler

<https://projecteuler.net/>

Math Algorithm Problems

Rosalind

Web: <http://rosalind.info/>

Bioinformatics and Data Science Algorithm Problems and Exercises

Mathematical Notation

- https://en.wikipedia.org/wiki/Outline_of_mathematics#Mathematical_notation
- https://en.wikipedia.org/wiki/List_of_mathematical_symbols
- https://en.wikipedia.org/wiki/List_of_mathematical_symbols_by_subject
- https://en.wikipedia.org/wiki/Greek_letters_used_in_mathematics,_science,_and_engineering
- https://en.wikipedia.org/wiki/Latin_letters_used_in_mathematics

See:

- *Knowledge Engineering* > *Symbols*
- *Units* > Units and RDF

LaTeX

Wikipedia: <https://en.wikipedia.org/wiki/LaTeX>

- <https://en.wikipedia.org/wiki/LaTeX#Examples>

MathJax

Wikipedia: <https://en.wikipedia.org/wiki/MathJax>

Docs: <http://docs.mathjax.org/en/latest/tex.html>

MathJax is a Javascript library for displaying *MathML*, *LaTeX*, and *ASCIIMathML* markup in a browser.

- <http://meta.math.stackexchange.com/questions/5020/mathjax-basic-tutorial-and-quick-reference>

MathJax and IPython Notebook / Jupyter Notebook:

- <http://ipython.org/ipython-doc/dev/install/install.html#mathjax>
- <http://nbviewer.ipython.org/github/ipython/ipython/blob/master/examples/Notebook/Typesetting%20Equations.ipynb>
- <http://nbviewer.ipython.org/gist/rpmuller/5920182>

MathML

Wikipedia: <https://en.wikipedia.org/wiki/MathML>

ASCIIMathML

Wikipedia: <https://en.wikipedia.org/wiki/ASCIIMathML>

- *ASCII*
- *MathML*

Information Theory

https://en.wikipedia.org/wiki/Information_theory

[https://en.wikipedia.org/wiki/Entropy_\(information_theory\)](https://en.wikipedia.org/wiki/Entropy_(information_theory))

[https://en.wikipedia.org/wiki/Signal_\(electrical_engineering\)](https://en.wikipedia.org/wiki/Signal_(electrical_engineering))

[https://en.wikipedia.org/wiki/Noise_\(signal_processing\)](https://en.wikipedia.org/wiki/Noise_(signal_processing))

https://en.wikipedia.org/wiki/Signal-to-noise_ratio

https://en.wikipedia.org/wiki/Probability_theory

- <https://www.khanacademy.org/math/probability>

Linear Algebra

https://en.wikipedia.org/wiki/Linear_algebra

- <https://www.khanacademy.org/math/linear-algebra>
- <http://www.ulaff.net/>
- <https://github.com/ULAFF/notebooks/> (Jupyter Notebooks)

Calculus

<https://en.wikipedia.org/wiki/Calculus>

- <https://www.khanacademy.org/math/precalculus>
- <https://www.khanacademy.org/math/differential-calculus>
- <https://www.khanacademy.org/math/integral-calculus>
- <https://www.khanacademy.org/math/multivariable-calculus>
- <https://www.khanacademy.org/math/differential-equations>
- https://en.wikipedia.org/wiki/AP_Calculus
- http://apcentral.collegeboard.com/apc/public/courses/teachers_corner/2178.html
- <http://www.sagemath.org/calctut/>
- <http://boxen.math.washington.edu/home/wdj/teaching/calc1-sage/>
- <http://nbviewer.ipython.org/github/jrjohansson/scientific-python-lectures/blob/master/Lecture-5-Sympy.ipynb>
- <http://scipy-lectures.github.io/advanced/sympy.html#calculus>
- <https://www.class-central.com/subject/calculus-and-mathematical-analysis>

Statistics

<https://en.wikipedia.org/wiki/Statistics>

https://en.wikipedia.org/wiki/Outline_of_statistics

<https://en.wikipedia.org/wiki/Category:Statistics>

- https://en.wikipedia.org/wiki/Notation_in_probability_and_statistics

- http://apcentral.collegeboard.com/apc/public/courses/teachers_corner/2151.html
- <https://www.class-central.com/search?q=statistics>

Parametric Statistics

https://en.wikipedia.org/wiki/Parametric_statistics

Regression Analysis

https://en.wikipedia.org/wiki/Regression_analysis

https://en.wikipedia.org/wiki/Template:Regression_bar

- https://en.wikipedia.org/wiki/Simple_linear_regression
- https://en.wikipedia.org/wiki/Ordinary_least_squares

Nonparametric Statistics

https://en.wikipedia.org/wiki/Nonparametric_statistics

Descriptive Statistics

https://en.wikipedia.org/wiki/Descriptive_statistics

Statistical Inference

https://en.wikipedia.org/wiki/Statistical_inference

- https://en.wikipedia.org/wiki/Statistical_inference#Models_and_assumptions
- https://en.wikipedia.org/wiki/Statistical_inference#Modes_of_inference
- https://en.wikipedia.org/wiki/Multivariate_statistics
 - https://en.wikipedia.org/wiki/Factor_analysis

Causality

<https://en.wikipedia.org/wiki/Causality>

https://en.wikipedia.org/wiki/Correlation_and_dependence

https://en.wikipedia.org/wiki/Correlation_does_not_imply_causation

https://en.wikipedia.org/wiki/Sensitivity_analysis

https://en.wikipedia.org/wiki/Receiver_operating_characteristic

https://en.wikipedia.org/wiki/Post_hoc_ergo_propter_hoc

Analysis

https://en.wikipedia.org/wiki/Data_analysis

https://en.wikipedia.org/wiki/Big_data

https://en.wikipedia.org/wiki/Data_processing#Data_processing_functions

Learning

<https://en.wikipedia.org/wiki/Learning>

- <http://plato.stanford.edu/entries/learning-formal/>
- <http://plato.stanford.edu/entries/logic-inductive/>

<https://en.wikipedia.org/wiki/Autodidacticism>

https://en.wikipedia.org/wiki/Perceptual_learning

[https://en.wikipedia.org/wiki/Pattern_recognition_\(psychology\)#False_pattern_recognition](https://en.wikipedia.org/wiki/Pattern_recognition_(psychology)#False_pattern_recognition)

<https://en.wikipedia.org/wiki/Rhetoric>

https://en.wikipedia.org/wiki/Socratic_method

https://en.wikipedia.org/wiki/Socratic_questioning

https://en.wikipedia.org/wiki/Platonic_dialogue#The_dialogues

<https://en.wikipedia.org/wiki/Dialectic>

<https://en.wikipedia.org/wiki/Dialogue>

[https://en.wikipedia.org/wiki/Perturbation_theory_\(quantum_mechanics\)](https://en.wikipedia.org/wiki/Perturbation_theory_(quantum_mechanics))

https://en.wikipedia.org/wiki/Validated_learning

https://en.wikipedia.org/wiki/Organizational_learning

See: *Knowledge Engineering*

Data Mining

https://en.wikipedia.org/wiki/Data_mining

https://en.wikipedia.org/wiki/Knowledge_extraction

https://en.wikipedia.org/wiki/Extract,_transform,_load

Data Dredging

Wikipedia: https://en.wikipedia.org/wiki/Data_dredging

- !
- *Causality*
- spurious correlations
 - <http://tylervigen.com/spurious-correlations>

Machine Learning

Wikipedia: https://en.wikipedia.org/wiki/Machine_learning

Awesome: <https://github.com/onurakpolat/awesome-bigdata>

Awesome: <https://github.com/josephmisiti/awesome-machine-learning>

https://en.wikipedia.org/wiki/Online_machine_learning

- https://en.wikipedia.org/wiki/Supervised_learning
- https://en.wikipedia.org/wiki/Unsupervised_learning

Deep Learning

Wikipedia: https://en.wikipedia.org/wiki/Deep_learning

- https://en.wikipedia.org/wiki/Biological_neural_network
- https://en.wikipedia.org/wiki/Artificial_neural_network
- https://en.wikipedia.org/wiki/Recurrent_neural_network
- http://www.scholarpedia.org/article/Recurrent_neural_networks
- https://en.wikipedia.org/wiki/Feedforward_neural_network
- https://en.wikipedia.org/wiki/Convolutional_neural_network
- <https://en.wikipedia.org/wiki/Perceptron>
- https://en.wikipedia.org/wiki/Reservoir_computing
- <http://deeplearning.net/>
 - <http://deeplearning.net/deep-learning-research-groups-and-labs/>
 - <http://deeplearning.net/datasets/>
 - http://deeplearning.net/software_links/

Datasets

awesome-public-datasets

<https://github.com/caesar0301/awesome-public-datasets>

- <https://github.com/caesar0301/awesome-public-datasets#search-engines>

Awesome

<https://github.com/bayandin/awesome-awesomeness>

- <https://github.com/onurakpolat/awesome-bigdata>
- <https://github.com/josephmisiti/awesome-machine-learning>
- <https://github.com/caesar0301/awesome-public-datasets>

Tools

ETL

Wikipedia: https://en.wikipedia.org/wiki/Extract,_transform,_load

- https://en.wikipedia.org/wiki/Extract,_transform,_load#Real-life_ETL_cycle

Workflow

- *Scientific Method*
- *Project Management*
- <https://en.wikipedia.org/wiki/Checklist>
- https://en.wikipedia.org/wiki/Scientific_workflow_system
- *Units of measure*
- *I/O Transforms of information(/energy)*

“Data Provenance”, “Data Lineage”

- https://en.wikipedia.org/wiki/Provenance#Data_provenance
- https://en.wikipedia.org/wiki/Data_lineage#Data_Provenance
- *W3C PROV* Provenance Ontology
 - <http://www.w3.org/TR/prov-overview/>
 - <http://www.w3.org/TR/prov-o/>

See:

- *Knowledge Engineering*
- *Tools*
- *Education Technology > Jupyter and Reproducibility*
- *Education Technology > Publishing*

Techniques

Automated Workflows

Standard, Automated Workflows

- *Scientific Method*
- *Reproducibility*
- https://en.wikipedia.org/wiki/Occam's_razor

Q: Is there confirmation bias in starting with e.g. simple regression analysis?

Q: Which factors did we know we were capturing?

5 Linked Open Data

<http://www.w3.org/TR/ld-glossary/#x5-star-linked-open-data>

Publish data on the Web in any format (e.g., PDF, JPEG) accompanied by an explicit Open License (expression of rights).

Publish *structured data* on the Web in a machine-readable format (e.g. *XML*).

Publish structured data on the Web in a documented, non-proprietary data format (e.g. *CSV*, *KML*).

Publish structured data on the Web as RDF (e.g. *Turtle*, *RDFa*, *JSON-LD*, *SPARQL*.)

In your *RDF*, have the identifiers be links (URLs) to useful data sources.

—<http://5stardata.info/>

See: *Knowledge Engineering, Semantic Web Standards*

Data Visualization

Wikipedia: https://en.wikipedia.org/wiki/Data_visualization

Visualizing Data Science

The Data Science Venn Diagram

- <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>
- <http://datascienceassn.org/content/fourth-bubble-data-science-venn-diagram-social-sciences>

Field representations

- <https://github.com/josephmisiti/awesome-machine-learning>
- http://scikit-learn.org/stable/tutorial/machine_learning_map/
- *LODCloud*

Data Visualization Tools

- <https://github.com/vinta/awesome-python#data-visualization>
- <https://github.com/sorrycc/awesome-javascript#data-visualization>
- <https://pandas.pydata.org/pandas-docs/stable/ecosystem.html#visualization>

Matplotlib

Wikipedia: <https://en.wikipedia.org/wiki/Matplotlib>

Homepage: <https://matplotlib.org/>

Src: <https://github.com/matplotlib/matplotlib>

Docs: <https://matplotlib.org/contents.html>

- ref: *Scipy lectures*:
<http://scipy-lectures.github.io/intro/matplotlib/matplotlib.html>
- Scientific-python-lectures:
<http://nbviewer.ipython.org/github/jrjohansson/scientific-python-lectures/blob/master/Lecture-4-Matplotlib.ipynb>
- <http://stanford.edu/~mwaskom/software/seaborn/index.html>
- http://tonysyu.github.com/mpltools/auto_examples/index.html#style-package
- <http://mpld3.github.io/> (Matplotlib + D3.js)
- `conda install matplotlib` (Conda (Anaconda))
- *pandas* plot functions generate matplotlib charts.

Seaborn

Src: <https://github.com/mwaskom/seaborn>

Docs: <http://seaborn.pydata.org/>

Docs: <http://seaborn.pydata.org/examples/>

- “Seaborn is a Python visualization library based on *Matplotlib*. It provides a high-level interface for drawing attractive statistical graphics.”

Mayavi

Wikipedia: <https://en.wikipedia.org/wiki/MayaVi>

Src: <https://github.com/enthought/mayavi>

Docs: <http://docs.enthought.com/mayavi/mayavi/>

- “Mayavi: 3D scientific data visualization and plotting in Python“
- ref:*Scipy lectures*:
https://scipy-lectures.github.io/packages/3d_plotting/

Bokeh

Src: <https://github.com/bokeh/bokeh>

Docs: <https://bokeh.pydata.org/>

VisPy

Homepage: <http://vispy.org/> (*OpenGL*)

Src: <https://github.com/vispy/vispy>

Vega

Homepage: <https://trifacta.github.io/vega/>

Vincent

Src: <https://github.com/wrobstory/vincent>

Plotly

Wikipedia: <https://en.wikipedia.org/wiki/Plotly>

Homepage: <https://plot.ly/>

PyQtGraph

<http://www.pyqtgraph.org/> (*OpenGL*)

qgrid

Src: <https://github.com/quantopian/qgrid>

- (SlickGrid w/ IPython Notebook/ Jupyter Notebook)
- *pandas* support

D3.js

Wikipedia: <https://en.wikipedia.org/wiki/D3.js>

Homepage: <http://d3js.org/>

Three.js

Wikipedia: <https://en.wikipedia.org/wiki/Three.js>

Homepage: <http://threejs.org/>

(*WebGL*)

- Google ARCore Web is built on Three.js
- React VR is built on Three.js

Sigma.js

Homepage: <http://sigma.js.org/>

- *Graphs* in Javascript

CHAPTER 10

See Also

- Tools > *Semantic Web Tools*
- *Art & Design*
- Machine Learning

Wikipedia: https://en.wikipedia.org/wiki/Units_of_measurement

Wikipedia: https://en.wikipedia.org/wiki/System_of_measurement

Mass noun

Wikipedia: https://en.wikipedia.org/wiki/Mass_noun

- e.g. *knowledge* is a mass noun.

RDF and Units

Units of measurement can be expressed with *RDF*, *RDFS*, and *OWL*.

- *QUDT* defines *SI Units* and *Imperial units* with *RDF* and *OWL*.

CSVW and Units

See: *CSVW, Tool Support for Units*

Tool Support for Units

Spreadsheet tools:

- Specify units in each column heading (e.g. "time [[s]]", "speed [[m/s]]")
- Extra column per column for units (e.g. "time", "time unit", "speed", "speed unit")

- *CSVW*: Columns have URIs, which are objects that can have subjects and predicates (for *XSD* types; metadata (author, date, `rdfs:label "time"@en`), and possibly units of measure)

Unit and/or precision-aware calculations:

- `pypi:pint` is a Python library for physical quantities which supports but does not require *NumPy*.

PyPI: <https://pypi.python.org/pypi/pint>

Src: <https://github.com/hgrecco/pint>

Docs: <https://pint.readthedocs.io/en/latest/>

Storage formats:

- [] Spreadsheet formats do not have unit of measure support.
- *CSV* does not have unit of measure support.
- *JSON* does not have unit of measure support.
- *RDF* vocabularies support units of measure:
 - *CSVW* supports units of measure.
 - *JSON-LD* supports units of measure.

Binary Prefixes

Table 11.1: Table of Binary Prefixes (adapted from: https://en.wikipedia.org/wiki/Binary_prefix)

Text	Symbol	Scalar (base 10)	Scalar (base 1000)	Scalar (base 1e1)
kilo	k	1000	1000**1	1e3
mega	M	1000000	1000**2	1e6
giga	G	1000000000	1000**3	1e9
tera	T	1000000000000	1000**4	1e12
peta	P	1000000000000000	1000**5	1e15
exa	E	1000000000000000000	1000**6	1e18
zetta	Z	1000000000000000000000	1000**7	1e21
yotta	Y	1000000000000000000000000	1000**8	1e24

See also: *SI Prefixes*

Metric System

Wikipedia: https://en.wikipedia.org/wiki/Metric_system

Wikipedia: https://en.wikipedia.org/wiki/Outline_of_the_metric_system

SI Units

Wikipedia: https://en.wikipedia.org/wiki/International_System_of_Units

Wikipedia: https://en.wikipedia.org/wiki/SI_base_unit

Wikipedia: https://en.wikipedia.org/wiki/SI_derived_unit

SI Units (*International System of Units*) are the standard units of measurement for almost every country on Earth.

- https://en.wikipedia.org/wiki/SI_derived_unit#Derived_units_with_special_names
- https://en.wikipedia.org/wiki/SI_derived_unit#Examples_of_derived_quantities_and_units
- https://en.wikipedia.org/wiki/Non-SI_units_mentioned_in_the_SI
- *US customary units* do not yet specify *SI Units* because the USA has not yet converted to the *Metric System*.

SI Base Units

Wikipedia: https://en.wikipedia.org/wiki/International_System_of_Units#Base_units

- *metre*
- *kilogram*
- *second*
- *ampere*
- *kelvin*
- *mole*
- *candela*

SI Prefixes

Wikipedia: https://en.wikipedia.org/wiki/Metric_prefix

Table 11.2: Table of SI Prefixes (adapted from: https://en.wikipedia.org/wiki/Template:Common_metric_prefixes and https://en.wikipedia.org/wiki/Binary_prefix)

Text	Symbol	Scaling Factor	Scaling Factor (e)
pico	p	0.000000000001	1e-12
nano	n	0.000000001	1e-9
micro	μ	0.000001	1e-6
milli	m	0.001	1e-3
centi	c	0.01	1e-2
deci	d	0.1	1e-1
(none)	(none)	1	1e0
deca	da	10	1e1
hecto	h	100	1e2
kilo	k	1000	1e3
mega	M	1000000	1e6
giga	G	1000000000	1e9
tera	T	1000000000000	1e12
peta	P	1000000000000000	1e15
exa	E	1000000000000000000	1e18
zetta	Z	1000000000000000000000	1e21
yotta	Y	1000000000000000000000000	1e24

See also: *Binary Prefixes*

SI Distance Units

metre

Wikipedia: <https://en.wikipedia.org/wiki/Metre>

Wikipedia: <https://en.wikipedia.org/wiki/Meter>

Abbr: m

The metre is the length of the path travelled by light in vacuum during a time interval of $1/299792458$ of a second

SI Volume Units

litre

Wikipedia: <https://en.wikipedia.org/wiki/Litre>

Abbr: L

- Conversion: 1 *litre* == $10e3 \text{ m}^3$ (*metres cubed*)
- Conversion: 1 *litre* == 0.2641720523 US *gallon*

SI Mass Units

kilogram

Wikipedia: <https://en.wikipedia.org/wiki/Kilogram>

- A *kilogram* is the mass of TODO
- *gram* is defined in terms of a *kilogram*

gram

Wikipedia: <https://en.wikipedia.org/wiki/Gram>

Abbr: g

- Conversion: 1 *gram* == 1/28 *ounce*
- Conversion: 1 *gram* == 1/1000 kilo- *gram*

SI Time Units

second

Wikipedia: <https://en.wikipedia.org/wiki/Second> #TODO

Abbr: *sec*

Abbr: s

microsecond

Wikipedia: <https://en.wikipedia.org/wiki/Microsecond> #TODO

Abbr: TODO

- Conversion: 1 *microsecond* == 1/100000 TODO of a *second*
- Conversion: 1 *microsecond* == 1e-1000 TODO *seconds*

millisecond

Wikipedia: <https://en.wikipedia.org/wiki/Millisecond> #TODO

Abbr: ms

- Conversion: 1 *millisecond* == 1/1000 of a *second*
- Conversion: 1 *millisecond* == 1e-3 *seconds*

minute

Wikipedia: <https://en.wikipedia.org/wiki/Minute>

Abbr: min

Abbr: m

- Conversion: 1 *minute* == 60 *seconds*

hour

Wikipedia: <https://en.wikipedia.org/wiki/Hour>

Abbr: hr

Abbr: hrs

Abbr: h

- Conversion: 1 *hour* == 60 *minutes*
- Conversion: 1 *hour* == 3600 *seconds*

day

Wikipedia: <https://en.wikipedia.org/wiki/Day>

Abbr: d

- Conversion: 1 *day* == 24 *hours*
- Conversion: 1 *day* == 1440 *minutes*
- Conversion: 1 *day* == 86400 *seconds*
- Rotational definition: 1 *day* == one complete planetary rotation

week

Wikipedia: <https://en.wikipedia.org/wiki/Week>

Abbr: wk

Abbr: wks

Abbr: w

- Conversion: 1 *week* == 7 *days*
- Conversion: 1 *week* == 168 *hours*
- Conversion: 1 *week* == 10080 *minutes*
- Conversion: 1 *week* == 604800 *seconds*

month

Wikipedia: <https://en.wikipedia.org/wiki/Month>

Abbr: mon

Abbr: mons

Abbr: m

- A month contains either 28 (Feb; 29 on a leap year), 30, or 31 *days*.
- https://en.wikipedia.org/wiki/Thirty_days_hath_September
- Two hands of knuckles with peaks and valleys together:
 - start with an outside knuckle
 - up (peak; knuckle) – 31 days
 - down (peak; knuckle) – 30 day (except for February, which has 28/29)

year

Wikipedia: <https://en.wikipedia.org/wiki/Year>

Abbr: yr

Abbr: yrs

Abbr: y

- Conversion: 1 *year* == 365.25 *days* (1 *leap day*/4.0)
- Conversion: 1 *year* == 365 *days*
- Conversion: 1 *year* == 8760 *hours*
- Conversion: 1 *year* == 525600 *minutes*
- Conversion: 1 *year* == 31536000 *seconds*
- Rotational definition: 1 *year* == 1 complete revolution around our singular planetary star: *the sun*.

SI Frequency Units

hertz

Wikipedia: <https://en.wikipedia.org/wiki/Hertz>

Abbr: Hz

- TODO: cycles per time
- TODO: em.py

SI Electric Current Units

ampere

Wikipedia: <https://en.wikipedia.org/wiki/Ampere>

Abbr: A # TODO

Abbr: amp

TODO

volt

Wikipedia: <https://en.wikipedia.org/wiki/Volt>

Abbr: V # TODO

TODO frequency / current relation

SI Power Units

joule

Wikipedia: <https://en.wikipedia.org/wiki/Joule>

watt

Wikipedia: <https://en.wikipedia.org/wiki/Watt>

SI Temperature Units

celsius

Wikipedia: <https://en.wikipedia.org/wiki/Celsius>

Abbr: C

- Water (H₂O) freezes at 0 degrees *celsius*.
- Water (H₂O) boils at 100 degrees *celsius*.
- [] *celsius* / *kelvin* relation

kelvin

Wikipedia: <https://en.wikipedia.org/wiki/Kelvin>

Abbr: K

SI Amount Units

mole

Wikipedia: [https://en.wikipedia.org/wiki/Mole_\(unit\)](https://en.wikipedia.org/wiki/Mole_(unit))

Abbr: mol

SI Luminous Intensity Units

candela

Wikipedia: <https://en.wikipedia.org/wiki/Candela>

Abbr: cd

SI Data Units

bit

Wikipedia: <https://en.wikipedia.org/wiki/Bit>

Abbr: b

A bit can be 1 or 0.

- A bit may indicate *set containment* (e.g. True or False, Black or White) [*Set Theory*]
- A bit may be part of an ordered set of bits which ascribe left-to-right (*little endian*) or right-to-left (*big endian*) place values to each binary digit:

```
1 2 4 8 # little endian
0 1 0 1 # == 0 + 2 + 0 + 8 == 9 (base 10)

8 4 2 1 # big endian
0 1 0 1 # == 0 + 4 + 0 + 1 == 5 (base 10)
```

- See also: *Information Theory* > Shannon bit (*Shannon entropy*)

byte

Wikipedia: <https://en.wikipedia.org/wiki/Byte>

Abbr: B

- Conversion: 1 *byte* == 8 *bits*
- Storage vendors use *powers of ten* (e.g. MB, GB, TB) to describe storage capacity; and also binary prefixes (kilo, mega, giga, tera, peta, exa, zetta).
- Many/most software programs use *powers of two* and binary prefixes (e.g. MiB, GiB, TiB) to describe e.g. partition and file sizes.

- On-disk file sizes are often larger than the file content because of file allocation tables, redundancy, block size and allocation; but may be smaller after compression/deduplication.

Table 11.3: Table of Bytes and Binary Prefixes

unit	derivation	value
nibble	2^{**2} bits	4 bits
byte	2^{**3} bits	8 bits
octet	2^{**3} bits	8 bits
word size	<i>CPU</i> register width (in bits)	8+, 16, 24, 32, 64 bits 32 bits (<i>x86, ARM</i>) 64 bits (<i>x86_64, ARM 64</i>)
kibibyte (KiB)	2^{**10} bytes 1024^{**1} bytes	1024 bytes
kiloyte (KB)	$1e3$ bytes 1000^{**1} bytes	1000 bytes
mebibyte (MiB)	2^{**20} bytes 1024^{**2} bytes	1048576 bytes
megabyte (MB)	$1e6$ bytes 1000^{**2} bytes	1000000 bytes
gibibyte (GiB)	2^{**30} bytes 1024^{**3} bytes	1073741824 bytes
gigabyte (GB)	$1e9$ bytes 1000^{**3} bytes	1000000000 bytes
tebibyte (TiB)	2^{**40} bytes 1024^{**4} bytes	1099511627776 bytes
terabyte (TB)	$1e12$ bytes 1000^{**4} bytes	1000000000000 bytes
pebibyte (PiB)	2^{**50} bytes 1024^{**5} bytes	1125899906842624 bytes
petabyte (PB)	$1e15$ bytes 1000^{**5} bytes	1000000000000000 bytes
exbibyte (EiB)	2^{**60} bytes 1024^{**6} bytes	1152921504606846976 bytes
exabyte (EB)	$1e18$ bytes 1000^{**6} bytes	1000000000000000000 bytes
zebibyte (ZiB)	2^{**70} bytes 1024^{**7} bytes	1180591620717411303424 bytes
zettabyte (ZB)	$1e21$ bytes 1000^{**7} bytes	1000000000000000000000 bytes
yobibyte (YiB)	2^{**80} bytes 1024^{**8} bytes	1208925819614629174706176 bytes
yottabyte (YB)	$1e24$ bytes 1000^{**8} bytes	1000000000000000000000000 bytes

Imperial units

Wikipedia: https://en.wikipedia.org/wiki/Imperial_units

Wikipedia: https://en.wikipedia.org/wiki/English_units

Imperial units may refer to either *English units* (ended in 1824) or *US customary units* (e.g. gram, ounce, gallon, pound, foot, mile).

- The UK (*United Kingdom*) (of which England is a part) now specifies the *Metric System of SI Units*.
- The USA (*United States of America*) *US customary units* still include many *Imperial units*; though *Science* disciplines outside of food, transportation, and weather do now specify the *Metric System of SI Units*.

Imperial Distance Units

inch

Wikipedia: <https://en.wikipedia.org/wiki/Inch>

Abbr: in

- Conversion: 1 *inch* \approx 2.54 *cm* (centi- *meters*)

foot

Wikipedia: <https://en.wikipedia.org/wiki/Foot>

Abbr: ft

- Conversion: 1 *foot* == 12 *inches*

yard

Wikipedia: <https://en.wikipedia.org/wiki/Foot>

Abbr: yd

- Conversion: 1 *yard* == 3 *feet*

mile

Wikipedia: <https://en.wikipedia.org/wiki/Mile>

Abbr: mi

Abbr: m

- Conversion 1 *mile* == 1760 *yards*
- Conversion: 1 *mile* == 5280 *feet*
- Conversion: 1 *mile* == 63360 *inches*

Imperial Volume Units

teaspoon

Wikipedia: <https://en.wikipedia.org/wiki/Teaspoon>

fluid ounce

Wikipedia: https://en.wikipedia.org/wiki/Fluid_ounce

Abbr: fl oz

Abbr: oz

- Serving size: 1 US can of e.g. soda == 12 US *fluid ounce*
- Conversion: 1 *fluid ounce* == 29.573 mL

pint

Wikipedia: <https://en.wikipedia.org/wiki/Pint>

Abbr: pt

- Conversion: 1 US *pint* == 16 US *fluid ounce*
- Conversion: 1 Imperial *pint* == 1.2009499255 US *pints*
- Conversion: 1 US *pint* = 0.85936700738 Imperial *pints*

quart

Wikipedia: <https://en.wikipedia.org/wiki/Quart>

Abbr: qt

- Conversion: 1 US liquid *quart* == 32 *fluid ounce*
- Conversion: 1 *quart* == 4 *cups*
- Conversion: 1 *quart* == 2 *pints*
- Conversion: 1 *quart* == 1/4 *gallon*
- Conversion: 1 US liquid quart == 0.946352946 *litres*

gallon

Wikipedia: <https://en.wikipedia.org/wiki/Gallon>

Abbr: gal

- Conversion: 1 *gallon* == 128 *oz*
- Conversion: 1 *gallon* == 8 *pints*
- Conversion: 1 *gallon* == 3.78541 *litres*

keg

Wikipedia: <https://en.wikipedia.org/wiki/Keg>

- Conversion: 1 *keg* == 124 US *pints*

- Conversion: 1 *keg* == 1984 US *fluid ounce*

Imperial Mass Units

gram

Wikipedia: <https://en.wikipedia.org/wiki/Gram>

Abbr: g

ounce

Wikipedia: <https://en.wikipedia.org/wiki/Ounce>

Abbr: oz

pound

Wikipedia: <https://en.wikipedia.org/wiki/Pound>

Abbr: lb

Abbr: lbm

- Conversion: 1 *pound* == 16 *ounces (oz)*
- Conversion: 1 *pound* == 448 *grams (g)*
- Conversion: 1 *pound* == 0.45359237 kg (kilo- *grams*)

ton

Wikipedia: <https://en.wikipedia.org/wiki/Ton>

- Conversion: 1 *ton* == 2000 *pounds*
- Conversion: 1 kilo- *ton* == 1000 *tons*
- Colloquial: *half ton truck* refers to a minimum towing or hauling capacity; not the curb weight of a vehicle.
- Conversion: 1 *tonne (metric ton)* == 1000 kg == 2204 *pound*
- Conversion: 1 *long ton* == 2240 *pound*
- Conversion: 1 *short ton* == 2000 *pound*

US customary units

Wikipedia: https://en.wikipedia.org/wiki/United_States_customary_units

Common US Fluid Conversions:

- Fluid serving size: 8 *fluid ounce*
- 1 *pint* = 16 *fluid ounce*
- 1 *gallon* = 128 *fluid ounce*
- 1 *keg* = 124 US *pints*

Industry Units

Freight

- US: *pound, ton, cubic inches, mile, second*
- World: *gram, cubic metres (litre), metre, second*

Fuel

- US: *gallon*
- World: *litre*

Wikipedia: https://en.wikipedia.org/wiki/Knowledge_engineering

Wikipedia: https://en.wikipedia.org/wiki/Knowledge_representation_and_reasoning

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Knowledge>

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Graph_theory

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Ontology>

WikipediaCategory: [https://en.wikipedia.org/wiki/Category:Ontology_\(information_science\)](https://en.wikipedia.org/wiki/Category:Ontology_(information_science))

- https://en.wikipedia.org/wiki/Knowledge#Communicating_knowledge
- <https://en.wikipedia.org/wiki/Schema>

Symbols

Wikipedia: <https://en.wikipedia.org/wiki/Symbol>

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Symbols>

- [https://en.wikipedia.org/wiki/Symbol_\(disambiguation\)](https://en.wikipedia.org/wiki/Symbol_(disambiguation))
- https://en.wikipedia.org/wiki/List_of_logic_symbols
- *Art & Design*
 - [...]
- *URI* are symbols:
 - *URN* (term: *URN*)
 - *URL* (term: *URL*)
- Linguistics <https://en.wikipedia.org/wiki/Linguistics>
 - <https://en.wikipedia.org/wiki/Morpheme>

- * <https://en.wikipedia.org/wiki/Phoneme>
- * <https://en.wikipedia.org/wiki/Grapheme>
- <https://en.wikipedia.org/wiki/Word>
- <https://en.wikipedia.org/wiki/Phrase>
- <https://en.wikipedia.org/wiki/Clause>
- [https://en.wikipedia.org/wiki/Sentence_\(linguistics\)](https://en.wikipedia.org/wiki/Sentence_(linguistics))
- <https://en.wikipedia.org/wiki/Paragraph>
- <https://en.wikipedia.org/wiki/Document>
- *Mathematical Notation*
 - *LaTeX*
 - *MathML*
 - *ASCIIMathML*
 - *MathJax*

Character encoding

Wikipedia: https://en.wikipedia.org/wiki/Character_encoding

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Character_encoding

- https://en.wikipedia.org/wiki/Character_encoding#Common_character_encodings

Control Characters

Wikipedia: https://en.wikipedia.org/wiki/Control_character

- *ASCII* Control Characters
 - https://en.wikipedia.org/wiki/Control_character#In_ASCII
- *Unicode* Control Characters
 - https://en.wikipedia.org/wiki/Unicode_control_characters

Warning: Control characters are often significant.

Common security errors involving control characters:

- <https://cwe.mitre.org/data/definitions/74.html>

CWE-74: Improper Neutralization of Special Elements in Output Used by a Downstream Component ('Injection')

- <https://cwe.mitre.org/data/definitions/93.html>

CWE-93: Improper Neutralization of CRLF Sequences ('CRLF Injection')

```
x = "line1_start"  
x2 = "thing\r\n\0line1_end"  
x = x + x2  
x = x + "line2_line2_end\n"  
records = x.splitlines() # ! error
```

- <https://cwe.mitre.org/data/definitions/140.html>

CWE-140: Improper Neutralization of Delimiters

- <https://cwe.mitre.org/data/definitions/141.html>

CWE-141: Improper Neutralization of Parameter/Argument Delimiters

- <https://cwe.mitre.org/data/definitions/142.html>

CWE-142: Improper Neutralization of Value Delimiters

- <https://cwe.mitre.org/data/definitions/143.html>

CWE-143: Improper Neutralization of Record Delimiters

- <https://cwe.mitre.org/data/definitions/144.html>

CWE-144: Improper Neutralization of Line Delimiters

- <https://cwe.mitre.org/data/definitions/145.html>

CWE-145: Improper Neutralization of Section Delimiters

Escape Sequences

Wikipedia: https://en.wikipedia.org/wiki/Escape_sequence

- https://en.wikipedia.org/wiki/Escape_sequences_in_C#Table_of_escape_sequences
- <https://en.wikipedia.org/wiki/CDATA>
 - <https://en.wikipedia.org/wiki/CDATA#Nesting>
- *XML, HTML* & escape sequences:

```
&
```

```
<>/> ” <!-- -> <![CDATA[ ]]>
```

```
# HTML & Templates <p id="{attr}">text</p> # attr='here's one'
```

- Python escape sequences:

```
s = "Here's one"
s = 'Here\'s one'
s = '''Here's one'''
s = 'Here\N{APOSTROPHE}s one'
s = 'Here'"'s"' one'
```

- Bash escape sequences:

```
s1="$Here's one"
s1="${Here}'s one"
s2='${Here}\s one' # ! error
s2='${Here}'"'"s"' one'
s3=""$Here"s one"
s3=""${Here}"'s one"
```

ASCII

Wikipedia: <https://en.wikipedia.org/wiki/ASCII>

ASCII (*American Standard Code for Information Exchange*) defines 128 characters.

- https://en.wikipedia.org/wiki/Teleprinter#Teleprinter_operation
- https://en.wikipedia.org/wiki/Telegraph_code
- Python:

```
from __future__ import print_function
for i in range(0,128):
    print("{0:<3d} {1!r} {1:s}.".format(i, chr(i)))
```

Unicode

Wikipedia: <https://en.wikipedia.org/wiki/Unicode>

Wikipedia: https://en.wikipedia.org/wiki/Unicode_symbols

- https://en.wikipedia.org/wiki/Unicode_symbols#Symbol_block_list
- Entering Unicode Symbols:

https://en.wikipedia.org/wiki/Unicode_input#Hexadecimal_code_input

– https://en.wikipedia.org/wiki/Unicode_input#Hexadecimal_code_input

– – Therefore – u+2234

* X11: ctrl-shift-u 2234

* Vim: ctrl-v u2234

* Python:

• Python 3 Unicode HOWTO: <https://docs.python.org/3/howto/unicode.html>

• Python 2 Unicode HOWTO: <https://docs.python.org/2/howto/unicode.html>

```
c1 = u'' # Python 2.6-3.2, 3.4+
c2 = '' # Python 3.0+
c3 = '\N{THEREFORE}' # howto/unicode#the-string-type glyph name
u1 = unichr(0x2234) # Python 2+
u2 = chr(0x2234) # Python 3.0+
from builtins import chr # Python 2 & 3
u3 = chr(0x2234) # Python 2 & 3
u4 = chr(8756) # int(hex(8756)[2:], 16) == 8756 (0x2234)
chars = [c1, c2, u1, u2, u3, u4]
from operator import eq
assert all((eq(x, chars[0]) for x in chars))
```

- Python and *UTF-8*:

– Python 2 Codecs docs: <https://docs.python.org/2/library/codecs.html>

-
- <https://pymotw.com/2/codecs/>
- e.g. *JSON* with *UTF-8*:

```
# Read an assumed UTF-8 encoded JSON file with Python 2+, 3+
import codecs
with codecs.open('filename.json', encoding='utf8') as file_:
    text = file_.read()
```

Unicode encodings:

- UTF-1
- UTF-5
- UTF-6
- *UTF-8*
- UTF-9, UTF-18
- UTF-16
- UTF-32

UTF-8

Wikipedia: <https://en.wikipedia.org/wiki/UTF-8>

UTF-8 is a *Unicode Character encoding* which can represent all Unicode symbols with 8-bit code units.

- <https://en.wikipedia.org/wiki/UTF-8#Examples>
- In 2015, UTF-8 is the most common web character encoding.
 - *HTML* charset meta attribute:


```
<meta charset="UTF-8">
```
 - *XML* Header:


```
<?xml version="1.0" encoding="UTF-8"?>
```
 - *HTTP* Header:


```
content-type: text/html; charset=UTF-8
```
- Why use UTF-8? <https://www.w3.org/International/questions/qa-choosing-encodings#useunicode>

Logic, Reasoning, and Inference

<https://en.wikipedia.org/wiki/Epistemology>

- <https://en.wikipedia.org/wiki/Truth>
- <https://en.wikipedia.org/wiki/Belief>
- https://en.wikipedia.org/wiki/Theory_of_justification

- *Logic*
 - *Set Theory*
 - *Boolean Algebra*
 - *Many-valued Logic*
 - * *Three-valued Logic*
 - * *Fuzzy Logic*
 - * *Probabilistic Logic*
 - *Propositional Calculus*
 - * *Modus ponens*
 - *Predicate Logic*
 - * *Existential quantification*
 - * *Universal quantification*
 - *Hoare Logic*
 - *First-order Logic*
 - *Description Logic*
- *Reasoning*
 - *Inference*
 - * *Entailment*

Logic

Wikipedia: <https://en.wikipedia.org/wiki/Logic>

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Logic>

- https://en.wikipedia.org/wiki/List_of_logic_symbols
- https://en.wikipedia.org/wiki/Category:Latin_logical_phrases

See:

- *Inference*

Set Theory

Wikipedia: https://en.wikipedia.org/wiki/Set_theory

Boolean Algebra

Wikipedia: https://en.wikipedia.org/wiki/Boolean_algebra

Many-valued Logic

Wikipedia: https://en.wikipedia.org/wiki/Many-valued_logic

Three-valued Logic

Wikipedia: https://en.wikipedia.org/wiki/Three-valued_logic

```

{ True, False, Unknown }
{ T, F, NULL } # SQL
{ T, F, None } # Python
{ T, F, nil } # Ruby
{ 1, 0, -1 } #

```

Fuzzy Logic

Wikipedia: https://en.wikipedia.org/wiki/Fuzzy_logic

Probabilistic Logic

Wikipedia: https://en.wikipedia.org/wiki/Probabilistic_logic

Propositional Calculus

Wikipedia: https://en.wikipedia.org/wiki/Propositional_calculus

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Propositional_calculus

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Theorems_in_propositional_logic

- Premise P
- Conclusion Q

Modus ponens

Wikipedia: https://en.wikipedia.org/wiki/Modus_ponens

- $P \rightarrow Q$ – Premise 1 P1 P_1 (“P sub 1”)
- P – Premise 2 P2 P_2 (“P sub 2”)
- Q – Conclusion Q Q_0 (“Q sub 0”)

Predicate Logic

Wikipedia: https://en.wikipedia.org/wiki/Predicate_logic

- Universe of discourse
- Predicate
 - – There exists – Existential quantifier
 - – For all – Universal quantifier

Existential quantification

Wikipedia: https://en.wikipedia.org/wiki/Existential_quantification

- – “There exists” is the **Existential quantifier** symbol.
- An existential quantifier is true (“holds true”) if there is one (or more) example in which the condition holds true.
- An existential quantifier is *satisfied* by **one** (or more) examples.

Universal quantification

Wikipedia: https://en.wikipedia.org/wiki/Existential_quantification

- – “For all” is the **Universal quantifier** symbol.
- A universal quantification is **disproven by one counterexample** where the condition does not hold true.
 - disproven by one counterexample.

Hoare Logic

Wikipedia: https://en.wikipedia.org/wiki/Hoare_logic

- precondition P
- command C
- postcondition Q

See:

- Propositional Calculus, *Predicate Logic*
- *Given-When-Then*

First-order Logic

Wikipedia: https://en.wikipedia.org/wiki/First-order_logic

First-order logic (*FOL*)

- Terms
 - Variables
 - * x, y, z
 - * x, x_0 (“x subscript 0”, “x sub 0”)
 - Functions
 - * $f(x)$ – function symbol (arity 1)
 - * a – constant symbol (arity 0) ($a()$)
- Formulas (“formulae”)
 - Equality
 - * $=$ – equality
 - Logical Connectives (“unary”, “binary”, sequence/tuple/list)
 - * $\neg, \sim, !$ – negation (unary)
 - * ...
 - * $\wedge, \&\&,$ and – conjunction
 - * $\vee, ||, \text{or}$ – disjunction
 - * $\rightarrow, \Rightarrow,$ – implication
 - * $\leftrightarrow,$ – biconditional
 - * ...
 - * XOR
 - * NAND
 - Grouping Operators
 - * Parentheses ()
 - * Brackets < >
 - Relations
 - * $P(x)$ – predicate symbol ($n_args=1$, arity 1, valence 1)
 - * $R(x)$ – relation symbol ($n_args=1$, arity 1, valence 1)
 - * $Q(x, y)$ – binary predicate/relation symbol ($n_args=2, \dots$)
 - Quantifier Symbols “universe relation”
 - *
 - *
 - ... https://en.wikipedia.org/wiki/First-order_logic

Description Logic

Wikipedia: https://en.wikipedia.org/wiki/Description_logic

Description Logic (*DL*; DLP (Description Logic Programming))

- https://en.wikipedia.org/wiki/Description_logic#Notation
- https://en.wikipedia.org/wiki/Description_logic#Relationship_with_other_logics

Knowledge Base = TBox + ABox

- <https://en.wikipedia.org/wiki/TBox> (Schema: Class/Property Ontology)
- <https://en.wikipedia.org/wiki/ABox> (Facts / Instances)

See:

- *OWL, Entailment*
- *Semantic Web*
- *N3* for => implies

Reasoning

https://en.wikipedia.org/wiki/Deductive_reasoning

<https://en.wikipedia.org/wiki/Category:Reasoning>

https://en.wikipedia.org/wiki/Semantic_reasoner

See: *Description Logic*

Inference

Inference: <https://en.wikipedia.org/wiki/Inference>

- https://en.wikipedia.org/wiki/Rule_of_inference (Logic)
- https://en.wikipedia.org/wiki/List_of_rules_of_inference
- https://en.wikipedia.org/wiki/Category:Statistical_inference (Logic + Math)

Entailment

Wikipedia: <https://en.wikipedia.org/wiki/Entailment>

- <http://www.w3.org/TR/owl2-profiles/#Introduction>

See: *Data Science*

Data Engineering

Data Engineering is about the 5 Ws (who, what, when, where, why) and **how** data are stored.

Who: schema:author @westurner ;

What: schema:name “WRD R&D Documentation”@en ;

When: schema:codeRepository <<https://github.com/wrdrd/docs/commits/master>> ;

Where: schema:codeRepository <<https://github.com/wrdrd/docs>> ;

Why: schema:description “Documentation purposes”@en ;
How: schema:programmingLanguage :ReStructuredText ;
How: schema:runtimePlatform [:Python, :CPython, :Sphinx] ;

- *File Structures*
 - *Git File Structures*
 - * *Git Object*
 - * *Git Reference*
 - * *Git Packfile*
 - *bup*
 - *Torrent file structure*
- *File Locking*
- *Data Structures*
 - *Arrays*
 - *Matrices*
 - *Lists*
 - *Graphs*
 - * *NetworkX*
 - * *DFS*
 - * *BFS*
 - * *Topological Sorting*
 - *Trees*
- *Compression Algorithms*
 - *bzip2*
 - *gzip*
 - *tar*
 - *zip*
- *Hash Functions*
 - *CRC*
 - *MD5*
 - *SHA*
- *Filesystems*
 - *RAID*
 - *MBR*
 - *GPT*

- *LVM*
- *btrfs*
- *ext*
- *FAT*
- *ISO9660*
- *HFS+*
- *NTFS*
- *FUSE*
 - * *SSHFS*
- *Network Filesystems*
 - *Ceph*
 - *CIFS*
 - *DDFS*
 - *GlusterFS*
 - *HDFS*
 - *NFS*
 - *S3*
 - *Swift*
 - *SMB*
 - *WebDAV*
- *Databases*
 - *Object Relational Mapping*
 - *Relation Algebra*
 - *Relational Algebra*
 - *Relational Databases*
 - * *SQL*
 - * *Drizzle*
 - * *MySQL*
 - * *PostgreSQL*
 - * *SQLite*
 - * *Virtuoso*
 - *NoSQL Databases*
 - *Graph Databases*
 - * *Blazegraph*
 - * *Blueprints*

- * *Gremlin*
- * *Neo4j*
- *RDF Triplestores*
- *Distributed Databases*
 - * *Accumulo*
 - * *BigTable*
 - * *Apache Beam*
 - * *Cassandra*
 - * *Hadoop*
 - * *HBase*
 - * *Hive*
 - * *Parquet*
 - * *Presto*
 - * *Spark*
 - *GraphX*
- *Distributed Algorithms*
 - *Distributed Computing Problems*
 - *Non-blocking algorithm*
 - *DHT*
 - *MapReduce*
 - *Paxos*
 - *Raft*
 - *Bulk Synchronous Parallel*
- *Distributed Computing Protocols*
 - *CORBA*
 - *Message Passing*
 - *ESB*
 - *MPI*
 - *XML-RPC*
 - *JSON-RPC*
 - *Avro*
 - *Protocol Buffers*
 - *Thrift*
 - *SOA*
 - * *WS-**

- * *WSDL*
- *JSON-WSP*
- *ROA*
- * *REST*
- *WAMP*
- *Data Grid*
- *Search Engine Indexing*
 - *ElasticSearch*
 - *Haystack*
 - *Lucene*
 - *Nutch*
 - *Solr*
 - *Whoosh*
 - *Xapian*
 - *Information Retrieval*

File Structures

https://en.wikipedia.org/wiki/File_format

[https://en.wikipedia.org/wiki/Record_\(computer_science\)](https://en.wikipedia.org/wiki/Record_(computer_science))

[https://en.wikipedia.org/wiki/Field_\(computer_science\)](https://en.wikipedia.org/wiki/Field_(computer_science))

https://en.wikipedia.org/wiki/Index#Computer_science

- *tar* and *zip* are file structures that have a *manifest* and a *payload*
 - *Filesystems* often have redundant manifests (and/or deduplication according to a hash table manifest with an interface like a *DHT*)
- *Web Standards* and *Semantic Web Standards* which define file structures (and stream protocols):
 - *XML*
 - *RDF (RDF/XML, Turtle, N3, RDFa, JSON-LD)*
 - *JSON (JSON-LD)*
 - *HTTP*

Git File Structures

Git specifies a number of file structures: *Git Objects*, *Git References*, and *Git Packfiles*.

Git implements something like **on-disk shared snapshot objects** with commits, branching, merging, and multi-protocol push/pull semantics: https://en.wikipedia.org/wiki/Shared_snapshot_objects

Git Object

Docs: <https://git-scm.com/book/en/v2/Git-Internals-Git-Objects>

Git Reference

Docs: <https://git-scm.com/book/en/v2/Git-Internals-Git-References>

Git Packfile

Docs: <https://git-scm.com/book/en/v2/Git-Internals-Packfiles>

“Git is a content-addressable *filesystem*“

bup

Homepage: <https://bup.github.io/>

Source: git <https://github.com/bup/bup>

Docs: <https://github.com/bup/bup/blob/master/README.md>

Docs: <https://bup.github.io/man.html>

Docs: <https://github.com/bup/bup/blob/master/DESIGN>

Bup (*backup*) is a backup system based on *git packfiles* and rolling checksums.

[*bup* is a very] efficient backup system based on the *Git Packfile* format, providing fast incremental saves and global deduplication (among and within files, including virtual machine images).

Torrent file structure

A *bittorrent torrent file* is an encoded manifest of tracker, *DHT*, and *web seed URIs*; and segment checksum hashes.

- Like *MPEG-DASH* and *HTTP Live Streaming*, *BitTorrent* downloads file segments over *HTTP*.

See: *BitTorrent*, *Named Data Networking*, *Web Distribution*

File Locking

Wikipedia: https://en.wikipedia.org/wiki/File_locking

File locking is one strategy for synchronization with concurrency and parallelism.

- An auxilliary `<filename>.lock` file is still susceptible to *race conditions*
- C file locking functions: `fcntl`, `lockf`, `flock`

- Python file locking functions: `fcntl.fcntl`, `fcntl.lockf`, `fcntl.flock`: <https://docs.python.org/2/library/fcntl.html>
- To lock a file for all processes with Linux requires a *mandatory file locking* mount option (`mount -o mand`) and per-file `setgid` and `noexec` bits (`chmod g+s, g-s`).
- To lock a file (or a range / record of a file) for all processes with Windows requires no additional work beyond `win32con.LOCKFILE_EXCLUSIVE_LOCK`, `win32file.LockFileEx`, and `win32file.UnlockFileEx`.
- CWE-667: Improper Locking: <https://cwe.mitre.org/data/definitions/667.html#Relationships>
 - https://en.wikipedia.org/wiki/File_locking#Problems
 - https://en.wikipedia.org/wiki/Race_condition
 - CWE-833: Deadlock
 - <https://cwe.mitre.org/data/definitions/833.html>
 - <https://en.wikipedia.org/wiki/Deadlock>

Data Structures

Wikipedia: https://en.wikipedia.org/wiki/Data_structure

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Data_structures

Docs: https://en.wikipedia.org/wiki/List_of_data_structures

- http://rosettacode.org/wiki/Category:Programming_Tasks
 - http://rosettacode.org/wiki/Greatest_common_divisor
 - http://rosettacode.org/wiki/Go_Fish

Arrays

Wikipedia: https://en.wikipedia.org/wiki/Array_data_structure

Docs: https://en.wikipedia.org/wiki/List_of_data_structures#Arrays

An array is a data structure for unidimensional data.

- Arrays must be resized when data grows beyond the initial shape of the array.
- Sparse arrays are sparsely allocated.
- A multidimensional array is said to be a *matrix*.

Matrices

Wikipedia: [https://en.wikipedia.org/wiki/Matrix_\(computer_science\)](https://en.wikipedia.org/wiki/Matrix_(computer_science))

A matrix is a data structure for multidimensional data; a multidimensional *array*.

Lists

Wikipedia: https://en.wikipedia.org/wiki/Linked_list

Docs: https://en.wikipedia.org/wiki/List_of_data_structures#Lists

A list is a data structure with nodes that link to a next and/or previous node.

Graphs

Wikipedia: [https://en.wikipedia.org/wiki/Graph_\(abstract_data_type\)](https://en.wikipedia.org/wiki/Graph_(abstract_data_type))

Wikipedia: [https://en.wikipedia.org/wiki/Graph_\(mathematics\)](https://en.wikipedia.org/wiki/Graph_(mathematics))

Wikipedia: https://en.wikipedia.org/wiki/Graph_theory

Docs: https://en.wikipedia.org/wiki/Conceptual_graph

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Graphs>

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Graph_data_structures

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Graph_theory

A graph is a *system* of nodes connected by edges; an abstract data type for which there are a number of suitable data structures.

- A node has edges.
- An edge connects nodes.
- Edges of **directed graphs** flow in only one direction; and so require two edges with separate attributes (e.g. ‘magnitude’, ‘scale’)

Wikipedia: https://en.wikipedia.org/wiki/Directed_graph

- Edges of an **undirected graph** connect nodes in both directions (with the same attributes).

Wikipedia: [https://en.wikipedia.org/wiki/Graph_\(mathematics\)#Undirected_graph](https://en.wikipedia.org/wiki/Graph_(mathematics)#Undirected_graph)

- Graphs and *Trees* are **traversed** (or *walked*); according to a given algorithm (e.g. *DFS*, *BFS*).
- Graph nodes can be listed in many different *orders* (or with a given *ordering*):
 - Preorder
 - Inorder
 - Postorder
 - Level-order
- There are many *data structure* representations for *Graphs*.
- There are many data serialization/marshalling formats for graphs:
 - Graph edge lists can be stored as adjacency *matrices*.
 - *NetworkX* supports a number of graph storage formats.

- *RDF* is a *standard semantic web Linked Data* format for *Graphs*.
- *JSON-LD* is a *standard semantic web Linked Data* format for *Graphs*.
- There are many *Graph Databases* and *RDF Triplestores* for storing graphs.
- A cartesian product has an interesting graph representation. (See *Compression Algorithms*)

NetworkX

Wikipedia: <https://en.wikipedia.org/wiki/NetworkX>

Homepage: <https://networkx.github.io/>

Source: git <https://github.com/networkx/networkx>

Docs: <https://networkx.readthedocs.io/en/latest/>

Docs: <https://networkx.readthedocs.io/en/latest/tutorial/>

Docs: <https://networkx.readthedocs.io/en/latest/reference/classes.html>

Docs: <https://networkx.readthedocs.io/en/latest/reference/algorithms.html>

NetworkX is an *Open Source* graph algorithms library written in Python.

DFS

Wikipedia: https://en.wikipedia.org/wiki/Depth-first_search

DFS (*Depth-first search*) is a *graph* traversal algorithm.

```
# Given a tree:
1
  1.1
  1.2
2
  2.1
  2.2

# BFS:
[1, 1.1, 1.2, 2, 2.1, 2.2]
```

See also: *Bulk Synchronous Parallel*, *Firefly Algorithm*

BFS

Wikipedia: https://en.wikipedia.org/wiki/Breadth-first_search

BFS (*Breadth-first search*) is a *graph* traversal algorithm.

```
# Given a tree:
1
  1.1
  1.2
2
```

```

2.1
2.2

# BFS:
1, 2, 1.1, 1.2, 2.1, 2.2

```

- [] BFS and *Bulk Synchronous Parallel*

Topological Sorting

Wikipedia: https://en.wikipedia.org/wiki/Topological_sorting

A DAG (*directed acyclic graph*) has a topological sorting, or is topologically sorted.

- The unix `tsort` utility does a topological sorting of a space and newline delimited list of edge labels:

```

$ tsort --help
Usage: tsort [OPTION] [FILE]
Write totally ordered list consistent with the partial ordering in FILE.
With no FILE, or when FILE is -, read standard input.

    --help      display this help and exit
    --version   output version information and exit

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
For complete documentation, run: info coreutils 'tsort invocation'

$ echo -e '1 2\n2 3\n3 4\n2 a' | tsort
1
2
a
3
4

```

- Installing a set of packages with dependencies is a topological sorting problem; plus e.g. version and platform constraints (as solvable with a SAT constraint satisfaction solver (see `conda` (`pypi:pycosat`)))
- A topological sorting can identify the “root” of a **directed acyclic graph**.
 - *Information gain* can be useful for less discrete problems.

Trees

Wikipedia: https://en.wikipedia.org/wiki/Tree_data_structure

Docs: http://rosettacode.org/wiki/Tree_traversal

A tree is a directed *graph*.

- A tree is said to have branches and leaves; or just nodes.

There are many types of and applications for trees:

- https://en.wikipedia.org/wiki/List_of_data_structures#Trees
- <https://en.wikipedia.org/wiki/B-tree>

- <https://en.wikipedia.org/wiki/Trie>
- https://en.wikipedia.org/wiki/Abstract_syntax_tree
- https://en.wikipedia.org/wiki/Parse_tree
- https://en.wikipedia.org/wiki/Decision_tree
- <https://en.wikipedia.org/wiki/Minmax>
- https://en.wikipedia.org/wiki/Database_index
- Search: Indexing, Lookup

Compression Algorithms

bzip2

Wikipedia: <https://en.wikipedia.org/wiki/Bzip2>

File Extension: `.bz2`

Homepage: <http://bzip.org/>

bzip2 is an *Open Source* lossless compression algorithm based upon the Burrows–Wheeler algorithm.

- bzip2 is usually slower than *gzip* or *zip*, but more space efficient

gzip

Wikipedia: <https://en.wikipedia.org/wiki/Gzip>

Homepage: <https://www.gnu.org/software/gzip/>

File Extension: `.gz`

Source: <http://ftp.gnu.org/gnu/gzip/>

Docs: <https://www.gnu.org/software/gzip/manual/>

Docs: <https://www.gnu.org/software/gzip/manual/gzip.html>

gzip is a compression algorithm based on DEFLATE and LZ77.

- gzip is similar to *zip*, in that both are based upon DEFLATE

tar

Wikipedia: [https://en.wikipedia.org/wiki/Tar_\(computing\)](https://en.wikipedia.org/wiki/Tar_(computing))

File Extension: `.tar`

tar is a file archiving format for storing a manifest of records of a set of files with paths and attributes at the beginning of the actual files all concatenated into one file.

- TAR = (table of contents + data stream)
- `.tar.gz` is *tar* + *gzip*
- `.tar.bz2` is *tar* + *bzip2*

TAR and *gzip* or *bzip2* can be streamed over SSH:

```
# https://unix.stackexchange.com/a/95994
tar czf - . | ssh remote "( cd ~/ ; cat > file.tar.gz )"
tar bzf - . | ssh remote "( cd ~/ ; cat > file.tar.bz2 )"
```

See also: *zip* (windows)

zip

Wikipedia: [https://en.wikipedia.org/wiki/Zip_\(file_format\)](https://en.wikipedia.org/wiki/Zip_(file_format))

zip is a lossless file archive compression

Hash Functions

Wikipedia: https://en.wikipedia.org/wiki/Hash_function

Wikipedia: https://en.wikipedia.org/wiki/Cryptographic_hash_function

Hash functions (or *checksums*) are one-way functions designed to produce uniquely identifying identifiers for blocks or whole files in order to verify data *Integrity*.

- A *hash* is the output of a hash function.
- In Python, `dict` keys must be *hashable* (must have a `__hash__` method).
- In Java, Scala, and many other languages `dicts` are called `HashMaps`.
- *MD5* is a checksum algorithm.
- *SHA* is a group of checksum algorithms.

CRC

Wikipedia: https://en.wikipedia.org/wiki/Cyclic_redundancy_check

A CRC (*Cyclical Redundancy Check*) is a hash function for error detection based upon an extra *check value*.

- *Hard Drives* and *SSDs* implement CRCs.
- *Ethernet* implements CRCs.

MD5

Wikipedia: <https://en.wikipedia.org/wiki/MD5>

MD5 is a 128-bit hash function which is now broken, and deprecated in favor of *SHA-2* or better.

```
md5
md5sums
```

SHA

Wikipedia: https://en.wikipedia.org/wiki/Secure_Hash_Algorithm

- SHA-0 – 160 bit (retracted 1993)
- SHA-1 — 160 bit (deprecated 2010)
- SHA-2 — sha-256, sha-512
- SHA-3 (2012)

```
shasum
shasum -a 1
shasum -a 224
shasum -a 256
shasum -a 384
shasum -a 512
shasum -a 512224
shasum -a 512256
```

Filesystems

Wikipedia: https://en.wikipedia.org/wiki/File_system

Filesystems (*file systems*) determine how files are represented in a persistent physical medium.

- On-disk filesystems determine where and how redundantly data is stored
- On-disk filesystems: *ext*, *btrfs*, *FAT*, *NTFS*, *HFS+*
- *Network Filesystems* link disk storage pools with other resources (e.g. *NFS*, *Ceph*, *GlusterFS*)

RAID

Wikipedia: <https://en.wikipedia.org/wiki/RAID>

RAID (*redundant array of independent disks*) is set of configurations for *Hard Drives* and *SSDs* to *stripe* and/or *mirror* with *parity*.

```
RAID 0 -- striping,          -,          no parity ... throughput
RAID 1 -- no striping, mirroring,          no parity ...
RAID 2 -- bit striping,     -,          no parity ... legacy
RAID 3 -- byte striping,   -,          dedicated parity ... uncommon
RAID 4 -- block striping,  -,          dedicated parity
RAID 5 -- block striping,  -,          distributed parity ... min. 3; n-1 rebuild
RAID 6 -- block striping,  -, 2x distributed parity
```

RAID Implementations:

- RAID may be implemented by a physical controller with multiple drive connectors.
- RAID may be implemented as a BIOS setting.
- RAID may be implemented with software e.g. *LVM*, *btrfs*.

- <https://en.wikipedia.org/wiki/RAID#Software-based>
- https://en.wikipedia.org/wiki/RAID#Firmware-_and_driver-based (“*fake RAID*”)
- **Data Scrubbing** Data scrubbing is a technique for checking for inconsistencies between redundant copies of data

Data scrubbing is routinely part of RAID (with *mirrors* and/or *parity* bits).

https://en.wikipedia.org/wiki/Data_scrubbing

MBR

Wikipedia: https://en.wikipedia.org/wiki/Master_boot_record

MBR (*Master Boot Record*) is a boot record format and a file partition scheme.

- DOS and Windows use MBR partition tables.
- Many/most UNIX variants support MBR partition tables.
- Linux supports MBR partition tables.
- Most PCs since 1983 boot from MBR partition tables.
- When a PC boots, it reads the MBR on the first configured drive in order to determine where to find the boot-loader.

GPT

Wikipedia: https://en.wikipedia.org/wiki/GUID_Partition_Table

GPT (*GUID Partition Table*) is a boot record format and a file partition scheme wherein partitions are assigned GUIDs (*Globally Unique Identifiers*).

- OSX uses GPT partition tables.
- Linux supports GPT partition tables.
- https://en.wikipedia.org/wiki/GUID_Partition_Table#UNIX_and_Unix-like_operating_systems

LVM

Wikipedia: [https://en.wikipedia.org/wiki/Logical_Volume_Manager_\(Linux\)](https://en.wikipedia.org/wiki/Logical_Volume_Manager_(Linux))

Homepage: <https://www.sourceware.org/lvm2/>

Source: <ftp://sources.redhat.com/pub/lvm2/>

Docs: <https://www.sourceware.org/dm/>

Docs: <http://www.tldp.org/HOWTO/LVM-HOWTO/index.html>

Docs: <http://www.tldp.org/HOWTO/LVM-HOWTO/anatomy.html>

LVM (*Logical Volume Manager*) is an *Open Source* software disk abstraction layer with snapshotting, copy-on-write, online resize and allocation and a number of additional features.

- In LVM, there are *Volume Groups* (VG), *Physical Volumes* (PV), and *Logical Volumes* (LV).

- LVM can do striping and high-availability software *RAID*.
- LVM and `device-mapper` are now part of the Linux kernel tree (the LVM linux kernel modules are built and included with most distributions' default kernel build).
- LVM Logical Volumes can be resized online (without e.g. rebooting to busybox or a LiveCD); but many *Filesystems* support only online grow (and not online shrink).
- There is feature overlap between *LVM* and *btrfs* (pooling, snapshotting, copy-on-write).

btrfs

Wikipedia: <https://en.wikipedia.org/wiki/Btrfs>

Homepage: https://btrfs.wiki.kernel.org/index.php/Main_Page

Source: https://btrfs.wiki.kernel.org/index.php/Btrfs_source_repositories

Source: git [git://git.kernel.org/pub/scm/linux/kernel/git/mason/btrfs-progs.git](https://git.kernel.org/pub/scm/linux/kernel/git/mason/btrfs-progs.git)

Docs: https://btrfs.wiki.kernel.org/index.php/Getting_started#Basic_Filesystem_Commands

Docs: https://btrfs.wiki.kernel.org/index.php/Problem_FAQ

Docs: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Storage_Administration_Guide/ch-btrfs.html

Docs: <https://wiki.archlinux.org/index.php/Btrfs>

Docs: <https://help.ubuntu.com/community/btrfs>

btrfs (*B-tree filesystem*) is an *Open Source* pooling, snapshotting, checksumming, deduplicating, union mounting copy-on-write on-disk Linux filesystem.

ext

Wikipedia: <https://en.wikipedia.org/wiki/Ext2>

Wikipedia: <https://en.wikipedia.org/wiki/Ext3>

Wikipedia: <https://en.wikipedia.org/wiki/Ext4>

ext2, ext3, and ext4 are the ext (*extended filesystem*) *Open Source* on-disk filesystems.

- ext filesystems are the default filesystems of many Linux distributions.
- windows machines can access ext2, ext3, and ext4 filesystems with ext2explore and ext2fsd.
- OSX machines can access ext2, ext3, and ext4 filesystems with OSXFuse and FUSE-EXT2.

FAT

Wikipedia: https://en.wikipedia.org/wiki/File_Allocation_Table

FAT is a group of on-disk filesystem standards.

- FAT is used on cross-platform USB drives.
- FAT is found on older Windows and DOS machines.
- FAT12, FAT16, and FAT32 are all FAT filesystem standards.

- FAT32 has a maximum filesize of 4GB and a maximum volume size of 2 TB.
- Windows machines can read and write FAT partitions.
- OSX machines can read and write FAT partitions.
- Linux machines can read and write FAT partitions.

ISO9660

Wikipedia: https://en.wikipedia.org/wiki/ISO_9660

FileExt: `.iso`

ISO9660 is an *ISO* standard for *disc drive* images which specifies a standard for booting from a filesystem image.

- Many Operating System distributions are distributed as *ISO9660* `.iso` files.
- ISO9660 and Linux:
 - An ISO9660 ISO can be *loop mounted*:

```
mount -o loop,ro -t iso9660 ./path/to/file.iso /mnt/cdrom
```

- An ISO8660 CD can be *mounted*:

```
mount -o ro -t iso9660 /dev/cdrom /mnt/cdrom
```

- Most CD/DVD burning utilities support ISO9660 `.iso` files.
- ISO9660 is useful in that it specifies how to encode the boot sector (*El Torito*) and partition layout.
- Nowadays, ISO9660 `.iso` files are often converted to raw drive images and written to bootable *USB* Mass Storage devices (e.g. to write a install / recovery disq for Debian, Ubuntu, Fedora, Windows)

HFS+

Wikipedia: https://en.wikipedia.org/wiki/HFS_Plus

HFS+ (*Hierarchical Filesystem*) or *Mac OS Extended*, is the filesystem for Mac OS 8.1+ and OSX.

- HFS+ is required for OSX and Time Machine.
 - <http://www.cnet.com/how-to/the-best-ways-to-format-an-external-drive-for-windows-and-mac/>
- Windows machines can access HFS+ partitions with: HFSExplorer (free, Java), Paragon HFS+ for Windows, or MacDrive
 - <http://www.makeuseof.com/tag/4-ways-read-mac-formatted-drive-windows/>
- Linux machines can access HFS+ partitions with `hfsprogs` (`apt-get install hfsprogs`, `yum install hfsprogs`).

NTFS

Wikipedia: <https://en.wikipedia.org/wiki/NTFS>

NTFS is a proprietary journaling filesystem.

- Windows machines since Windows NT 3.1 and Windows XP default to NTFS filesystems.
- Non-Windows machines can access NTFS partitions through NTFS-3G: <https://en.wikipedia.org/wiki/NTFS-3G>

FUSE

Wikipedia: https://en.wikipedia.org/wiki/Filesystem_in_Userspace

Homepage: <http://fuse.sourceforge.net/>

Download: <http://sourceforge.net/projects/fuse/files/fuse-2.X/>

Source: git <http://git.code.sf.net/p/fuse/fuse>

Docs: <http://fuse.sourceforge.net/doxygen/index.html>

Docs: <http://sourceforge.net/p/fuse/wiki/FileSystems/>

Docs: <http://sourceforge.net/p/fuse/wiki/LanguageBindings/>

Docs: <http://sourceforge.net/p/fuse/wiki/OperatingSystems/>

FUSE (*Filesystem in Userspace*) is a userspace filesystem API for implementing filesystems in userspace.

- FUSE support is included in the Linux kernel since 2.6.14.
- FUSE is available for most POSIX platforms.

Interesting FUSE implementations:

- PyFilesystem is a Python *language api* interface which supports *FUSE*: <http://docs.pyfilesystem.org/en/latest/>
- There are FUSE bindings for *Hadoop HDFS*.
- *Ceph* can be mounted with/over/through *FUSE*.
- *GlusterFS* can be mounted with/over/through *FUSE*.
- *NTFS-3G* mounts volumes with *FUSE*.
- virtualbox-fuse supports mounting of virtualbox VDI images with FUSE.
- *SSHFS*, GitFS, GmailFS, GdriveFS, WikipediaFS and Gnome GVFS are all FUSE filesystems.

SSHFS

Wikipedia: <https://en.wikipedia.org/wiki/SSHFS>

Homepage: <http://fuse.sourceforge.net/sshfs.html>

Download: <http://sourceforge.net/projects/fuse/files/sshfs-fuse/>

Source: git <http://git.code.sf.net/p/fuse/sshfs>

Docs: <https://wiki.archlinux.org/index.php/Sshfs>

Docs: <https://help.ubuntu.com/community/SSHFS>

Docs: <https://github.com/osxfuse/osxfuse/wiki/SSHFS>

SSHFS is a *FUSE* filesystem for mounting remote directories over SSH.

Network Filesystems

Wikipedia: https://en.wikipedia.org/wiki/Network_filesystem

Ceph

Wikipedia: [https://en.wikipedia.org/wiki/Ceph_\(software\)](https://en.wikipedia.org/wiki/Ceph_(software))

Homepage: <http://ceph.com/>

Download: <http://ceph.com/resources/downloads/>

Source: git <https://github.com/ceph/ceph>

Docs: <http://ceph.com/docs/master/>

Docs: <http://ceph.com/docs/master/rados/>

Docs: <http://ceph.com/docs/master/radosgw/>

Docs: <http://ceph.com/docs/master/radosgw/s3/>

Docs: <http://ceph.com/docs/master/radosgw/swift/>

Docs: <http://ceph.com/docs/master/radosgw/keystone/>

Docs: <http://ceph.com/docs/master/rbd/rbd-openstack/>

Ceph is an *Open Source* network filesystem (a *distributed database* for files with attributes like owner, group, permissions) written in C++ and Perl which runs over top of one or more on-disk filesystems.

- Ceph Block Device (*rbd*) – striping, caching, snapshots, copy-on-write, kvm, libvirt, OpenStack Cinder block storage
- Ceph Filesystem (*cephfs*) – POSIX *filesystem* with *FUSE*, *NFS*, *CIFS*, and *HDFS* APIs
- Ceph Object Gateway (*radosgw*) – *RESTful API*, *Amazon AWS S3 API*, OpenStack Swift API, OpenStack Keystone authentication

CIFS

CIFS (*Common Internet File System*) is a centralized network filesystem protocol.

- Samba `smbd` is one implementation of a *CIFS* network file server.

DDFS

DDFS (*Disco Distributed File System*) is a distributed network filesystem written in Python and C.

- DDFS is like a python implementation of *HDFS* (which is written in Java).

GlusterFS

Wikipedia: <https://en.wikipedia.org/wiki/GlusterFS>

Homepage: <http://www.gluster.org/>

Project: <https://forge.gluster.org/glusterfs-core>

Source: git <https://git.forge.gluster.org/glusterfs-core/glusterfs.git>

Docs: <https://gluster.readthedocs.io/en/latest/>

Docs: <https://gluster.readthedocs.io/en/latest/Quick-Start-Guide/Quickstart/>

Docs: https://gluster.readthedocs.io/en/latest/Install-Guide/Setup_virt/

Docs: https://gluster.readthedocs.io/en/latest/Install-Guide/Setup_Bare_metal/

Docs: https://gluster.readthedocs.io/en/latest/Install-Guide/Setup_aws/

Docs: <https://gluster.readthedocs.io/en/latest/Administrator%20Guide/GlusterFS%20Cinder/>

Tcp ports: 111, 24007, 24008, 24009, 24010, 24011, 38465:38469

GlusterFS is an *Open Source* network filesystem (a *distributed database* for files with attributes like owner, group, permissions) which runs over top of one or more on-disk filesystems.

- GlusterFS can serve volumes for OpenStack Cinder block storage

HDFS

Wikipedia: https://en.wikipedia.org/wiki/Apache_Hadoop#HDFS

HDFS (*Hadoop Distributed File System*) is an *Open Source* distributed network filesystem.

- HDFS runs code next to data; rather than streaming data through code across the network.
- HDFS is especially suitable for *MapReduce*-style distributed computation.
- Apache *Hadoop* works with files stored over HDFS, FTP, *S3*, WASB (Azure)
- There are HDFS *language apis* for many languages: Java, Scala, Go, Python, Ruby, Perl, Haskell, C++
- Mesos can manage distributed HDFS grids.
- *ElasticSearch*
- It's possible to configure a *Jenkins Continuous Integration* cluster as *Hadoop* cluster.
- Many databases support storage over HDFS (*HBase*, *Cassandra*, *Accumulo*, *Spark*)
- *Ceph* can now serve files over *HDFS*.
- HDFS can be mounted as a *FUSE* filesystem (e.g. with Linux).
- HDFS can be accessed from the commandline with the Hadoop *FS shell*: <https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/FileSystemShell.html>
- HDFS can be browsed with hdfs-du: <https://github.com/twitter/hdfs-du>

NFS

Wikipedia: <https://en.wikipedia.org/wiki/NFS>

NFS (*Network File System* #TODO) is an *Open Source* centralized network filesystem.

S3

- *Amazon AWS S3*
- OpenStack Swift
- *Ceph*
- *GlusterFS*

Swift

- OpenStack Swift
- *Ceph*
- *GlusterFS*

SMB

Wikipedia: https://en.wikipedia.org/wiki/Server_Message_Block

SMB (*Server Message Block*) is a centralized network filesystem.

- SMB has been superseded by *CIFS*.

WebDAV

Wikipedia: <https://en.wikipedia.org/wiki/WebDAV>

Standard: <https://tools.ietf.org/html/rfc2518>

Standard: <https://tools.ietf.org/html/rfc4918>

WebDAV (*Web Distributed Authoring and Versioning*) is a network filesystem protocol built with *HTTP*.

- WebDAV specifies a number of unique *HTTP* methods:
 - PROPFIND (*ls*, *stat*, *getfacl*),
 - PROPPATCH (*touch*, *setfacl*)
 - MKCOL (*mkdir*)
 - COPY (*cp*)
 - MOVE (*mv*)
 - LOCK (*File Locking*)
 - UNLOCK ()

Databases

Wikipedia: <https://en.wikipedia.org/wiki/Database>

- https://en.wikipedia.org/wiki/Database_schema

- https://en.wikipedia.org/wiki/Create,_read,_update_and_delete
- <https://en.wikipedia.org/wiki/CRUD>
- <https://en.wikipedia.org/wiki/ACID>
- https://en.wikipedia.org/wiki/Query_plan
- https://en.wikipedia.org/wiki/Database_index
- *Search Engine Indexing*
- https://en.wikipedia.org/wiki/Category:Database_software_comparisons
 - <http://db-engines.com/en/ranking>

Object Relational Mapping

Wikipedia: https://en.wikipedia.org/wiki/Object-relational_mapping

- https://en.wikipedia.org/wiki/Data_mapper_pattern
- https://en.wikipedia.org/wiki/Active_record_pattern

https://en.wikipedia.org/wiki/Object-relational_impedance_mismatch

- https://en.wikipedia.org/wiki/List_of_object-relational_mapping_software

Relation Algebra

Wikipedia: https://en.wikipedia.org/wiki/Relation_algebra

- https://en.wikipedia.org/wiki/Relation_algebra#Expressing_properties_of_binary_relations_in_RA

See: *Relational Algebra*

Relational Algebra

Wikipedia: https://en.wikipedia.org/wiki/Relational_algebra

- [https://en.wikipedia.org/wiki/Projection_\(relational_algebra\)](https://en.wikipedia.org/wiki/Projection_(relational_algebra))
- https://en.wikipedia.org/wiki/Relational_algebra#Joins_and_join-like_operators
- https://en.wikipedia.org/wiki/Relational_algebra#Common_extensions

See: *Relation Algebra, Relational Databases*

Relational Databases

Wikipedia: https://en.wikipedia.org/wiki/Relational_database

https://en.wikipedia.org/wiki/Relational_model

Relational Algebra

https://en.wikipedia.org/wiki/Database_normalization

- https://en.wikipedia.org/wiki/Referential_integrity
- https://en.wikipedia.org/wiki/Functional_dependency
- https://en.wikipedia.org/wiki/Dangling_pointer
- https://en.wikipedia.org/wiki/Natural_key
- https://en.wikipedia.org/wiki/Surrogate_key
- https://en.wikipedia.org/wiki/Foreign_key
- <https://en.wikipedia.org/wiki/Denormalization>

https://en.wikipedia.org/wiki/Relational_database_management_system

- https://en.wikipedia.org/wiki/Comparison_of_relational_database_management_systems
- *MySQL*
- *PostgreSQL*
- *SQLite*
- *Virtuoso*
- <http://db-engines.com/en/ranking/relational+dbms>

What doesn't SQL do?

- *RDF*, *OWL*
- <https://en.wikipedia.org/wiki/OLAP>

SQL

Wikipedia: <https://en.wikipedia.org/wiki/SQL>

- [https://en.wikipedia.org/wiki/Null_\(SQL\)#Comparisons_with_NULL_and_the_three-valued_logic_.283VL.29](https://en.wikipedia.org/wiki/Null_(SQL)#Comparisons_with_NULL_and_the_three-valued_logic_.283VL.29)
- [https://en.wikipedia.org/wiki/Join_\(SQL\)](https://en.wikipedia.org/wiki/Join_(SQL))
- https://en.wikipedia.org/wiki/SQL_injection
- <http://cwe.mitre.org/top25/#CWE-89> (#1 Most Prevalent Dangerous Security Error (2011))

See: *Object Relational Modeling*

Drizzle

Wikipedia: [https://en.wikipedia.org/wiki/Drizzle_\(database_server\)](https://en.wikipedia.org/wiki/Drizzle_(database_server))

Homepage: <http://www.drizzle.org/>

Project: <https://launchpad.net/drizzle>

Download: <http://www.drizzle.org/content/download>

Source: `bzr lp:drizzle`

Docs: <http://www.drizzle.org/content/documentation>

Docs: <http://docs.drizzle.org/>

Drizzle is an *Open Source* relational database “for the cloud” which was forked from *MySQL* 6.0.

- Drizzle stores all data as UTF-8.
- Drizzle has a minimal core and a plugin API.

MySQL

Wikipedia: <https://en.wikipedia.org/wiki/MySQL>
Homepage: <https://www.mysql.com/>
Download: <https://dev.mysql.com/downloads/mysql/>
Source: git <https://github.com/mysql/mysql-server>
Doc: <https://dev.mysql.com/doc/>

MySQL Community Edition is an *Open Source* relational database.

PostgreSQL

Wikipedia: <https://en.wikipedia.org/wiki/PostgreSQL>
Homepage: <http://www.postgresql.org/>
Download: <http://www.postgresql.org/download/>
Source: git <http://git.postgresql.org/git/postgresql.git>
Docs: <http://www.postgresql.org/docs/>
Docs: <http://www.postgresql.org/docs/9.4/static/index.html>
Docs: <http://www.postgresql.org/docs/9.4/static/sql.html>

PostgreSQL is an *Open Source* relational database.

- PostgreSQL has native support for storing and querying *JSON*.
- PostgreSQL has support for geographical queries (*PostGIS*).

SQLite

Wikipedia: <https://en.wikipedia.org/wiki/SQLite>
Homepage: <https://www.sqlite.org/>
Download: <https://www.sqlite.org/download.html>
Source:
Docs: <https://www.sqlite.org/docs.html>
Docs: <https://www.sqlite.org/different.html>
Docs: <https://www.sqlite.org/threadsafe.html>
Docs: <https://www.sqlite.org/uri.html>
FileExt: `.sqlite`

SQLite is a serverless *Open Source* relational database which stores all data in one file.

- SQLite is included in the Python standard library.

Virtuoso

Wikipedia: https://en.wikipedia.org/wiki/Virtuoso_Universal_Server

Homepage: <http://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/>

Source: git <https://github.com/openlink/virtuoso-opensource>

Docs: <http://docs.openlinksw.com/virtuoso/>

Docs: <http://docs.openlinksw.com/virtuoso/sqlreference.html>

Docs: <http://docs.openlinksw.com/virtuoso/rdfandsparql.html>

Docs: <http://docs.openlinksw.com/virtuoso/rdfsparql.html>

Docs: <http://docs.openlinksw.com/virtuoso/rdfsparqlrule.html>

Docs: <http://docs.openlinksw.com/virtuoso/rdfgraphsecurity.html>

Docs: <http://docs.openlinksw.com/virtuoso/virtuososponger.html>

Virtuoso *Open Source* edition is a multi-paradigm *relational database / XML* document database / *RDF triplestore*.

- Relational Tables Data Management (Columnar or Column-Store *SQL* RDBMS)
- Relational Property Graphs Data Management (*SPARQL RDF* based Quad Store)
- Content Management (*HTML*, TEXT, *Turtle*, *RDF/XML*, *JSON*, *JSON-LD*, *XML*)
- Web and other Document File Services (Web Document or File Server)
- *Five-Star Linked Open Data* Deployment (*RDF*-based *Linked Data* Server)
- Web Application Server (SOAP or *RESTful* interaction modes).
- Virtuoso supports ODBC, JDBC, and DB-API relational database access.
- Virtuoso powers *DBpedia*.

NoSQL Databases

Wikipedia: <https://en.wikipedia.org/wiki/NoSQL>

[https://en.wikipedia.org/wiki/Keyspace_\(distributed_data_store\)](https://en.wikipedia.org/wiki/Keyspace_(distributed_data_store))

[https://en.wikipedia.org/wiki/Column_\(data_store\)](https://en.wikipedia.org/wiki/Column_(data_store))

- https://en.wikipedia.org/wiki/Column_family
- https://en.wikipedia.org/wiki/Super_column
- [https://en.wikipedia.org/wiki/Apache:](https://en.wikipedia.org/wiki/Apache)

Graph Databases

Wikipedia: https://en.wikipedia.org/wiki/Graph_database

https://en.wikipedia.org/wiki/Graph_database#Graph_database_projects

- <https://en.wikipedia.org/wiki/AllegroGraph> [*RDF*]
- *Blazegraph* [*RDF*, *OWL*]

- *Neo4j*
- *Accumulo* + <https://en.wikipedia.org/wiki/Sqrrl>
- *Virtuoso* [*RDF*, *OWL*]
- <http://db-engines.com/en/ranking/graph+dbs>

Graph Queries

- https://en.wikipedia.org/wiki/Graph_database#APIs_and_Graph_Query.2FProgramming_Languages
- *SPARQL*
- *Gremlin*
- *Blueprints*
- *Spark GraphX*

Blazegraph

Homepage: <http://www.blazegraph.com/>

Download: <http://www.blazegraph.com/download>

Src: git [git://git.code.sf.net/p/bigdata/git](https://git.code.sf.net/p/bigdata/git)

Docs: <http://www.blazegraph.com/learn>

Docs: <http://www.blazegraph.com/inference>

Docs: <http://www.blazegraph.com/blueprints>

Docs: <http://www.blazegraph.com/sesame>

Docs: <http://www.blazegraph.com/develop>

Docs: <http://www.blazegraph.com/docs/api/>

Docs: https://wiki.blazegraph.com/wiki/index.php/Main_Page

Blazegraph is an *Open Source graph database* written in Java with support for *Gremlin*, *Blueprints*, *RDF*, *RDFS* and *OWL* inferencing, *SPARQL*.

- Blazegraph was formerly known as *Bigdata*.
- Blazegraph 1.5.2 supports *Solr* (e.g. TF-IDF) indexing.
- Blazegraph will power the *Wikidata* Query Service (RDF, SPARQL):
<https://lists.wikimedia.org/pipermail/wikidata-tech/2015-March/000740.html>
- MapGraph is a set of *GPU*-accelerations for graph processing.

Blueprints

Wikipedia:

Homepage:

Src: git <https://github.com/tinkerpop/blueprints>

Docs: <https://github.com/tinkerpop/blueprints/wiki>

Blueprints is an *Open Source graph database* API (and reference graph data model).

Blueprints is a collection of interfaces, implementations, ouplementations, and test suites for the property graph data model.

Blueprints is analogous to the JDBC, but for graph databases. As such, it provides a common set of interfaces to allow developers to plug-and-play their graph database backend.

Moreover, software written atop Blueprints works over all Blueprints-enabled graph databases.

Within the TinkerPop software stack, Blueprints serves as the foundational technology for:

- Pipes: A lazy, data flow framework
 - *Gremlin*: A graph traversal language
 - Frames: An object-to-graph mapper
 - Furnace: A graph algorithms package
 - Rexster: A graph server
- There are many blueprints API implementations (e.g. Rexster, *Neo4j*, *Blazegraph*, *Accumulo*)

Gremlin

Wikipedia: [https://en.wikipedia.org/wiki/Gremlin_\(programming_language\)](https://en.wikipedia.org/wiki/Gremlin_(programming_language))

Src: git <https://github.com/tinkerpop/gremlin>

Docs: <https://github.com/tinkerpop/gremlin/wiki>

Gremlin is an *Open Source* domain-specific language for traversing property graphs.

- Gremlin works with databases that implement the *Blueprints* graph database API.

Neo4j

Wikipedia: <https://en.wikipedia.org/wiki/Neo4j>

Homepage: <http://neo4j.com/>

Download: <http://neo4j.com/download/>

Src: git <https://github.com/neo4j/neo4j>

Docs: <http://neo4j.com/developer/get-started/>

Docs: <http://neo4j.com/docs/>

Docs: <http://neo4j.com/docs/2.2.3/>

Docs: <http://neo4j.com/developer/cypher/>

Docs: <http://neo4j.com/docs/stable/cypher-refcard/>

Docs: https://en.wikipedia.org/wiki/Cypher_Query_Language

Docs: <http://neo4j.com/open-source-project/>

Neo4j is an *Open Source* HA graph database written in Java.

- Neo4j implements the *Paxos* distributed algorithm for HA (*high availability*).
- Neo4j can integrate with *Spark* and *ElasticSearch*.
- Neo4j is widely deployed in production environments.

- There is a *Blueprints* API implementation for Neo4j:
<https://github.com/tinkerpop/blueprints/wiki/Neo4j-Implementation>

RDF Triplestores

Wikipedia: <https://en.wikipedia.org/wiki/Triplestore>

https://en.wikipedia.org/wiki/List_of_subject-predicate-object_databases

- *Blazegraph*
- [https://en.wikipedia.org/wiki/Jena_\(framework\)](https://en.wikipedia.org/wiki/Jena_(framework))
- [https://en.wikipedia.org/wiki/Sesame_\(framework\)](https://en.wikipedia.org/wiki/Sesame_(framework))
- *Virtuoso*
- <http://db-engines.com/en/ranking/rdf+store>

Graph Pattern Query Results

- *SPARQL*
- https://en.wikipedia.org/wiki/Redland_RDF_Application_Framework
 - <http://librdf.org/notes/contexts.html>
- [https://en.wikipedia.org/wiki/Jena_\(framework\)](https://en.wikipedia.org/wiki/Jena_(framework))
- SAIL (Storage and Inferencing Layer) API
- <https://en.wikipedia.org/wiki/CubicWeb>
- *RDFLib*

`rdfs:seeAlso`

- *Linked Data*
- *Semantic Web*
- *Semantic Web Standards*
- *Semantic Web Tools*

Distributed Databases

Wikipedia: https://en.wikipedia.org/wiki/Distributed_database

Wikipedia: https://en.wikipedia.org/wiki/Distributed_data_store

See: *Distributed Algorithms*

Accumulo

Wikipedia:

Homepage: <https://accumulo.apache.org/>

Download: <https://accumulo.apache.org/downloads/>

Source: git <https://github.com/apache/accumulo>

Docs: https://accumulo.apache.org/1.7/accumulo_user_manual.html

Docs: https://accumulo.apache.org/1.7/accumulo_user_manual.html#_accumulo_design

Twitter: <https://twitter.com/apacheaccumulo>

Apache Accumulo is an *Open Source* distributed database key/value store written in Java based on *BigTable* which adds realtime queries, streaming iterators, row-level ACLs and a number of additional features.

- Accumulo supports *MapReduce*-style computation.
- Accumulo supports streaming iterator computation.
- Accumulo supports *HDFS*.
- Accumulo implements a programmatic Java query API.

BigTable

Wikipedia: <https://en.wikipedia.org/wiki/BigTable>

Docs: <http://research.google.com/archive/bigtable.html>

Google BigTable is a open reference design for a distributed key/value column store and a proprietary production database system.

- BigTable functionality overlaps with that of the newer Pregel and Spanner distributed databases.
- Cloud BigTable is a *PaaS / SaaS* service with Java integration through an adaptation of *HBase* API.

Apache Beam

Homepage: <https://beam.apache.org/>

Src: <git://git.apache.org/beam.git>

Src: <https://github.com/apache/beam>

Docs: <https://beam.apache.org/documentation/>

Apache Beam is an open source batch and streaming parallel data processing framework with support for Apache Apex, Apache Flink, **‘Apache Spark’**, and Google Cloud Dataflow.

Cassandra

Wikipedia: https://en.wikipedia.org/wiki/Apache_Cassandra

Homepage: <https://cassandra.apache.org/>

Download: <https://cassandra.apache.org/download/>

Source: git <https://github.com/apache/cassandra>

Docs: <https://wiki.apache.org/cassandra/FrontPage>

Docs: <https://wiki.apache.org/cassandra/GettingStarted>

Docs: <http://docs.datastax.com/en/latest-dsc/>

Docs: http://docs.datastax.com/en/cassandra/2.1/cassandra/architecture/architectureIntro_c.html

Apache Cassandra is an *Open Source* distributed key/value super column store written in Java.

- Cassandra is similar to *Amazon AWS* Dynamo and *BigTable*.
- Cassandra supports *MapReduce*-style computation.
- Cassandra supports *HDFS*.
- Facebook is one primary supporter of *Cassandra* development.

Hadoop

Wikipedia: https://en.wikipedia.org/wiki/Apache_Hadoop

Homepage: <https://hadoop.apache.org/>

Download: <https://hadoop.apache.org/releases.html>

Source: git [git://git.apache.org/hadoop.git](https://git.apache.org/hadoop.git)

Source: git <https://github.com/apache/hadoop>

Docs: <http://hadoop.apache.org/docs/current/>

Docs: <http://hadoop.apache.org/docs/stable/>

Apache Hadoop is a collection of *Open Source* distributed computing components; particularly for *MapReduce*-style computation over Hadoop *HDFS* distributed filesystem.

HBase

Wikipedia: https://en.wikipedia.org/wiki/Apache_HBase

Homepage: <https://hbase.apache.org/>

Download: <https://www.apache.org/dyn/closer.cgi/hbase/>

Source: git [git://git.apache.org/hbase.git](https://git.apache.org/hbase.git)

Source: git <https://github.com/apache/hbase>

Docs: <https://hbase.apache.org/book.html>

Docs: <https://hbase.apache.org/book.html#conceptual.view>

Apache HBase is an *Open Source* distributed key/value super column store based on *BigTable* written in Java that does *MapReduce*-style computation over Hadoop *HDFS*.

- HBase has a Java API, a *RESTful API*, an *avro* API, and a *Thrift* API

Hive

Wikipedia: https://en.wikipedia.org/wiki/Apache_Hive

Homepage: <https://hive.apache.org/>

Download: <https://hive.apache.org/downloads.html>

Docs: <https://cwiki.apache.org/confluence/display/Hive/LanguageManual>

Docs: <https://hive.apache.org/javadocs/r1.2.1/api/index.html>

Docs: <https://cwiki.apache.org/confluence/display/Hive/Home>

Apache Hive is an *Open Source* data warehousing platform written in java.

- Hive can read data from *HDFS* and *S3*.
- *Hive* supports *Avro*, *Parquet*.
- HiveQL is a *SQL*-like language.

Parquet

Homepage: <https://parquet.apache.org/>

Download: <https://parquet.apache.org/downloads/>

Source: git [git://git.apache.org/incubator-parquet-mr.git](https://git.apache.org/incubator-parquet-mr.git)

Source: git <https://github.com/apache/parquet-mr>

Standard: <https://github.com/apache/parquet-format>

Docs: <https://parquet.apache.org/documentation/latest/>

Apache Parquet is an *Open Source* columnar storage format for *Distributed Databases*

Apache Parquet is a columnar storage format available to any project in the *Hadoop* ecosystem, regardless of the choice of data processing framework, data model or programming language.

- The *Parquet format* and *Parquet metadata* are encoded with *Thrift*:
- See also: *CSV*, *CSVW*

Presto

Homepage: <https://prestodb.io/>

Source: git <https://github.com/facebook/presto>

Docs: <https://prestodb.io/docs/current/>

Presto is an *Open Source* distributed query engine designed to query multiple datastores at once.

- Presto has connectors for *Cassandra*, *Hive*, *JMX*, *Kafka*, *MySQL*, and *PostgreSQL*.
- Presto does not yet support *SPARQL*.
- Presto does not yet support *SPARQL* federated query.

Spark

Wikipedia: https://en.wikipedia.org/wiki/Apache_Spark

Homepage: <https://spark.apache.org/>

Download: <https://spark.apache.org/downloads.html>

Source: git [git://git.apache.org/spark.git](https://git.apache.org/spark.git)

Source: git <https://github.com/apache/spark>

Docs: <https://spark.apache.org/documentation.html>

Docs: <https://spark.apache.org/docs/latest/>

Docs: <https://spark.apache.org/docs/latest/cluster-overview.html>

Docs: <https://spark.apache.org/docs/latest/quick-start.html>

Apache Spark is an *Open Source* distributed computation platform.

- Spark is in-memory; and 100x faster than *MapReduce*.
- Spark can work with data in/over/through *HDFS*, *Cassandra*, OpenStack *Swift*, *Amazon AWS S3*, and the local filesystem.
- Spark can be provisioned by YARN or Mesos.
- Spark has Java, Scala, Python, and *R language APIs*.
- Spark set a world sorting benchmark record in 2014: <https://spark.apache.org/news/spark-wins-daytona-gray-sort-100tb-benchmark.html>

GraphX

Wikipedia: https://en.wikipedia.org/wiki/Apache_Spark#GraphX

Homepage: <https://spark.apache.org/graphx/>

Docs: <https://spark.apache.org/docs/latest/graphx-programming-guide.html>

GraphX is an *Open Source* graph query framework built with *Spark*.

Distributed Algorithms

Wikipedia: https://en.wikipedia.org/wiki/Distributed_algorithm

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Distributed_algorithms

Distributed Databases and distributed *Information Systems* implement *Distributed Algorithms* designed to solve for *Confidentiality*, *Integrity*, and *Availability*.

As separate records / statements to be `yield`-ed or emitted:

- *Distributed Databases* implement *Distributed Algorithms*.
- **Distributed Information Systems** implement *Distributed Algorithms*.

See Also:

- https://en.wikipedia.org/wiki/Parallel_computing
- https://en.wikipedia.org/wiki/Supercomputer#Distributed_supercomputing
-

Distributed Computing Problems

Wikipedia: https://en.wikipedia.org/wiki/Distributed_computing

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Distributed_computing_problems

- [https://en.wikipedia.org/wiki/Consensus_\(computer_science\)](https://en.wikipedia.org/wiki/Consensus_(computer_science))
- https://en.wikipedia.org/wiki/Leader_election
- https://en.wikipedia.org/wiki/Distributed_concurrency_control
- https://en.wikipedia.org/wiki/Distributed_lock_manager

-

Non-blocking algorithm

Wikipedia: https://en.wikipedia.org/wiki/Non-blocking_algorithm

- [https://en.wikipedia.org/wiki/Lock_\(computer_science\)#Disadvantages](https://en.wikipedia.org/wiki/Lock_(computer_science)#Disadvantages)
- See: *File Locking*

DHT

Wikipedia: https://en.wikipedia.org/wiki/Distributed_hash_table

A DHT (Distributed Hash Table*) is a distributed key value store for storing values under a consistent file checksum hash which can be looked up with e.g. an exact string match.

- At an API level, a DHT is a key/value store.
- *DNS* is basically a DHT
- *Distributed Databases* all implement some form of a structure similar to a DHT (a replicated *keystore*); often for things like bloom filters (for fast search)
 - *Cassandra*, *Ceph*, *GlusterFS*
- browsers that maintain a local cache could implement a DHT (e.g. with *WebSocket* or *WebRTC*)
 - *webtorrent* (Javascript, Node.js, *WebRTC*)
- *BitTorrent magnet URIs (URNs)* contain a *key*, which is a *checksum* of a manifest, which can be retrieved from a *DHT*:

```
# <a href="magnet:?xt=urn:btih:IJBDPDSBT4QZLBIJ6NX7LITSZHZQ7F5I">.</a>
# key_uri = "IJBDPDSBT4QZLBIJ6NX7LITSZHZQ7F5I"
dht = DHT(); value = dht.get(key_uri)
```

- *Named Data Networking* is also essentially a cached *DHT*.

MapReduce

Wikipedia: <https://en.wikipedia.org/wiki/MapReduce>

MapReduce is a *distributed algorithm* for distributed computation.

- *BigTable*, *Hadoop*, *HDFS*, *Disco*, *DDFS* all support *MapReduce*-style computation.
- See also: `bashreduce`

Paxos

Wikipedia: [https://en.wikipedia.org/wiki/Paxos_\(computer_science\)](https://en.wikipedia.org/wiki/Paxos_(computer_science))

Docs: [https://en.wikipedia.org/wiki/Paxos_\(computer_science\)#Production_use_of_Paxos](https://en.wikipedia.org/wiki/Paxos_(computer_science)#Production_use_of_Paxos)

- [https://en.wikipedia.org/wiki/Paxos_\(computer_science\)#Production_use_of_Paxos](https://en.wikipedia.org/wiki/Paxos_(computer_science)#Production_use_of_Paxos)
 - *BigTable*, *Spanner*, *Megastore*
 - *Ceph*
 - *Neo4j*

Raft

Wikipedia: [https://en.wikipedia.org/wiki/Raft_\(computer_science\)](https://en.wikipedia.org/wiki/Raft_(computer_science))

Homepage: <https://raft.github.io/>

- [https://en.wikipedia.org/wiki/Raft_\(computer_science\)#Basics](https://en.wikipedia.org/wiki/Raft_(computer_science)#Basics)
 - Leader / Candidate / Follower
 - Heartbeat (Leader -> Followers [-> Candidates])
 - etcd (CoreOS, Kubernetes, configuration management)
 - skydns

Bulk Synchronous Parallel

Wikipedia: https://en.wikipedia.org/wiki/Bulk_synchronous_parallel

Bulk Synchronous Parallel (*BSP*) is a *distributed algorithm* for distributed computation.

- Google Pregel, Apache Giraph, and Apache *Spark* are built for a *Bulk Synchronous Parallel* model
- *MapReduce* can be expressed very concisely in terms of BSP.

Distributed Computing Protocols

- *CORBA*
- *Message Passing*
- *ESB*
- *MPI*
- *XML-RPC*
- *JSON-RPC*
- *Avro*
- *Protocol Buffers*
- *Thrift*
- *SOA*
 - *WS-**
 - *WSDL*

- *JSON-WSP*
- *ROA*
 - *REST*
- *WAMP*

- https://en.wikipedia.org/wiki/Comparison_of_data_serialization_formats
- Programming Languages' implementations:
 - https://en.wikipedia.org/wiki/Java_Remote_Method_Invocation
 - <https://twisted.readthedocs.io/en/latest/core/howto/pb-usage.html>
- *WS-**
- *REST (RESTful HTTP API)*
- *Protocol Buffers*
- *Thrift*
- *Avro*
- *msgpack*
- *WebSocket*
- *WebRTC*
- *JSON-WSP*
- *LDP (Turtle or JSON-LD RDF over HTTP)*
- *REST*
- *WAMP*
- https://en.wikipedia.org/wiki/List_of_web_service_protocols

CORBA

Wikipedia: https://en.wikipedia.org/wiki/Common_Object_Request_Broker_Architecture

CORBA (*Common Object Request Broker Architecture*) is a *distributed computing protocol* now defined by *OMG* with implementations in many languages.

- CORBA is a distributed object-oriented protocol for platform-neutral distributed computing.
- CORBA objects are marshalled and serialized according to an IDL (*Interface Definition Language*) with a limited set of datatypes (see also *XSD*, *Distributed Computing Protocols: Protocol Buffers, Thrift, Avro, msgpack, JSON-LD*)
- CORBA ORBs (*Object Request Brokers*) route requests for objects (see also *ESB*)
- CORBA objects are either in local address space (see also `file:/// /dev/mem`) or remote address space (see also dereferencable *HTTP, HTTPS URLs*)
- CORBA objects can be looked up by reference (by *URL*, or *NameService* (see also *DNS*))
- “CORBA Objects are passed by reference, while data (integers, doubles, structs, enums, etc.) are passed by value” – https://en.wikipedia.org/wiki/Common_Object_Request_Broker_Architecture#Features

Message Passing

Wikipedia: https://en.wikipedia.org/wiki/Message_passing
https://en.wikipedia.org/wiki/Messaging_pattern
https://en.wikipedia.org/wiki/Message_passing_in_computer_clusters
https://en.wikipedia.org/wiki/Active_message

- https://en.wikipedia.org/wiki/Message_passing#Synchronous_versus_asynchronous_message_passing
- https://en.wikipedia.org/wiki/Dataflow_programming
 - https://en.wikipedia.org/wiki/Flow-based_programming
 - <https://en.wikipedia.org/wiki/Spreadsheet>
 - https://en.wikipedia.org/wiki/Reactive_programming
- https://en.wikipedia.org/wiki/Actor_model_implementation
- https://en.wikipedia.org/wiki/Factor_graph#Message_passing_on_factor_graphs
- *Bulk Synchronous Parallel*

ESB

Wikipedia: https://en.wikipedia.org/wiki/Enterprise_service_bus

An ESB (*Enterprise Service Bus*) is a centralized distributed computing component which relays (or *brokers*) messages with or as a message queue (*MQ*).

- ESB is generally the name for a message queue / task worker pattern in the *SOA* (particularly Java).
- ESBs host service endpoints for message producers and consumers.
- ESBs can also maintain state, or logging.
- ESB services can often be described with e.g. *WSDL* and/or *JSON-WSP*.
- https://en.wikipedia.org/wiki/Category:Message-oriented_middleware

MPI

Wikipedia: https://en.wikipedia.org/wiki/Message_Passing_Interface

MPI (*Message Passing Interface*) is a distributed computing protocol for structured data interchange with implementations in many languages.

- Many supercomputing applications are built with MPI.
- MPI is faster than *JSON*.
- IPython `ipyparallel` supports MPI: <https://ipyparallel.readthedocs.io/en/latest/>

XML-RPC

Wikipedia: <https://en.wikipedia.org/wiki/XML-RPC>

XML Remote Procedure Call defines method names with parameters and values for making function calls with XML.

- Python `xmlrpc`lib: <https://docs.python.org/2/library/xmlrpclib.html>
<https://docs.python.org/3/library/xmlrpc.client.html>
<https://docs.python.org/3/library/xmlrpc.server.html>

See also:

- *JSON-RPC*
- `~:ref:C` structs: *Protocol Buffers*, *Thrift*, *Avro*
- *SOA* Web Services: *WS-**, *WSDL*
- *ROA* Web Services: *REST*

JSON-RPC

Wikipedia: <https://en.wikipedia.org/wiki/JSON-RPC>

Specification: <http://www.jsonrpc.org/specification>

Avro

Wikipedia: https://en.wikipedia.org/wiki/Apache_Avro

Homepage: <https://avro.apache.org/>

Standard: <https://avro.apache.org/docs/current/spec.html>

Standard: <https://avro.apache.org/docs/current/trevni/spec.html>

Download: <https://avro.apache.org/releases.html#Download>

Docs: <https://avro.apache.org/docs/current/>

Docs: <https://avro.apache.org/docs/current/gettingstartedjava.html>

Docs: <https://avro.apache.org/docs/current/api/java/>

Docs: <https://avro.apache.org/docs/current/gettingstartedpython.html>

Docs: <https://avro.apache.org/docs/current/api/c/>

Docs: <https://avro.apache.org/docs/current/api/cpp/html/>

Docs: <https://avro.apache.org/docs/current/api/csharp/>

Apache Avro is an RPC distributed computing protocol with implementations in many languages.

- Avro *schemas* are defined in *JSON*.
- Avro is similar to *Protocol Buffers* and *Thrift*, but does not require code generation.
- Avro stores *schemas* within the data.

seeAlso:

- *JSON-LD* maps to *RDF*

- 5stardata

Protocol Buffers

Homepage: <https://developers.google.com/protocol-buffers/>

Src: <https://github.com/google/protobuf>

Docs: <https://developers.google.com/protocol-buffers/docs/overview>

Protocol Buffers (*PB*) is a standard for structured data interchange.

- Protocol Buffers are faster than *JSON*

See also:

- *Thrift*
- *Avro*

Thrift

Wikipedia: https://en.wikipedia.org/wiki/Apache_Thrift

Homepage: <https://thrift.apache.org>

Src: <http://github.com/apache/thrift>

Docs: <https://thrift.apache.org/docs/>

Docs: <https://thrift.apache.org/docs/idl>

Thrift is a standard for structured data interchange in the style of *Protocol Buffers*.

- Thrift is faster than *JSON*.

See also:

- *Protocol Buffers*
- *Avro*

SOA

Wikipedia: https://en.wikipedia.org/wiki/Service-oriented_architecture

SOA (*Service Oriented Architecture*) is a collection of *Web Standards* (e.g *WS-**) and architectural patterns for distributed computing.

WS-*

Wikipedia: https://en.wikipedia.org/wiki/List_of_web_service_specifications

There are many web service specifications; many web service specifications often start with *WS-*.

- https://en.wikipedia.org/wiki/List_of_web_service_specifications

- Many/most WS-* standards specify *XML*.
- Some WS-* standards also specify *JSON*.

WSDL

Wikipedia: https://en.wikipedia.org/wiki/Web_Services_Description_Language

WSDL (*Web Services Description Language*) is a *web standard* for describing web services and the schema of their inputs and outputs.

JSON-WSP

Wikipedia: <https://en.wikipedia.org/wiki/JSON-WSP>

JSON-WSP (*JSON Web-Service Protocol*) is a *web standard* protocol for describing services and request and response objects.

- JSON-WSP is similar in function to *WSDL* and *CORBA IDL*.

See also: *Linked Data Platform (LDP)*

ROA

Wikipedia: https://en.wikipedia.org/wiki/Resource-oriented_architecture

REST

Wikipedia: https://en.wikipedia.org/wiki/Representational_state_transfer

Awesome: <https://github.com/marmelab/awesome-rest>

REST (*Representational State Transfer*) is a pattern for interacting with web resources using regular *HTTP* methods like GET, POST, PUT, and DELETE.

- A REST *API* is known as a RESTful API.
- A REST implementation maps Create, Read, Update, Delete (CRUD) methods for URI-named collections of **resources** onto HTTP verbs like GET, POST, PATCH.
- Sometimes, a REST implementation accepts a *URL* parameter like `?method=PUT` e.g. for Javascript implementations on browsers which only support e.g. GET and POST.
- There are many software libraries for implementing REST API Servers:
 - Java, JS: Restlet:

Wikipedia: <https://en.wikipedia.org/wiki/Restlet>

Src: <https://github.com/restlet>

- Ruby: Grape:

Src: <https://github.com/ruby-grape/grape>

- Python: Django REST Framework:

Src: <https://github.com/tomchristie/django-rest-framework>

- There are many software libraries for implementing REST API Clients:

- Python REST API client libraries:

- * requests:

Src:

Docs: <http://docs.python-requests.org/en/master/>

- httpie is a CLI utility written on top of requests:

Src: <https://github.com/jkbrzt/httpie>

- * WebTest:

Src: <https://github.com/Pylons/webtest>

Docs: <https://webtest.readthedocs.io/en/latest/>

- <https://pypi.python.org/pypi/webtest-plus/> (requests-auth)
- <https://github.com/django-webtest/django-webtest>

- * Docs: <https://westerner.github.io/wiki/awesome-python-testing#web-applications>

WAMP

Wikipedia: https://en.wikipedia.org/wiki/Web_Application_Messaging_Protocol

Homepage: <http://wamp-proto.org>

Specification: <https://tools.ietf.org/html/draft-oberstet-hybi-tavendo-wamp>

Src: <https://github.com/wamp-proto/wamp-proto>

Docs: <http://wamp-proto.org/why/>

Docs: <http://wamp-proto.org/faq/>

Docs: <http://wamp-proto.org/implementations/>

WAMP (*Web Application Messaging Protocol*) defines Publish/Subscribe (PubSub) and Remote Procedure Call (RPC) over *WebSocket*, *JSON*, and *URIs*

Using WAMP, you can have a browser-based UI, the embedded device and your backend talk to each other in real-time:

- WAMP Router = Broker (PubSub topic broker) + Dealer (RPC)
- WAMP can run on other transports (e.g. msgpack) than the preferred *WebSocket* w/ *JSON*.
 - *JSON-LD*
- Implementations:
 - <http://wamp-proto.org/implementations/>
 - <http://autobahn.ws/> (Python, JS, Cpp, Android, Test Suite)
 - * <http://autobahn.ws/#code>
- <https://tools.ietf.org/html/draft-oberstet-hybi-tavendo-wamp#section-6.5>

WAMP Message Codes and Direction

Data Grid

Wikipedia: https://en.wikipedia.org/wiki/Data_grid

Search Engine Indexing

Wikipedia: https://en.wikipedia.org/wiki/Search_engine_indexing

- https://en.wikipedia.org/wiki/Web_search_engine
- *Information Retrieval*
- *Semantic Web graph of Linked Data, : RDFa, JSON-LD, Schema.org.*

ElasticSearch

Wikipedia: <https://en.wikipedia.org/wiki/Elasticsearch>

Homepage: <https://www.elastic.co/products/elasticsearch>

Download: <https://www.elastic.co/downloads/elasticsearch>

Source: git <https://github.com/elastic/elasticsearch>

Docs: <https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>

Docs: <https://www.elastic.co/guide/en/elasticsearch/guide/current/index.html>

DockerHub: <https://registry.hub.docker.com/u/library/elasticsearch/>

ElasticSearch is an *Open Source* realtime search server written in Java built on Apache *Lucene* with a *RESTful API* for indexing *JSON* documents.

- ElasticSearch supports geographical (bounded) queries.
- ElasticSearch can build better indexes for faster search response times when *ElasticSearch Mappings* are specified.
- ElasticSearch mappings can be (manually) transformed to *JSON-LD* @context mappings: <https://github.com/westurner/elasticsearchjsonld>

Haystack

Homepage: <http://haystacksearch.org/>

Source: git <https://github.com/django-haystack/django-haystack>

PyPI: <https://pypi.python.org/pypi/django-haystack>

Docs: <https://django-haystack.readthedocs.io/en/latest/>

Haystack is an *Open Source* Python Django API for a number of search services (e.g. *Solr*, *ElasticSearch*, *Whoosh*, *Xapian*).

Lucene

Wikipedia: <https://en.wikipedia.org/wiki/Lucene>

Homepage: <https://lucene.apache.org/>

Download: <https://lucene.apache.org/core/downloads.html>

Source: svn <http://svn.apache.org/repos/asf/lucene/dev/trunk>

Docs: <https://lucene.apache.org/core/>

Docs: https://lucene.apache.org/core/5_2_0/

Apache Lucene is an *Open Source* search indexing service written in java.

- *ElasticSearch*, *Nutch*, and *Solr* are implemented on top of Lucene.

Nutch

Wikipedia: <https://en.wikipedia.org/wiki/Nutch>

Homepage: <https://nutch.apache.org/>

Download: <https://nutch.apache.org/downloads.html>

Source: git <git://git.apache.org/nutch.git>

Source: git <https://github.com/apache/nutch>

Docs: <https://nutch.apache.org/apidocs/apidocs-2.3/index.html>

Docs: <https://wiki.apache.org/nutch/>

Docs: <https://wiki.apache.org/nutch/#Tutorials>

Apache Nutch is an *Open Source* distributed web crawler and search engine written in Java and implemented on top of *Lucene*.

- Nutch has a pluggable storage and indexing API with support for e.g. *Solr*, *ElasticSearch*.

Solr

Wikipedia:

Homepage: <https://lucene.apache.org/solr/>

Download: <https://lucene.apache.org/solr/mirrors-solr-latest-redirect.html>

Docs: <https://lucene.apache.org/solr/resources.html>

Docs: <https://www.apache.org/dyn/closer.cgi/lucene/solr/ref-guide/>

Docs: <https://wiki.apache.org/solr/>

Apache Solr is an *Open Source* web search platform written in Java and implemented on top of *Lucene*.

Whoosh

Homepage:

PyPI: <https://pypi.python.org/pypi/Whoosh>

Docs: <https://pythonhosted.org/Whoosh/>

Whoosh is an *Open Source* search indexing service written in Python.

Xapian

Wikipedia: <https://en.wikipedia.org/wiki/Xapian>

Homepage: <http://xapian.org/>

Docs: <http://xapian.org/docs/>

Docs: <http://xapian.org/docs/apidoc/html/inherits.html>

Xapian is an *Open Source* search library written in C++ with bindings for many languages.

Information Retrieval

Wikipedia: https://en.wikipedia.org/wiki/Information_retrieval

Docs: <http://nlp.stanford.edu/IR-book/information-retrieval.html>

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press. 2008.

<http://nlp.stanford.edu/IR-book/>

Time Standards

International Atomic Time (IAT)

Wikipedia: https://en.wikipedia.org/wiki/International_Atomic_Time

International Atomic Time (*IAT*) is an international standard for extremely precise time keeping; which is the basis for *UTC* Earth time and for *Terrestrial Time* (Earth and Space).

Long Now Dates

Homepage: https://en.wikipedia.org/wiki/Long_Now_Foundation

Docs: https://en.wikipedia.org/wiki/Year_10,000_problem

```
2015      # ISO8601 date
02015     # 5-digit Y10K date
```

Decimal Time

Wikipedia: https://en.wikipedia.org/wiki/Decimal_time

- https://en.wikipedia.org/wiki/Decimal_time#Conversions
- https://en.wikipedia.org/wiki/Decimal_time#Fractional_days
- https://en.wikipedia.org/wiki/Leap_year (~365.25 days/yr)
- https://en.wikipedia.org/wiki/Leap_second (rotation time ~= atomic time)

Unix Time

Wikipedia: https://en.wikipedia.org/wiki/Unix_time

Defined as the number of seconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970, not counting leap seconds

Unix time is the delta in seconds since 1970-01-01T00:00:00Z, not counting leap seconds:

```
0          # Unix time
1970-01-01T00:00:00Z # ISO8601 timestamp

1435255816 # Unix time
2015-06-25T18:10:16Z # ISO8601 timestamp
```

Note: Unix time does not count leap seconds.

https://en.wikipedia.org/wiki/Unix_time#Leap_seconds

See also: *Swatch Internet Time (Beat Time)*

Year Zero

Wikipedia: [https://en.wikipedia.org/wiki/0_\(year\)](https://en.wikipedia.org/wiki/0_(year))

- The Gregorian Calendar (e.g. *Common Era*, *Julian Calendar*) does not include a *year zero*; (1 BCE is followed by 1 CE).
- *Astronomical year numbering* includes a *year zero*.
- *Before Present* dates do not specify a *year zero*. (because they are relative to the current (or *published*) date).

Astronomical year numbering

Wikipedia: https://en.wikipedia.org/wiki/Astronomical_year_numbering

- Astronomical year numbering includes a year zero:

Tools with support for *Astronomical year numbering*:

- AstroPy is a Python library that supports astronomical year numbering:
<https://astropy.readthedocs.io/en/latest/time/>

Before Present (BP)

Wikipedia: https://en.wikipedia.org/wiki/Before_Present

Before Present (*BP*) dates are relative to the current date (or *date of publication*); e.g. “2.6 million years ago”.

Common Era (CE)

Wikipedia: https://en.wikipedia.org/wiki/Common_Era

Docs: https://en.wikipedia.org/wiki/Pax_Romana

Docs: *Year Zero*

- BCE (*Before Common Era*) == BC
 - <https://en.wiktionary.org/wiki/BCE>
 - <https://en.wiktionary.org/wiki/BC>
- CE (*Common Era*) == **AD** (*Anno Domini*)
 - <https://en.wiktionary.org/wiki/CE>
 - <https://en.wiktionary.org/wiki/AD>

Common Era and *Year Zero*:

5000 BCE	==	-5000 CE
1 BCE	==	-1 CE
0 BCE	==	0 CE
0 CE	==	0 BCE
1 CE	==	1 CE
2015 CE	==	2015 CE

Note: Are these off by one?

- *Astronomical year numbering* – you must convert from julian/gregorian dates to *Astronomical year numbering*.
- *Year Zero* – they are off by one (“there is no year zero”).

Common Era and Python datetime calculations:

```
# Paleolithic Era (2.6m years ago -> 12000 years ago)
# "2.6m years ago" = (2.6m - (2015)) BCE = 2597985 BCE = -2597985 CE

2597985 BCE == -2597985 CE

### Python datetime w/ scientific notation string formatter
>>> import datetime
>>> year = datetime.datetime.now().year
>>> '{:.6e}'.format(2.6e6 - year)
'2.597985e+06'

### Python datetime supports (dates >= 1 BCE).
>>> datetime.date(1, 1, 1)
datetime.date(1, 1, 1)
>>> datetime.datetime(1, 1, 1)
>>> datetime.datetime(1, 1, 1, 0, 0)

### Python pypi:arrow supports (dates >= 1 BCE).
>>> !pip install arrow
>>> arrow.get(1, 1, 1)
<Arrow [0001-01-01T00:00:00+00:00]>

### astropy.time.Time supports (1 BCE <= dates >= 1 CE) and/or *Year Zero*
### https://astropy.readthedocs.io/en/latest/time/
>>> !conda install astropy
>>> import astropy.time
>>> # TimeJulianEpoch (Julian date (jd) ~= Common Era (CE))
>>> astropy.time.Time(-2.6e6, format='jd', scale='utc')
<Time object: scale='utc' format='jd' value=-2600000.0>
```

Time Zones

Wikipedia: https://en.wikipedia.org/wiki/Time_zone

https://en.wikipedia.org/wiki/Daylight_saving_time

https://en.wikipedia.org/wiki/List_of_UTC_time_offsets

https://en.wikipedia.org/wiki/List_of_tz_database_time_zones

- *ISO8601*

UTC

Wikipedia: https://en.wikipedia.org/wiki/Coordinated_Universal_Time

UTC (*Coordinated Universal Time*) is the primary terrestrial Earth-based clock time.

- Earth *Time Zones* are specified as offsets from *UTC*.
- UTC time is set determined by *International Atomic Time (IAT)*; with occasional leap seconds to account for the difference between Earth's rotational time and the actual passage of time according to the decay rate of cesium atoms (an *SI Unit* calibrated with an *atomic clock*; see *QUDT*).

- Many/most computer systems work with UTC, but are not exactly synchronized with *International Atomic Time (IAT)* (see also: *RTC*, *NTP* and *time drift*).

US Time Zones

Wikipedia: https://en.wikipedia.org/wiki/Time_in_the_United_States

https://en.wikipedia.org/wiki/Time_in_the_United_States#Standard_time_and_daylight_saving_time

https://en.wikipedia.org/wiki/History_of_time_in_the_United_States

Time Zone names, URIs, and *ISO8601* UTC offsets:

Table 12.1: Table of US Time Zones

Time Zone names, URNs, URIs	UTC Offset	UTC DST Offset
https://en.wikipedia.org/wiki/Coordinated_Universal_Time #tz: Coordinated Universal Time, UTC, Zulu	-0000 Z	+0000 Z
https://en.wikipedia.org/wiki/Atlantic_Time_Zone https://en.wikipedia.org/wiki/America/Halifax #tz: Atlantic, Antarctica (Palmer), AST, ADT America/Halifax	-0400 AST	-0300 ADT
https://en.wikipedia.org/wiki/America/St_Thomas #tz: America/St_Thomas, America/Virgin	-0400	-0400
https://en.wikipedia.org/wiki/Eastern_Time_Zone https://en.wikipedia.org/wiki/EST5EDT #tz: Eastern, EST, EDT America/New_York	-0500 EST	-0400 EDT
https://en.wikipedia.org/wiki/Central_Time_Zone https://en.wikipedia.org/wiki/CST6CDT #tz: Central, CST, CDT America/Chicago	-0600 CST	-0500 CDT
https://en.wikipedia.org/wiki/Mountain_Time_Zone https://en.wikipedia.org/wiki/MST7MDT #tz: Mountain, MST, MDT America/Denver	-0700 MST	-0600 MDT
https://en.wikipedia.org/wiki/Pacific_Time_Zone https://en.wikipedia.org/wiki/PST8PDT #tz: Pacific, PST, PDT America/Los_Angeles	-0800 PST	-0700 PDT
https://en.wikipedia.org/wiki/Alaska_Time_Zone AKST9AKDT #tz: Alaska, AKST, AKDT America/Juneau	-0900 AKST	-0800 AKDT
https://en.wikipedia.org/wiki/Hawaii-Aleutian_Time_Zone HAST10HADT #tz: Hawaii Aleutian, HAST, HADT Pacific/Honolulu	-1000 HAST	-0900 HADT
https://en.wikipedia.org/wiki/Samoa_Time_Zone #tz: Samoa Time Zone, SST Pacific/Samoa	-1100 SST	-1100 SST
https://en.wikipedia.org/wiki/Chamorro_Time_Zone #tz: Chamorro Time Zone, CHST, CHADT Pacific/Chuuk	+1000	+1000

US Daylight Saving Time

Wikipedia: https://en.wikipedia.org/wiki/Daylight_saving_time_in_the_United_States

Currently, daylight saving time **starts on the second Sunday in March** and **ends on the first Sunday in November**, with the time changes taking place **at 2:00 a.m. local time**.

With a mnemonic word play referring to seasons, clocks “*spring forward and fall back*” — that is, in spring (technically late winter) the clocks are moved forward from 2:00 a.m. to 3:00 a.m., and in fall they are moved back from 2:00 am to 1:00 am.

Daylight Savings Time Starts and Ends on the following dates (from https://en.wikipedia.org/wiki/Time_in_the_United_States#Daylight_saving_time):

Year	DST start date	DST end date
2015	2015-03-08 02:00	2015-11-01 02:00
2016	2016-03-13 02:00	2016-11-06 02:00
2017	2017-03-12 02:00	2017-11-05 02:00
2018	2018-03-11 02:00	2018-11-04 02:00
2019	2019-03-10 02:00	2019-11-03 02:00
2020	2020-03-08 02:00	2020-11-01 02:00

ISO8601

Wikipedia: https://en.wikipedia.org/wiki/ISO_8601

Standard: <http://www.iso.org/iso/iso8601>

ISO8601 is an *ISO* standard for specifying Gregorian dates, times, datetime intervals, durations, and recurring date-times.

- The `date` command can print *ISO8601* -compatible datestrings:

```
$ date +%FT%T%z'
2016-01-01T22:11:59-0600

$ date +%F %T%z'
2016-01-01 22:11:59-0600
```

- Roughly, an ISO8601 datetime is specified as: year, dash month, dash day, (T or “ “ [space-character]), hour, colon, minute, colon, second, (Z [for UTC] or a *time zone* offset (e.g. +/- -0000, +0000)); where the dashes and colons are optional.
- ISO8601 specifies a standard for absolute time durations: start date, forward-slash, end date.
- ISO8601 specifies a standard for relative time durations: number of years Y, months M, days D, hours H, minutes M, and seconds S.
- A Z timezone specifies **UTC** (*Universal Coordinated Time*) (or “Zulu”) time.
- Many/most *W3C* standards (such as *XSD*) specify *ISO8601* time formats: <http://www.w3.org/TR/NOTE-datetime>

A few examples of ISO8601:

```
2014
2014-10
2014-10-23
20141023
2014-10-23T20:59:30+Z      # UTC / Zulu
2014-10-23T20:59:30Z      # UTC / Zulu
2014-10-23T20:59:30-06:00 # CST
2014-10-23T20:59:30-06    # CST
2014-10-23T20:59:30-05:00 # CDT
2014-10-23T20:59:30-05    # CDT
20
20:59
2059
20:59:30
205930
2014-10-23T20:59:30Z/2014-10-23T21:00:00Z
2014-10-23T20:59:30-05:00/2014-10-23T21:00:00-06
PT1H
PT1M
P1M
P1Y1M1W1DT1H1M1S
```

Note: AFAIU, ISO8601 does not specify standards for milliseconds, microseconds, nanoseconds, picoseconds, femtoseconds, or attoseconds.

NTP

Wikipedia: https://en.wikipedia.org/wiki/Network_Time_Protocol

Homepage: <http://www.pool.ntp.org/en/>

NTP (*Network Time Protocol*) is a standard for synchronizing clock times.

- Most Operating Systems and mobile devices support *NTP*.
- NTP clients calculate *time drift* (or *time skew*) and network latency and then gradually adjust the local system time to the most recently retrieved server time.
- Many OS distributions run their own *NTP* servers (in order to reduce load on the core NTP pool servers).

Linked Data

Wikipedia: https://en.wikipedia.org/wiki/Linked_data

- <http://www.w3.org/DesignIssues/LinkedData.html>
- Linked Data Standards:
 - W3C: https://www.w3.org/TR/#tr_Linked_Data

5 Linked Data

<http://www.w3.org/TR/ld-glossary/#x5-star-linked-open-data>

Publish data on the Web in any format (e.g., PDF, JPEG) accompanied by an explicit Open License (expression of rights).

Publish structured data on the Web in a machine-readable format (e.g. *XML*).

Publish structured data on the Web in a documented, non-proprietary data format (e.g. *CSV*, *KML*).

Publish structured data on the Web as RDF (e.g. *Turtle*, *RDFa*, *JSON-LD*, *SPARQL*.)

In your *RDF*, have the identifiers be links (URLs) to useful data sources.

—<http://5stardata.info/>

See: *Semantic Web*

Semantic Web

Wikipedia: https://en.wikipedia.org/wiki/Semantic_Web

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Semantic_Web

https://en.wikipedia.org/wiki/Template:Semantic_Web

[https://en.wikipedia.org/wiki/Semantics_\(computer_science\)](https://en.wikipedia.org/wiki/Semantics_(computer_science))

W3C Semantic Web Wiki:

- <http://www.w3.org/2001/sw/wiki>
- <http://www.w3.org/2001/sw/wiki/Books>
- <http://www.w3.org/2001/sw/wiki/Tools>

Semantic Web Standards

[https://en.wikipedia.org/wiki/Statement_\(computer_science\)](https://en.wikipedia.org/wiki/Statement_(computer_science))

[https://en.wikipedia.org/wiki/Resource_\(computing\)](https://en.wikipedia.org/wiki/Resource_(computing))

https://en.wikipedia.org/wiki/Entity-attribute-value_model

<https://en.wikipedia.org/wiki/Tuple>

- [https://en.wikipedia.org/wiki/Triple_\(mathematics\)](https://en.wikipedia.org/wiki/Triple_(mathematics))
- <https://en.wikipedia.org/wiki/3-tuple>
- [https://en.wikipedia.org/wiki/Quad_\(mathematics\)](https://en.wikipedia.org/wiki/Quad_(mathematics))
- <https://en.wikipedia.org/wiki/4-tuple>

[https://en.wikipedia.org/wiki/Reification_\(computer_science\)#Reification_on_Semantic_Web](https://en.wikipedia.org/wiki/Reification_(computer_science)#Reification_on_Semantic_Web)

https://en.wikipedia.org/w/index.php?title=Eigenclass_model&oldid=592778140#In_RDF_Schema

Representations / Serializations

- *RDF: N-Triples, RDF/XML, TriX, N3, Turtle, TriG, RDFa, JSON-LD*

Vocabularies

- *RDFS: DCMI, SKOS, Schema.org*

Query APIS

- *SPARQL, LDP*

Ontologies

- *OWL: PROV, OA, QUDT*

Reasoners

- See:
 - *Description Logic*
 - *OWL 2 Profiles*
 - *Entailment*

Web Standards

Wikipedia: https://en.wikipedia.org/wiki/Web_standards

Web Names

URL

- *URL*

URI

- *URI*
- *Magnet URI*

URN

- *URN*

IEC

Wikipedia: https://en.wikipedia.org/wiki/International_Electrotechnical_Commission

Homepage: <http://www.iec.ch/>

IEC (*International Electrotechnical Commission*) is a standards body.

- List of IEC standards: https://en.wikipedia.org/wiki/List_of_IEC_standards

IETF

Wikipedia: https://en.wikipedia.org/wiki/Internet_Engineering_Task_Force

Homepage: <https://www.ietf.org/>

IETF (*Internet Engineering Task Force*) is a standards body.

- List of IETF standards: <https://tools.ietf.org/html/>

ISO

Wikipedia: https://en.wikipedia.org/wiki/International_Organization_for_Standardization

Homepage: <http://www.iso.org/>

ISO (*International Organization for Standardization*) is a standards body.

- List of ISO standards: <http://www.iso.org/iso/home/standards.htm>

OMG

Wikipedia: https://en.wikipedia.org/wiki/Object_Management_Group

Homepage: <http://www.omg.org/>

OMG (*Object Management Group*) is a standards body.

- UML is an OMG standard.
- *CORBA* is now an OMG standard.
- List of OMG standards: <http://www.omg.org/spec/>
https://en.wikipedia.org/wiki/Object_Management_Group#OMG_Standards

W3C

Wikipedia: https://en.wikipedia.org/wiki/World_Wide_Web_Consortium

Homepage: <http://www.w3.org/>

W3C (*World Wide Web Consortium*) is a standards body.

- List of W3C standards: <http://www.w3.org/TR/>
 - https://www.w3.org/TR/#tr_Linked_Data

HTTP

Wikipedia: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol

Standard: <https://tools.ietf.org/html/rfc2616>

Standard: <http://tools.ietf.org/html/rfc7230#page-5>

Docs: https://www.mnot.net/blog/2014/06/07/rfc2616_is_dead

URI Scheme: <http://>

URI Scheme: <https://>

HTTP (*HyperText Transfer Protocol*) is an *Open Source* text-based request-response TCP/IP protocol for text and binary data interchange.

- *HTTPS* (*Secure HTTP*) wraps HTTP in SSL/TLS to secure HTTP.

HTTP in RDF

Standard: <http://www.w3.org/TR/HTTP-in-RDF10/>

Namespace: <http://www.w3.org/2011/http#>

Namespace: `<http://www.w3.org/2011/http-headers> .'_`

Namespace: `<http://www.w3.org/2011/http-methods> .'_`

Namespace: `<http://www.w3.org/2011/http-statusCodes> .'_`

xmlns: @prefix http: <<http://www.w3.org/2011/http#>> .

xmlns: @prefix http-headers: <<http://www.w3.org/2011/http-headers>> .

xmlns: @prefix http-methods: <<http://www.w3.org/2011/http-methods>> .

xmlns: @prefix http-statusCodes: <<http://www.w3.org/2011/http-statusCodes>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/http>

HTTP-in-RDF is a standard for representing *HTTP* as *RDF*.

HTTPS

Standard: <https://tools.ietf.org/html/rfc2818> (2000)

Wikipedia: <https://en.wikipedia.org/wiki/HTTPS>

Wikipedia: https://en.wikipedia.org/wiki/Transport_Layer_Security

Wikipedia: https://en.wikipedia.org/wiki/Secure_Sockets_Layer

HTTPS (*HTTP over SSL*) is *HTTP* wrapped in TLS/SSL.

- TLS (*Transport Layer Security*)
- SSL (*Secure Sockets Layer*)

HTTP STS

Wikipedia: https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security

HTTP STS (*HTTP Strict Transport Security*) is a standardized extension for notifying browsers that all requests should be made over *HTTPS* indefinitely or for a specified time period.

See also: [https](https://) everywhere

CSS

Wikipedia: https://en.wikipedia.org/wiki/Cascading_Style_Sheets

Docs: *CSS*

CSS (*Cascading Style Sheets*) define the presentational aspects of *HTML* and a number of mobile and desktop web frameworks.

- CSS is designed to ensure separation of data and presentation. With javascript, the separation is then data, code, and presentation.

RTMP

Wikipedia: https://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol

RTMP is a TCP/IP protocol for streaming audio, video, and data originally for Flash which is now *Open Source*.

- https://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol#Client_software
 - Adobe Flash Player
 - *VLC*
- https://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol#Server_software
 - Adobe Flash Live Media Server
 - *Amazon AWS* S3 HTTP Object Storage, CloudFront *CDN*
 - Helix Universal Media Server
 - Red5 (*Open Source*)
 - *FFmpeg* (*Open Source*)
 - *nginx-rtmp-module* (*Open Source*)
 - FreeSwitch (*OpenSource*, VoIP, SIP, *Video Chat*)
- *WebRTC* solves for all of the RTMP use cases, and is becoming as or more widely deployed than Flash Player (especially with mobile devices).

WebSocket

Wikipedia: <https://en.wikipedia.org/wiki/WebSocket>

URI Scheme: *ws://*

WebSocket is a full-duplex (two-way) TCP/IP protocol for audio, video, and data which can interoperate with *HTTP* Web Servers.

- WebSockets are often more efficient than other methods for realtime HTTP like HTTP Streaming and long polling.
- WebSockets work with many/most HTTP proxies

https://en.wikipedia.org/wiki/Comparison_of_WebSocket_implementations

- Python: `pypi:gevent-websocket`, `pypi:websockets (asyncio)`, `pypi:autobahn (pypi:twisted, asyncio)`

See also: https://en.wikipedia.org/wiki/Push_technology, *WebRTC*

WebRTC

Wikipedia: <https://en.wikipedia.org/wiki/WebRTC>

Homepage: <http://www.webrtc.org/>

Standard: <http://tools.ietf.org/wg/rtcweb/>

Docs: <https://webrtc.github.io/samples/>

WebRTC is a *web standard* for decentralized or centralized streaming of audio, video, and data in browser, without having to download any plugins.

Note: WebRTC is supported by a growing number of browsers: <http://iswebrtcreadyyet.com/>

Notably, Internet Explorer and Safari still require a plugin to handle *WebRTC*.

HTTP/2

Wikipedia: <https://en.wikipedia.org/wiki/HTTP/2>

Homepage: <https://http2.github.io/>

Standard: <https://http2.github.io/http2-spec/>

Standard: <https://http2.github.io/http2-spec/compression.html>

Standard: <https://tools.ietf.org/html/rfc7540>

Docs: <https://github.com/http2/http2-spec/wiki/Implementations>

HTTP/2 (*HTTP2*) is the newest standard for *HTTP*.

- HTTP/2 is largely derived from the SPDY protocol.

HTML

Wikipedia: <https://en.wikipedia.org/wiki/HTML>

HTML (*HyperText Markup Language*) is a *Open Source* standard for representing documents with tags, attributes, and **hyperlinks**.

Recent HTML standards include *HTML4*, *XHTML*, and *HTML5*.

HTML4

Standard: <http://www.w3.org/TR/html4/>

HTML4 is the fourth generation *HTML* standard.

XHTML

Wikipedia: <https://en.wikipedia.org/wiki/XHTML>

Standard: <http://www.w3.org/TR/xhtml2/>

XHTML is an *XML*-conforming *HTML* standard which is being superseded by *HTML5*.

Compared to *HTML4*, XHTML requires closing tags, supports additional namespace declarations, and expects things to be wrapped in CDATA blocks, among a few other notable differences.

XHTML has not gained the widespread adoption of *HTML4*, and is being largely superseded by *HTML5*.

HTML5

Standard: <http://www.w3.org/TR/html5/>

HTML5 is the fifth generation *HTML* standard with many new (and removed) features.

Like its predecessors, HTML5 is not case sensitive, but it is recommended to use lowercased tags and attributes.

Differences Between HTML4 and HTML5

<https://html-differences.whatwg.org/>

- HTML5 does not require closing tags (many browsers had already implemented routines for auto-closing broken markup).
- Frames have been removed
- Presentational attributes have been removed (in favor of CSS)

HTML 5.1

HTML 5.1 is in the works:

- <http://www.w3.org/html/wg/drafts/html/master/>

XML

Wikipedia: <https://en.wikipedia.org/wiki/XML>

Standard: <http://www.w3.org/TR/xml/>

XML (*Extensible Markup Language*) is a standard for representing data with tags and attributes.

Like PDF, XML is derived from SGML.

XSD

Wikipedia: [https://en.wikipedia.org/wiki/XML_Schema_\(W3C\)](https://en.wikipedia.org/wiki/XML_Schema_(W3C))

Standard: <http://www.w3.org/TR/xmlschema11-2/>

Namespace: <http://www.w3.org/2001/XMLSchema#>

xmlns: @prefix xsd: <<http://www.w3.org/2001/XMLSchema#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/xsd>

XSD (*XML Schema Datatypes*) are standard datatypes for things like strings, integers, floats, and dates for *XML* and also *RDF*.

- <https://www.w3.org/TR/xmlschema11-2/#built-in-datatypes>

JSON

Wikipedia: <https://en.wikipedia.org/wiki/JSON>

Standard: <https://tools.ietf.org/html/rfc7159>

Homepage: <http://json.org/>

JSON (*JavaScript Object Notation*) is a standard for representing data in a JavaScript compatible way; with a restricted set of data types.

Conforming JSON does not contain JavaScript code, only data. It is not safe to `eval` JSON, because it could contain code.

There are many parsers for JSON.

JSON-LD adds *RDF* Linked Data support to JSON with `@context`.

CSV

Wikipedia: https://en.wikipedia.org/wiki/Comma-separated_values

Standard: <https://tools.ietf.org/html/rfc4180>

Extension: `.csv`

MIME Type: `text/csv`

CSV (*Comma Separated Values*) as a flat file representation for columnar data with rows and columns.

Most spreadsheet tools can export (raw and computed) data from a sheet into a CSV file, for use with many other tools.

CSVW

Homepage: <https://w3c.github.io/csvw/>

Standard: <http://www.w3.org/TR/tabular-data-model/>

Standard: <http://www.w3.org/TR/tabular-metadata/>

Standard: <http://www.w3.org/TR/csv2json/>

Standard: <http://www.w3.org/TR/csv2rdf/>

Namespace: <http://www.w3.org/ns/csvw#>

`xmlns: @prefix csvw: <http://www.w3.org/ns/csvw#> .`

`@context: http://www.w3.org/ns/csvw.jsonld`

CSVW (*CSV on the Web*) is a set of relatively new standards for representing *CSV* rows and columns as *RDF* (and *JSON* / *JSON-LD*) along with *metadata*.

- URIs for datatypes (*XSD*, ...)
- URIs for columns (*RDF*)
- Document Metadata

- *CSV* -> *JSON* (-> *JSON-LD* -> *RDF*)
- *CSV* -> *RDF*

RDF

Wikipedia: https://en.wikipedia.org/wiki/Resource_Description_Framework

xmlns: @prefix rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/rdf>

RDF (*Resource Description Framework*) is a standard data model for representing data as triples.

Primer

- <http://www.w3.org/TR/rdf-primer/>
- <http://www.w3.org/TR/rdf11-primer/>
- **** <http://www.w3.org/TR/rdf11-primer/#section-Introduction> ****
- <http://www.w3.org/TR/rdf11-primer/#section-vocabulary>
- <http://www.w3.org/TR/rdf11-primer/#section-data-model>

Concepts

- <http://www.w3.org/TR/rdf-concepts/>
- <http://www.w3.org/TR/rdf11-concepts/>
- <http://www.w3.org/TR/rdf11-concepts/#data-model>
- <http://www.w3.org/TR/rdf11-concepts/#change-over-time>
- <http://www.w3.org/TR/rdf11-concepts/#entailment>
- <http://www.w3.org/TR/rdf11-concepts/#section-blank-nodes>
- <http://www.w3.org/TR/rdf11-concepts/#xsd-datatypes>
- <http://www.w3.org/TR/rdf11-concepts/#section-html>

Useful Resources

- “Linked Data Patterns: A pattern catalogue for modelling, publishing, and consuming Linked Data” <http://patterns.dataincubator.org/book/>

See also: *RDF Triplestores*

RDF Interfaces

Standard: <http://www.w3.org/TR/rdf-interfaces/>

Docs: <http://www.w3.org/TR/rdf-interfaces/#high-level-api>

RDF Interfaces is an *Open Source* standard for *RDF* APIs (e.g. as implemented by RDF libraries and *RDF Triplestores*).

- `createBlankNode` -> `BlankNode`
- `createNamedNode` -> `NamedNode`
- `createLiteral` -> `Literal`

- `createTriple` -> `Triple (RDFNode s, RDFNode p, RDFNode, o)`
- `createGraph` -> `[]Triple`
- `createAction` -> `TripleAction (TripleFilter, TripleCallback)`
- `createProfile` -> `Profile`
- `createTermMap` -> `TermMap`
- `createPrefixMap` -> `PrefixMap`

Implementations of RDF Interfaces:

- Javascript and/or Node.js implementations of RDF Interfaces:

http://www.w3.org/community/rdfjs/wiki/Comparison_of_RDFJS_libraries

- <https://github.com/Acubed/node-rdf>
- <https://github.com/antoniogarrote/rdfstore-js>
- <https://github.com/bergos/rdf-ext>
 - * **Src:** <https://github.com/rdf-ext/rdf-ext>
 - * **Standard:** <https://github.com/rdf-ext/rdf-ext-spec>
 - * **Standard:** <https://github.com/rdf-ext/rdf-ext-spec/blob/gh-pages/API.md>

- *RDFLib* (python) mappings to RDF Interfaces:

- `BlankNode` -> `rdflib.term.BNode`
- `NamedNode` -> `rdflib.term.URIRef, rdflib.term.Variable ? TODO`
- `Literal` -> `rdflib.term.Literal`
- `Triple` -> `tuple()`
- `Graph` -> `rdflib.graph.Graph, rdflib.graph.ConjunctiveGraph, rdflib.graph.QuotedGraph, list()`
- `Action` -> _____ `TODO`
- `TripleFilter / TripleCallback` -> `rdflib.store.TripleAddedEvent`, ``rdflib.store.TripleRemovedEvent`
- <https://rdflib.readthedocs.io/en/latest/apidocs/rdflib.html#rdflib.term.Node>
- `Profile` -> _____ `TODO`
- `TermMap` -> _____ `TODO`
- `PrefixMap` -> `rdflib.namespace.NamespaceManager` <https://rdflib.readthedocs.io/en/latest/apidocs/rdflib.html#rdflib.namespace.NamespaceManager>

Note: `rdflib` is not order-preserving at this time, because internally Graphs are represented as `dict` and not yet `collections.OrderedDict` (for which there is a now C-implementation in the Python 3.5 standard library); so output may not be in the same sequence as input (or a `rdflib.store.Store`, even) even when there are no changes made to the graph.

- It would be preferable to maintain the input source order (though, especially for large distributed queries which merge triples into one context, sorted / source order is not a good assumption to make).
- `rdf:List` are ordered.
 - * `rdf:List` with *Turtle / N3*: `:examplePredicate ["uno"@es, "one"@en] ;)`

- http://www.w3.org/TR/rdf-schema/#ch_list
 - `rdf:first`, `rdf:rest`, `rdf:nil`: “RDFS does not require that there be only one first element of a list-like structure, or even that a list-like structure have a first element.”
- * `rdf:List` with *JSON-LD* @context:
- <http://www.w3.org/TR/json-ld/#lists-and-sets>
 - <http://www.w3.org/TR/json-ld/#sets-and-lists>
 - `{"@context": {"attr": {"@container": "@list"}}`
 - `{"attr": {"@list": ["one", "uno"]}}`
-

N-Triples

Wikipedia: <https://en.wikipedia.org/wiki/N-Triples>

Standard: <http://www.w3.org/TR/n-triples/>

Extension: `.nt`

MIME Type: `application/n-triples`

N-Triples is a standard for serializing *RDF* triples to text.

RDF/XML

Wikipedia: <https://en.wikipedia.org/wiki/RDF/XML>

Standard: <http://www.w3.org/TR/rdf-syntax-grammar/>

Extension: `.rdf`

MIME Type: `application/rdf+xml`

RDF/XML is a standard for serializing *RDF* as *XML*.

TriX

Wikipedia: [https://en.wikipedia.org/wiki/TriX_\(syntax\)](https://en.wikipedia.org/wiki/TriX_(syntax))

- <http://www.w3.org/2004/03/trix/rdfg-1/>

TriX is a standard which extends the *RDF/XML* *RDF* serialization standard with named graphs.

N3

Wikipedia: <https://en.wikipedia.org/wiki/Notation3>

Standard: <http://www.w3.org/TeamSubmission/n3/>

Extension: `.n3`

MIME Type: `text/n3`

N3 (*Notation3*) is a standard which extends the *Turtle* *RDF* serialization standard with a few extra features.

- => implies (useful for specifying production rules)

Turtle

Wikipedia: [https://en.wikipedia.org/wiki/Turtle_\(syntax\)](https://en.wikipedia.org/wiki/Turtle_(syntax))

Standard: <http://www.w3.org/TR/turtle/>

Extension: `.ttl`

MIME type: `text/turtle`

Turtle is a standard for serializing *RDF* triples into human-readable text.

TriG

Wikipedia: [https://en.wikipedia.org/wiki/TriG_\(syntax\)](https://en.wikipedia.org/wiki/TriG_(syntax))

Standard: <http://www.w3.org/TR/trig/>

Extension: `.trig`

MIME Type: `application/trig`

TriG (...) extends the *Turtle RDF* standard to allow multiple named graphs to be expressed in one file (as triples with a named graph IRI (“quads”)).

Triples without a specified named graph are, by default, part of the “Default Graph”.

RDFa

Wikipedia: <https://en.wikipedia.org/wiki/RDFa>

Homepage: <http://www.w3.org/2001/sw/wiki/RDFa>

Standard: <http://www.w3.org/TR/rdfa-core/>

Standard: <http://www.w3.org/TR/rdfa-lite/>

Standard: <http://www.w3.org/TR/html-rdfa/>

Standard: <http://www.w3.org/TR/rdfa-syntax/>

Standard: <https://www.w3.org/2011/rdfa-context/rdfa-1.1>

Docs: <http://www.w3.org/TR/rdfa-primer/>

RDFa (*RDF in attributes*) is a standard for storing structured data (*RDF* triples) in *HTML*, (*XHTML*, *HTML5*) attributes.

Schema.org structured data can be included in an HTML page as RDFa.

RDFa 1.1 Core Context

Standard: <https://www.w3.org/2011/rdfa-context/rdfa-1.1>

Standard: <http://www.w3.org/2013/json-ld-context/rdfa11>

Docs: <https://github.com/RDFLib/rdfliib/blob/master/rdfliib/plugins/parsers/pyRdfa/initialcontext.py>

The RDFa 1.1 Core Context defines a number of commonly used vocabulary namespaces and URIs (*prefix mappings*).

An example *RDFa HTML5* fragment with vocabularies drawn from the *RDFa 1.1 Core Context*:

```
<div vocab="schema: http://schema.org/">
  <div typeof="schema:Thing">
    <span property="schema:name">RDFa 1.1 JSON-LD Core Context</span>
    <a property="schema:url">http://www.w3.org/2013/json-ld-context/rdfa11</a>
  </div>
</div>
```

An example *JSON-LD* document with the *RDFa 1.1 Core Context*:

```
{ "@context": "http://www.w3.org/2013/json-ld-context/rdfa11",
  "@graph": [
    { "@type": "schema:Thing"
      "schema:name": "RDFa 1.1 JSON-LD Core Context",
      "schema:url": "http://www.w3.org/2013/json-ld-context/rdfa11"
    }
  ]
}
```

Note: *Schema.org* is included in the *RDFa 1.1 Core Context*.

Schema.org does, in many places, reimplement other vocabularies e.g. for consistency with Schema.org/DataType s like schema.org/Number.

There is also *Schema.org RDF*, which, for example maps `schema:name` to `rdfs:label`; and *OWL*.

JSON-LD

Wikipedia: <https://en.wikipedia.org/wiki/JSON-LD>

Homepage: <http://json-ld.org/>

Standard: <http://www.w3.org/TR/json-ld/>

Docs: <http://manu.sporny.org/2014/json-ld-origins-2/>

JSON-LD (*JSON Linked Data*) is a standard for expressing *RDF Linked Data* as *JSON*.

JSON-LD specifies a `@context` for regular JSON documents which maps JSON attributes to URIs with datatypes and, optionally, languages.

- <http://json-ld.org/playground/>

RDFS

Wikipedia: https://en.wikipedia.org/wiki/RDF_Schema

Standard: <http://www.w3.org/TR/rdf-schema/>

Namespace: <http://www.w3.org/2000/01/rdf-schema#>

xmlns: @prefix rdfs: <<http://www.w3.org/2000/01/rdf-schema#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/rdfs>

RDFS (*RDF Schema*) is an *RDF* standard for classes and properties.

A few notable RDFS classes:

- `rdfs:Resource` (everything in RDF)
- `rdfs:Literal` (strings, integers)
- `rdfs:Class`

A few notable / frequently used properties:

- `rdfs:label`
- `rdfs:comment`
- `rdfs:seeAlso`
- `rdfs:domain`
- `rdfs:range`
- `rdfs:subPropertyOf`

OWL builds upon many RDFS concepts.

DCMI

Wikipedia: https://en.wikipedia.org/wiki/Dublin_Core

Wikipedia: https://en.wikipedia.org/wiki/Dublin_Core#DCMI_Metadata_Terms

Namespace: <http://purl.org/dc/terms>

xmlns:@prefix dcterms: <<http://purl.org/dc/terms>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/dcterms>

Namespace: <http://purl.org/dc/dcmitype/>

xmlns:@prefix dctype: <<http://purl.org/dc/dcmitype/>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/dctype>

DCTYPES (*Dublin Core Types*) and DCTERMS (*Dublin Core Terms*) are standards for common types, classes, and properties that have been mapped to *XML* and *RDF*.

EARL

Standard: <https://www.w3.org/TR/EARL10/>

Namespace: <http://www.w3.org/ns/earl#>

xmlns:@prefix earl: <http://www.w3.org/ns/earl#>

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/earl>

W3C EARL (Evaluation and Reporting Language) is an RDFS vocabulary for automated, semi-automated, and manual test results.

- The JSON-LD Implementation test results are expressed with EARL:

<http://json-ld.org/test-suite/>

<http://json-ld.org/test-suite/reports/>

RDF Data Cubes

Standard: <http://www.w3.org/TR/vocab-data-cube/>

Namespace: <http://purl.org/linked-data/cube#>

xmlns: @prefix qb: <<http://purl.org/linked-data/cube#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/qb>

RDF Data Cubes vocabulary is an *RDF* standard vocabulary for expressing linked multi-dimensional statistical data and aggregations.

- Data Cubes have dimensions, attributes, and measures
- Pivot tables and crosstabulations can be expressed with RDF Data Cubes vocabulary

SKOS

Wikipedia: https://en.wikipedia.org/wiki/Simple_Knowledge_Organization_System

Standard: <http://www.w3.org/TR/skos-reference/>

Standard: <http://www.w3.org/TR/skos-reference/skos.html>

Namespace: <http://www.w3.org/2004/02/skos/core#>

xmlns: @prefix skos: <<http://www.w3.org/2004/02/skos/core#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/skos>

SKOS (*Simple Knowledge Organization System*) is an *RDF* standard vocabulary for linking concepts and vocabulary terms.

XKOS

Homepage: <http://www.ddialliance.org/Specification/RDF/XKOS>

Standard: <http://rdf-vocabulary.ddialliance.org/xkos.html>

Source: <https://github.com/linked-statistics/xkos>

Namespace: <http://rdf-vocabulary.ddialliance.org/xkos#>

xmlns: @prefix xkos: <<http://rdf-vocabulary.ddialliance.org/xkos#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/xkos>

XKOS (*Extended Knowledge Organization System*) is an *RDF* standard which extends *SKOS* for linking concepts and statistical measures.

FOAF

Wikipedia: [https://en.wikipedia.org/wiki/FOAF_\(ontology\)](https://en.wikipedia.org/wiki/FOAF_(ontology))

Homepage: <http://www.foaf-project.org/>

Standard: <http://xmlns.com/foaf/spec/>

Namespace: <http://xmlns.com/foaf/0.1/>

xmlns: @prefix foaf: <<http://xmlns.com/foaf/0.1/>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/foaf>

FOAF (*Friend of a Friend*) is an *RDF* standard vocabulary for expressing social networks and contact information.

SHACL

Standard: <https://www.w3.org/TR/shacl/>

Namespace: <http://www.w3.org/ns/shacl#>

xmlns:@prefix sh: <<http://www.w3.org/ns/shacl#>> .

LOVLink:

W3C SHACL (Shapes Constraint Language) is a language for describing *RDF* and *RDFS* graph shape constraints.

- SHACL relaxes specific RDFS restrictions: <https://www.w3.org/TR/shacl/#shacl-rdfs>
- Required RDFS / *OWL Entailment* can be specified in SHACL with the `sh:entailment` property and e.g. *SPARQL* 1.1 entailment IRIs. <https://www.w3.org/TR/shacl/#entailment>
- <https://github.com/TopQuadrant/shacl>

SIOC

Wikipedia: https://en.wikipedia.org/wiki/Semantically-Interlinked_Online_Communities

Homepage: <http://www.sioc-project.org/>

Namespace: <http://rdfs.org/sioc/ns#>

xmlns:@prefix sioc: <<http://rdfs.org/sioc/ns#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/sioc>

SIOC (*Semantically Interlinked Online Communities*) is an *RDF* standard for online social networks and resources like blog, forum, and mailing list posts.

OA

Homepage: <http://www.openannotation.org/>

Standard: <http://www.openannotation.org/spec/core/>

Namespace: <http://www.w3.org/ns/oa#>

xmlns:@prefix oa: <<http://www.w3.org/ns/oa#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/oa>

OA (*Open Annotation*) is an *RDF* standard for commenting on anything with a URI.

Features:

- Web Annotation: https://en.wikipedia.org/wiki/Web_annotation
- Comment on any resource with a (stable) URI
- Comment on text fragments
- Comment on SVG items

Implementations:

- <https://github.com/hypothesis/h> (Python, Pyramid)

- <https://github.com/openannotation/annotator> (<http://annotatorjs.org/>)

Schema.org

Wikipedia: <https://en.wikipedia.org/wiki/Schema.org>

Homepage: <https://schema.org>

Download: <https://schema.org/version/latest/>

Source: <https://github.com/schemaorg/schemaorg>

Source: <https://github.com/schemaorg/schemaorg/tree/sdo-phobos/data/releases/2.2>

Docs: <http://dataliberate.com/2016/02/evolving-schema-org-in-practice-pt1-the-bits-and-pieces/>

Issues: <https://github.com/schemaorg/schemaorg/issues>

IssueLabels: <https://github.com/schemaorg/schemaorg/labels>

Schema.org is a vocabulary for expressing structured data on the web.

Schema.org can be expressed as microdata, *RDF*, *RDFa*, and *JSON-LD*.

- <http://> – vocab *URIs*
- <https://> – docs, *HTTP STS* (?)
- <https://schema.org/docs/full.html>
- <https://schema.org/docs/schemas.html>
- <https://schema.org/docs/releases.html>
- <https://www.w3.org/wiki/WebSchemas>
- <https://www.w3.org/wiki/WebSchemas/SchemaDotOrgProposals>
- <https://www.w3.org/wiki/WebSchemas/Accessibility>
- “Schema.org: Evolution of Structured Data on the Web” (2015) <https://queue.acm.org/detail.cfm?id=2857276>
- “Evolving Schema.org in Practice Pt1: The Bits and Pieces” (2016) <http://dataliberate.com/2016/02/evolving-schema-org-in-practice-pt1-the-bits-and-pieces/>
- *RDFa*
 - <https://github.com/schemaorg/schemaorg/blob/sdo-callisto/data/schema.rdfa>
 - <https://raw.githubusercontent.com/schemaorg/schemaorg/sdo-callisto/data/schema.rdfa>
- *JSON-LD*
 - <https://github.com/schemaorg/schemaorg/blob/sdo-callisto/data/releases/3.2/all-layers.jsonld>
 - <https://github.com/schemaorg/schemaorg/raw/sdo-callisto/data/releases/3.2/all-layers.jsonld>

Note: The <https://schema.org/> site is served over HTTPS, but the schema.org terms are HTTP URIs

Schema.org RDF

xmlns:@prefix schema: <<http://schema.org/>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/schema>

Standard: https://schema.org/docs/schema_org_rdfa.html [*RDFa*]

- *RDFa*
 - <https://github.com/schemaorg/schemaorg/blob/sdo-callisto/data/schema.rdfa>
 - <https://raw.githubusercontent.com/schemaorg/schemaorg/sdo-callisto/data/schema.rdfa>

Schema.org TopBraid RDF

Homepage: <http://topbraid.org/schema/>

Docs: <http://topbraid.org/schema/>

xmlns:@prefix schema: <<http://schema.org/>> .

xmlns:@prefix schemax: <<http://topbraid.org/schemax/>> .

TopBraid maintains more complete *OWL RDF* transformations of *Schema.org*.

- <http://topbraid.org/schema/schema.rdf> *RDF/XML*
- <http://topbraid.org/schema/schema.ttl> *Turtle*
- <http://topbraid.org/schema/schema-single-range.ttl> *Turtle* with only one type per range

SPARQL

Wikipedia: <https://en.wikipedia.org/wiki/SPARQL>

Standard: <http://www.w3.org/TR/sparql11-overview/>

Standard: <http://www.w3.org/TR/sparql11-query/>

Standard: <http://www.w3.org/TR/sparql11-update/>

Standard: <http://www.w3.org/TR/sparql11-entailment/>

Standard: <http://www.w3.org/TR/sparql11-federated-query/>

SPARQL is a text-based query and update language for *RDF* triples (and quads).

- <http://www.w3.org/wiki/SparqlImplementations>
- <http://www.w3.org/2009/sparql/implementations/#sparql11-entailment>

Challenges:

- SPARQL query requests and responses are over HTTP; however, it's best – and often required – to build SPARQL queries with a server application, on behalf of clients.
- SPARQL default `LIMIT` clauses and paging windows could allow for more efficient caching
- See: *LDP* for more of a resource-based RESTful API that can be implemented on top of the graph pattern queries supported by SPARQL.

LDP

Spec <http://www.w3.org/TR/ldp/>

xmlns:@prefix ldp: <<http://www.w3.org/ns/ldp#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/ldp>

LDP (*Linked Data Platform*) is a standard for building *HTTP* REST APIs for *RDF Linked Data*.

- <http://www.w3.org/TR/ldp/#terms>

Features:

- *HTTP* REST API for *Linked Data Platform Containers* (LDPC) containing Linked Data Platform **Resources** (LDPR)
- Server-side *Paging*

OWL

Wikipedia: https://en.wikipedia.org/wiki/Web_Ontology_Language

Standard: <http://www.w3.org/TR/owl2-overview/>

Standard: <http://www.w3.org/TR/owl2-primer/>

Standard: <http://www.w3.org/TR/owl2-quick-reference/>

Standard: <http://www.w3.org/TR/owl2-profiles/>

xmlns:@prefix owl: <<http://www.w3.org/2002/07/owl#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/owl>

OWL (*Web Ontology Language*) layers semantics, reasoning, inference, and entailment capabilities onto *RDF* (and general logical set theory).

- OWL as *Turtle*: <http://www.w3.org/2002/07/owl#>
- https://www.w3.org/TR/owl2-quick-reference/#Names.2C_Prefixes.2C_and_Notation
- https://www.w3.org/TR/owl2-quick-reference/#OWL_2_constructs_and_axioms

A few notable OWL classes:

- owl:Class a owl:Class ; rdfs:subClassOf rdfs:Class (*RDFS*)
- owl:Thing a owl:Class – **universal class**
- owl:Nothing a owl:Class – **empty class**
- owl:Restriction a rdfs:Class ; rdfs:subClassOf owl:Class

A few OWL Property types:

- owl:DatatypeProperty
- owl:ObjectProperty
- owl:ReflexiveProperty
- owl:IrreflexiveProperty
- owl:SymmetricProperty
- owl:TransitiveProperty
- owl:FunctionalProperty
- owl:InverseFunctionalProperty
- owl:OntologyProperty
- owl:AnnotationProperty
- owl:AsymmetricProperty

<https://en.wikipedia.org/wiki/Cardinality>

- `owl:minCardinality`
- `owl:cardinality`
- `owl:maxCardinality`

.

- `owl:intersectionOf`
- `owl:unionOf`
- `owl:complementOf`
- `owl:oneOf`

.

- `owl:allValuesFrom`
- `owl:someValuesFrom`

.

<https://www.w3.org/2002/07/owl#>

<https://www.w3.org/TR/owl2-quick-reference/>

PROV

Homepage: http://www.w3.org/2011/prov/wiki/Main_Page

Standard: <http://www.w3.org/ns/prov.owl>

Standard: <http://www.w3.org/TR/prov-overview/>

Standard: <http://www.w3.org/TR/prov-primer/>

Standard: <http://www.w3.org/TR/prov-o/>

Namespace: <http://www.w3.org/ns/prov#>

xmlns:@prefix prov: <<http://www.w3.org/ns/prov#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/prov>

PROV (*Provenance*) ontology is an *OWL RDF* standard for expressing data provenance (who, what, when, and how, to a certain extent).

- https://en.wikipedia.org/wiki/Provenance#Data_provenance

DBpedia

Homepage: <http://wiki.dbpedia.org/Ontology>

Namespace: <http://dbpedia.org/ontology/>

xmlns:@prefix dbpedia-owl: <<http://dbpedia.org/ontology/>> .

LOVLink: <http://dbpedia.org/ontology/>

DBpedia is an *OWL RDF* vocabulary for expressing structured data from Wikipedia sidebar infoboxes.

DBpedia is currently the most central (most linked to and from) *RDF* vocabulary. (see: *LODCloud*)

Example:

- <http://dbpedia.org/page/DBpedia>
- <http://en.wikipedia.org/wiki/DBpedia>
- <https://en.wikipedia.org/wiki/DBpedia>
- <https://en.wikipedia.org/wiki/dbpedia>
- <https://es.wikipedia.org/wiki/DBpedia>
- <http://ja.dbpedia.org/resource/DB>`\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČŽ(U+30DA)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČĜ(U+30C7)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČč(U+30A3)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČć(U+30A2)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup`
- <https://ja.wikipedia.org/wiki/DB>`\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČŽ(U+30DA)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČĜ(U+30C7)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČč(U+30A3)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup\protect\begin\group\immediate\write\@unused\def\MessageBreak`\let\protect\edef>Yourcommandwasignored.\MessageBreakTypeI<command><return>to replace it with another command,\MessageBreakor<return>to continue without it.\errhelp\let\def\MessageBreak'(inputenc)\def\errmessagePackageinputencError:UnicodecharãČć(U+30A2)\MessageBreaknotsetupforusewithLaTeX.`Seetheinputencpackagedocumentationforexplanation.`TypeH<return>forimmediatehelp\endgroup`

```
`TypeH<return>forimmediatehelp\endgroup
```

- <https://www.wikidata.org/wiki/Q465>
- <http://sw.cyc.com/concept/Mx4r7PXDuWpqTcmfojZNIu56eQ>

DBpedia is generated by batch extraction on a regular basis.

QUDT

Homepage: <http://www.linkedmodel.org/doc/qudt/1.1/>

Standard: <http://qudt.org/>

Docs: <http://www.linkedmodel.org/catalog/qudt/1.1/>

Namespace: <http://qudt.org/schema/qudt#>

Namespace: <http://qudt.org/1.1/schema/qudt#>

xmlns: @prefix qudt: <<http://qudt.org/schema/qudt#>> .

xmlns: @prefix qudt-1.1: <<http://qudt.org/1.1/schema/qudt#>> .

LOVLink: <http://lov.okfn.org/dataset/lov/vocabs/qudt>

QUDT (*Quantities, Units, Dimensions, and Types*) is an *RDF* standard vocabulary for representing physical units.

- QUDT is composed of a number of sub-vocabularies
- QUDT maintains conversion factors for Metric and Imperial Units

Examples:

- `qudt:SpaceAndTimeUnit`

```
qudt:SpaceAndTimeUnit
  rdf:type owl:Class ;
  rdfs:label "Space And Time Unit"^^xsd:string ;
  rdfs:subClassOf qudt:PhysicalUnit ;
  rdfs:subClassOf
    [ rdf:type owl:Restriction ;
      owl:hasValue "UST"^^xsd:string ;
      owl:onProperty qudt:typePrefix
    ] .
```

- QUDT Namespaces:

```
@prefix qudt: <http://qudt.org/schema/qudt#> .
@prefix qudt-1.1: <http://qudt.org/1.1/schema/qudt#> .
@prefix qudt-dimension: <http://qudt.org/vocab/dimension#> .
@prefix qudt-quantity: <http://qudt.org/vocab/quantity#> .
@prefix qudt-unit-1.1: <http://qudt.org/1.1/vocab/unit#> .
@prefix unit: <http://qudt.org/vocab/unit#> .
```

This diagram explains how each of the vocabularies are linked and derived: <http://www.linkedmodel.org/catalog/qudt/1.1/>

QUDT Quantities

Schema

Standard: <http://qudt.org/1.1/schema/quantity>

Namespace: <http://qudt.org/1.1/schema/quantity#>

xmlns: @prefix quantity: <<http://data.nasa.gov/qudt/owl/quantity#>> .

Turtle: [http://qudt.org/1.1/schema/OSG_quantity-\(v1.1\).ttl](http://qudt.org/1.1/schema/OSG_quantity-(v1.1).ttl)

Vocabulary

xmlns: @prefix qudt-quantity: <<http://qudt.org/1.1/vocab/quantity#>> .

Namespace: <http://qudt.org/1.1/vocab/quantity#>

Turtle: [http://qudt.org/1.1/vocab/OVG_quantities-qudt-\(v1.1\).ttl](http://qudt.org/1.1/vocab/OVG_quantities-qudt-(v1.1).ttl)

QUDT Quantities is an *RDF* schema and vocabulary for describing physical quantities.

Examples from [http://qudt.org/1.1/vocab/OVG_quantities-qudt-\(v1.1\).ttl](http://qudt.org/1.1/vocab/OVG_quantities-qudt-(v1.1).ttl) :

- qudt-quantity:Time

```
qudt-quantity:Time
  rdf:type qudt:SpaceAndTimeQuantityKind ;
  rdfs:label "Time"^^xsd:string ;
  qudt:description "Time is a basic component of the measuring system used to
↳sequence events, to compare the durations of events and the intervals between
↳them, and to quantify the motions of objects."^^xsd:string ;
  qudt:symbol "T"^^xsd:string ;
  skos:exactMatch <http://dbpedia.org/resource/Time> .
# ...
unit:SecondTime
  qudt:quantityKind qudt-quantity:Time .
```

- qudt-quantity:AreaTimeTemperature

```
qudt-quantity:AreaTimeTemperature
  rdf:type qudt:ThermodynamicsQuantityKind ;
  rdfs:label "Area Time Temperature"^^xsd:string .
# ...
unit:SquareFootSecondDegreeFahrenheit
  qudt:quantityKind qudt-quantity:AreaTimeTemperature .
```

QUDT Units

Standard: <http://qudt.org/1.1/vocab/unit>

Namespace: <http://qudt.org/1.1/vocab/unit#>

xmlns: @prefix unit: <<http://qudt.org/1.1/vocab/unit>> .

xmlns: @prefix qudt-unit-1.1: <<http://qudt.org/1.1/vocab/unit#>> .

Turtle: [http://qudt.org/1.1/vocab/OVG_units-qudt-\(v1.1\).ttl](http://qudt.org/1.1/vocab/OVG_units-qudt-(v1.1).ttl)

The *QUDT* Units Ontology is an *RDF* vocabulary defining many units of measure.

Examples:

- unit:SecondTime

```

unit:SecondTime
  rdf:type qudt:SIBaseUnit , qudt:TimeUnit ;
  rdfs:label "Second"^^xsd:string ;
  qudt:abbreviation "s"^^xsd:string ;
  qudt:code "1615"^^xsd:string ;
  qudt:conversionMultiplier
    "1"^^xsd:double ;
  qudt:conversionOffset
    "0.0"^^xsd:double ;
  qudt:symbol "s"^^xsd:string ;
  skos:exactMatch <http://dbpedia.org/resource/Second> .
# ...

```

<http://www.qudt.org/qudt/owl/1.0.0/unit/Instances.html#SecondTime>

- unit:HorsepowerElectric

[http://qudt.org/1.1/vocab/OVG_units-qudt-\(v1.1\).ttl](http://qudt.org/1.1/vocab/OVG_units-qudt-(v1.1).ttl)

```

unit:HorsepowerElectric
  rdf:type qudt:NotUsedWithSIUnit , qudt:PowerUnit ;
  rdfs:label "Horsepower Electric"^^xsd:string ;
  qudt:abbreviation "hp/V"^^xsd:string ;
  qudt:code "0815"^^xsd:string ;
  qudt:symbol "hp/V"^^xsd:string .

```

- unit:SystemOfUnits_SI

[http://qudt.org/1.1/vocab/OVG_units-qudt-\(v1.1\).ttl](http://qudt.org/1.1/vocab/OVG_units-qudt-(v1.1).ttl)

```

unit:SystemOfUnits_SI
  rdf:type qudt:SystemOfUnits ;
  rdfs:label "International System of Units"^^xsd:string ;
  qudt:abbreviation "SI"^^xsd:string ;
  qudt:systemAllowedUnit
    unit:ArcMinute , unit:Day , unit:MinuteTime , unit:DegreeAngle ,
↪unit:ArcSecond , unit:ElectronVolt , unit:RevolutionPerHour , unit:Femtometer ,
↪unit:DegreePerSecond , unit:DegreeCelsius , unit:Liter , unit:MicroFarad ,
↪unit:AmperePerDegree , unit:RevolutionPerMinute , unit:MicroHenry ,
↪unit:Kilometer , unit:Revolution , unit:Hour , unit:PicoFarad , unit:Gram ,
↪unit:DegreePerSecondSquared , unit:MetricTon , unit:CubicCentimeter ,
↪unit:SquareCentimeter , unit:CubicMeterPerHour , unit:KiloPascal ,
↪unit:DegreePerHour , unit:UnifiedAtomicMassUnit , unit:MilliHenry ,
↪unit:KilogramPerHour , unit:KiloPascalAbsolute , unit:NanoFarad ,
↪unit:RadianPerMinute , unit:RevolutionPerSecond ;
  qudt:systemBaseUnit unit:Kilogram , unit:Unitless , unit:Kelvin ,
↪unit:Meter , unit:SecondTime , unit:Mole , unit:Candela , unit:Ampere ;
  qudt:systemCoherentDerivedUnit
    unit:PerCubicMeter , unit:WattPerSquareMeter , unit:Volt ,
↪unit:WattPerMeterKelvin , unit:CoulombPerCubicMeter , unit:Becquerel ,
↪unit:WattPerSquareMeterSteradian , unit:KelvinPerSecond , unit:Gray ,
↪unit:RadianPerSecond , unit:VoltPerMeter , unit:HenryPerMeter ,
↪unit:WattPerSteradian , unit:JouleMeterPerMole , unit:CoulombMeter ,
↪unit:PerTeslaMeter , unit:Pascal , unit:LumenPerWatt ,
↪unit:KilogramMeterPerSecond , unit:SquareMeterKelvin , unit:MoleKelvin ,
↪unit:MeterKelvinPerWatt , unit:Steradian , unit:AmperePerMeter ,
↪unit:SquareMeterKelvinPerWatt , unit:JouleSecond , unit:MeterPerFarad ,
↪unit:KilogramPerSecond , unit:HertzPerTesla , unit:KilogramMeterSquared ,
↪unit:WattPerSquareMeterQuarticKelvin , unit:PerMeterKelvin ,
↪unit:JoulePerCubicMeterKelvin , unit:JoulePerSquareTesla ,
↪unit:JoulePerCubicMeter , unit:MeterPerKelvin , unit:AmperePerSquareMeter ,
↪unit:CubicCoulombMeterPerSquareJoule , unit:CoulombPerMeter , unit:Katal ,
↪unit:CubicMeter , unit:LumenSecond , unit:Coulomb , unit:MolePerKilogram ,
↪unit:CubicMeterPerKilogramSecondSquared , unit:PerMeter , unit:AmperePerRadian ,
↪unit:CoulombPerKilogram , unit:QuarticCoulombMeterPerCubicEnergy , unit:Tesla ,
↪unit:JoulePerKilogram , unit:MeterKelvin , unit:MeterPerSecond ,

```

```

qudt:systemPrefixUnit
    unit:Hecto , unit:Nano , unit:Tera , unit:Atto , unit:Kilo ,
↪unit:Yocto , unit:Yotta , unit:Deci , unit:Zepto , unit:Pico , unit:Femto ,
↪unit:Milli , unit:Micro , unit:Zetta , unit:Mega , unit:Centi , unit:Giga ,
↪unit:Peta , unit:Deca , unit:Exa ;
    skos:exactMatch <http://dbpedia.org/resource/International_System_of_Units>
↪.

```

Wikidata

Wikipedia: <https://en.wikipedia.org/wiki/Wikidata>

Homepage: <https://www.wikidata.org/>

Wikidata is an *Open Source* collaboratively edited knowledgebase.

- *DBpedia* scrapes data from Wikipedia Infoboxes periodically. *Wikidata* is a database with forms, datatypes, and alphanumeric identifiers (which do not change or redirect).
- Wikidata *SPARQL*, *RDF*, and *OWL* will be powered by *Blazegraph*.

Semantic Web Tools

Homepage: <http://www.w3.org/2001/sw/wiki/Tools>

Semantic Web Tools are designed to work with *RDF* formats.

See also: *RDF Triplestores*

CKAN

Wikipedia: <https://en.wikipedia.org/wiki/CKAN>

Homepage: <http://ckan.org/>

Source: git <https://github.com/ckan/ckan>

Source: git <https://github.com/ckan/ckan-docker>

DockerHub: <https://registry.hub.docker.com/u/ckan/ckan/>

Docs: <http://docs.ckan.org/en/latest/>

Docs: <http://docs.ckan.org/en/latest/maintaining/installing/index.html>

Docs: <http://docs.ckan.org/en/latest/maintaining/data-viewer.html>

Docs: <http://docs.ckan.org/en/latest/maintaining/paster.html>

Docs: <http://docs.ckan.org/en/latest/maintaining/linked-data-and-rdf.html>

Docs: <http://docs.ckan.org/en/latest/api/>

CKAN (*Comprehensive Knowledge Archive Network*) is an *Open Source* data repository *web application* and API written in python with support for *RDF*.

- <https://datahub.io> is powered by CKAN. *LODCloud* draws from datahub.io datasets.
- Many national data.gov sites are powered by CKAN. (e.g <https://catalog.data.gov/>)

- Many public and private data repositories are powered by CKAN.
- CKAN is currently *not yet* built on an *RDF triplestore*.
- There are Docker Dockerfiles for CKAN.

Protégé

Wikipedia: [https://en.wikipedia.org/wiki/Protégé_\(software\)](https://en.wikipedia.org/wiki/Protégé_(software))

Homepage: <http://protege.stanford.edu/>

Homepage: <http://webprotege.stanford.edu/>

Protégé is a knowledge management software application with support for *RDF*, *OWL*, and a few different reasoners.

Web Protégé is a web-based version of Protégé with many similar features.

Protégé is a Free and Open Source software tool.

RDFJS

Homepage: <http://www.w3.org/community/rdfjs/>

Src: <https://github>

RDFJS (*RDF* Javascript) is an acronym for referring to tools for working with *RDF* in the Javascript programming language.

- http://www.w3.org/community/rdfjs/wiki/Comparison_of_RDFJS_libraries
- <https://github.com/rdfjs/rdfjs.github.io>
 - <https://github.com/rdfjs/rdfjs.github.io/wiki/Relevant-Specs>
 - <https://github.com/rdfjs/rdfjs.github.io/wiki/Relevant-Specs#tangential-technologies>
- <https://github.com/rdfjs/representation-task-force/wiki>
 - Subject, Predicate, Object
 - <https://github.com/rdfjs/representation-task-force/wiki/Existing-triple-and-quad-representations#subject-predicate-object>
 - Node type matrix
 - <https://github.com/rdfjs/representation-task-force/wiki/Existing-triple-and-quad-representations#node-types>

See:

- *RDF Interfaces*
- **ref**

RDFHDT

Homepage: <http://www.rdfhdt.org/>

RDFHDT (*RDF* Header Dictionary Triples) is an optimized binary format for storing and working with very many triples in highly compressed form.

HDT-IT is a software application for working with RDFHDT datasets:

- <https://code.google.com/p/hdt-it/>
- <https://www.youtube.com/watch?v=HMPkc725sMY>

RDFLib

Wikipedia: <https://en.wikipedia.org/wiki/RDFLib>

Homepage: <https://github.com/RDFLib>

Source: <https://github.com/RDFLib/rdfliib>

Docs: <https://rdfliib.readthedocs.io/en/latest/>

RDFLib is a library (and a collection of companion libraries) for working with *RDF* in the Python programming language.

- <https://rdfliib.readthedocs.io/en/latest/gettingstarted.html>
- <https://rdfliib.readthedocs.io/en/latest/apidocs/examples.html>
- <https://rdfliib.readthedocs.io/en/latest/apidocs/rdfliib.html#module-rdfliib.resource>

Semantic Web Schema Resources

prefix.cc

Homepage: <http://prefix.cc>

Docs:

Lookup *RDF* vocabularies, classes, and properties

LOV

Homepage: <http://lov.okfn.org/>

Source: git <https://github.com/pyvandenbussche/lov>

SPARQL: <http://lov.okfn.org/dataset/lov/sparql>

Docs: <http://lov.okfn.org/dataset/lov/api>

LOV (“Linked Open Vocabularies”) is a web application for cataloging and viewing metadata of and links between vocabularies (*RDF*, *RDFS*, *OWL*)

- All of the vocabularies stored in LOV as a bubble chart:
<http://lov.okfn.org/dataset/lov/>
- LOV has a “suggest a vocabulary” feature
- Many of the vocabularies stored in LOV can also be searched or looked up from *prefix.cc*.

URIs for Units

- <https://lists.w3.org/Archives/Public/public-vocabs/2014Jan/0157.html>
 - <https://lists.w3.org/Archives/Public/public-vocabs/2015May/>
 - <https://lists.w3.org/Archives/Public/public-vocabs/2015May/thread.html>

LODCloud

Homepage: <http://lod-cloud.net>

Source: git <https://github.com/lod-cloud/datahub2void>

Datasets: <http://datahub.io/group/lodcloud>

Download: <http://lod-cloud.net/data/void.ttl>

The LOD (“Linking Open Data”) cloud diagram visualizes the nodes and edges of the *Linked Open Data Cloud*

- <http://lod-cloud.net/#history>
- <http://lod-cloud.net/versions/2014-08-30/lod-cloud.svg>

LinkedReproducibility

Hashtag: #LinkedReproducibility

Twitter: <https://twitter.com/hashtag/LinkedReproducibility>

- *Data Science*
- *Knowledge Engineering > Linked Data*
 - *Units > RDF and Units [QUDT,]*
- *Information Systems*
- ELI5: Our data is probably already aware of a cure.
 - Cure: Win/Win solution

Note: This page (`linkedreproducibility.rst`) is licensed with CC0 1.0 (Public Domain).

Please do implement these ideas and specifications.

– @westurner

StudyGraph: Document Nodes and Link Edges

We should use annotations with typed, reified edges to link between various studies with comparable and incomparable analyses. (e.g. *OpenAnnotation OA RDF OWL* with more data than threaded comments).

- PDFs are the de-facto standard for scientific publishing
- Many journals also / instead request HTML
 - PLoS, for example
- Then, a “study” is a document

- title – schema.org/name
- authors – lead first (usu.)
 - * schema.org/TODO – author, creator, contributor
- abstract
- tags/labels/keywords are **edges** to a tag/label/keyword node
 - * hierarchical
 - MESH
 - PyPI Trove Classifiers
 - * folksonomy
 - tags
 - tags often require deduplication / part of speech normalization / de-pluralization / etc
 - * Concept URIs
 - wikipedia, dbpedia, wikidata, etc
- links to *Linked Data*
 - * <https://schema.org/Dataset>
 - * *CSVW*
 - * *StructuredPremises: Premises as structured data*
- What we lack are **structured edges/relations** between the actual studies
 - ex:confirms, ex:seemsToConfirm -> ex:confirmatoryEdge
 - * [strength of association [”magnitude”]]
 - ex:reproduces, ex:seemsToReproduce -> ex:reproducibilityEdge
 - ex:refutes
 - ex:disproves
 - TODO: see the list i brainstormed -
 - frameworks for edges:
 - * *NetworkX*
 - * *RDF*
 - <http://patterns.dataincubator.org/book/qualified-relation.html>
 - <http://patterns.dataincubator.org/book/nary-relation.html>
 - * OpenCog AtomSpace (strength of association)
- Develop best practices guidelines and and/or an *RDF* schema and vocabulary (“repro:”) for linking between studies, their supporting data, and their collection methods with URIs.
 - developing vocabularies:
 - * *Semantic Web Tools*
 - * *Git, GitHub Pages*
 - * [] *Schema.org* extension vocabularies

- linked reproducibility edges:
 - * `similarTo`
 - * `concursWith`
 - * `discordantWith`
 - * `intendedToReproduce`
 - * `reproduces`
- linked reproducibility classes and properties:
 - * [x] `schema.org/MedicalStudy`, `MedicalObservationalStudy`, `MedicalTrial`
 - [] <https://github.com/twamarc/ScheMed>
 - <http://schema.org/MedicalTrialDesign>
 - <http://schema.org/DoubleBlindedTrial>
 - <http://schema.org/InternationalTrial>
 - <http://schema.org/MultiCenterTrial>
 - <http://schema.org/OpenTrial>
 - <http://schema.org/PlaceboControlledTrial>
 - <http://schema.org/RandomizedTrial>
 - <http://schema.org/SingleBlindedTrial>
 - <http://schema.org/SingleCenterTrial>
 - <http://schema.org/TripleBlindedTrial>
 - * See: <https://westurner.github.io/opengov/us/#personal-health-agenda>

TODO: - pandas 3402 -

StructuredPremises: Premises as structured data

- And then URIs for controls / study design
 - see `schema.org/MedicalTrialDesign`
 - * [] these could/should be extended to all of science
- logical premises (sequence of propositions)
- i/o sequences
 - nbformat (IPython / Jupyter notebook format)
 - * insufficient because we need stable premise permalinks (across versioned publishing URIs)
 - `#premise-1`
 - `#premise-abc398f`
- conclusions (derivations)
 - this is a computation graph
 - it should have links (edges) to the datasets

- * <https://schema.org/Dataset>
- * “ENH: Linked Datasets (RDF)” <https://github.com/pydata/pandas/issues/3402>
- figures should have links (edges) to the datasets
 - * permalinks to premises
- #TenSimpleRules for Reproducible Computational Research | <http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1003285> | <https://wrdrd.github.io/docs/consulting/data-science#tensimplerules-for-reproducible-computational-research>
- further questions
 - “downstream” studies / implementations
 - retraction management
 - decisions / policy predicated on said conclusions

LinkedMetaAnalyses

- You evaluated 10, I evaluated (the same / a different) 10 studies
 - PRISMA
 - * Homepage: <http://www.prisma-statement.org/>
 - * <http://www.prisma-statement.org/PRISMAStatement/>
 - * Checklist: <http://www.prisma-statement.org/documents/PRISMA%202009%20checklist.pdf>
 - * Flow Diagram: <http://www.prisma-statement.org/documents/PRISMA%202009%20flow%20diagram.pdf>
 - evaluation of controls
 - * “the URI says they did a Triple Blind Study, but it doesn’t sound like they had groups named just e.g. X, Y, and Z”
 - disqualified / questionable / etc
 - schema.org/MedicalTrial -> schema.org/ScientificTrial
 - C = Class (RDFS)
 - P = Property (RDFS)
 - *schema.org/*
 - * [] C: MetaAnalysis
 - [] C: CriteriaBase type
 - [] C: Criterion
 - [] C: ScientificStudy
 - [x] C: MedicalStudy
 - [] C: MedicalObservationalStudy <- ScientificObservationalStudy
 - [] C: MedicalTrial <- ScientificTrial
 - [x] C: Dataset
- When do we show?

- Deadline
- Only if you also produce your own meta-analyses
- Only if we're doing Open Access (as required by stipulations of federal funding)

RDF Example

Linked Data + Reproducibility => Linked Reproducibility

Reproducibility	---___	Linked Reproducibility
Linked Data	---/	

In *Turtle RDF* syntax:

```
:LinkedData rdf:type skos:Concept ;
  rdfs:label "Linked Data"@en ;
  schema:name "Linked Data"@en ;
  owl:sameAs <https://en.wikipedia.org/wiki/Linked_data> ;
  owl:sameAs <http://dbpedia.org/page/Linked_data> ;

  owl:sameAs <http://ja.dbpedia.org/resource/Linked_data>
  owl:sameAs <http://es.dbpedia.org/resource/Datos_enlazados> ;
  owl:sameAs <http://fr.dbpedia.org/resource/Web_des_donn%C3%A9es> ;
  owl:sameAs <http://nl.dbpedia.org/resource/Linked_data>
  owl:sameAs <http://ko.dbpedia.org/resource/_> ;
  owl:sameAs <http://wikidata.org/entity/Q515701> ;
  .

:Reproducibility a skos:Concept ;
  rdfs:label "Reproducibility"@en ;
  schema:name "Reproducibility"@en ;
  owl:sameAs <https://en.wikipedia.org/wiki/Reproducibility> ;
  owl:sameAs <http://dbpedia.org/page/Reproducibility> ;
  .

:LinkedReproducibility a skos:Concept ;
  rdfs:label "Linked Reproducibility"@en ;
  schema:name "Linked Reproducibility"@en ;
  skos:related [ :LinkedData, :Reproducibility ] ;
  .
```

CSV, CSVW, and metadata rows

- *CSV* – Comma Separated Values
- *CSVW* – CSV on the Web (*RDF*, *JSON*, *JSON-LD*)
- *RDF* – Resource Description Framework
 - A two dimensional spreadsheet containing just data values (e.g. *CSV*) is a projection of an RDF graph which can be expressed as *CSVW*.

A classic data table with 1 metadata header row (column label):

Table 13.1: Table with 1 metadata header row

column label	sample	date	width	height
	1	2016-06-19T06:28:49-0500	20.0	30.0
	2	2016-06-19T06:29:22-0500	40.0	50.0
	3	2016-06-19T06:29:48-0500	60.0	70.0

A data table with 7 metadata header rows (column label, property URI path, DataType, unit, accuracy, precision, significant figures):

Table 13.2: Table with 7 metadata header rows

column label	sample	date	width	height
property URI path	schema.org/name	schema.org/dateCreated	schema.org/height , schema.org/value	schema.org/width , schema.org/value
	schema.org/DataType	schema.org/Integer	schema.org/Float	schema.org/Float
Unit			unit:Meter	unit:Meter
accuracy				
precision				
significant figures			*.1	*.1
	1	2016-06-19T06:28:49-0500	20.0	30.0
	2	2016-06-19T06:29:22-0500	40.0	50.0
	3	2016-06-19T06:29:48-0500	60.0	70.0

- https://en.wikipedia.org/wiki/Significant_figures
- https://en.wikipedia.org/wiki/Accuracy_and_precision
- *QUDT* unit:
 - <http://www.qudt.org/qudt/owl/1.0.0/unit/Instances.html#Meter>
 - <http://prefix.cc/unit:Meter>
 - These should be full *URIs* (*URIs for Units*)

References

- TODO: @westurner
 - reddit
 - twitter
 - hackernews

Folders and Labels

- Folders are exclusive.
- Labels are inclusive.
- #Hashtags are labels.
- Folders form a tree which may be flat.
- Labels can form a tree but are otherwise flat.
- Folder path: a/b/c
- Nested label: a.b.c

Citation Metadata

Bibliographic citations can take many forms.

Citations are most useful in a structured form (with a **schema**).

- *DCMI*
- *OAI-PMH*
- *Schema.org* CreativeWork

Citations in the bibliography or references or resources section of a textual document must be parsed in order to derive a **citation graph**.

- *Zotero*
- *Mendeley*
- <http://citationstyles.org/>

Many *impact statistics* are derived from *graph* metrics according to citation frequency (and, by implication, things like centrality).

See:

- *Knowledge Engineering > Graphs*
- *LinkedReproducibility > StudyGraph: Document Nodes and Link Edges*
- Zotero and Schema.org RDFa

Search engines

- *Knowledge Engineering > Search Engine Indexing*
- Query syntax
- Case sensitivity
- *Unicode* symbols (Zero, Zerö, Zerø, ero)
- Stemming & Spelling Correction
 - “walking” -> walk -> walk, walking, walkers, walked
- Fuzzy matching
 - *ElasticSearch*
 - * “Typos and Mispelings” > “Fuzziness” <https://www.elastic.co/guide/en/elasticsearch/guide/current/fuzziness.html>
 - String distance (*hamming distance*)
 - Substitution, Insertion, Deletion (see also: Operational Transformation)
- Regional language variants
 - https://en.wikipedia.org/wiki/American_and_British_English_spelling_differences#-our.2C_-or
 - * “Colour”, “Couleur”, and “Color”
 - <https://en.wikipedia.org/wiki/Romanization>
 - * “”, “”, and “Sushi”
- String prefixes
 - Does “Apple” also return e.g. “Grapple”; or just e.g. “apples”, “appleton”, “apple pie”
- Stop words
 - a, and*, the, or*, not*
- Logical Term grouping
 - “Quoting”, (Parentheses), Logical terms (*Logic*)
 - “This one” AND “That one”
 - “This one” AND (“that one”)
 - this one AND that one
 - -this one AND that one
 - -(“this one”) AND “that one”

- (NOT “this one”) AND (“that one”)
- Search algorithms:
 - *Search Engine Indexing*
 - *Data Structures*
 - natural language
 - Full table scan (match every row every time) [very slow]
 - Document-Term graph / tree
 - * “index” non-stop words *and phrases* as graph edges
 - * “entity recognition” / “entity extraction” / “phrase extraction”
 - OpenNLP (Java), NLTK (Python), Watson
 - “Mark Twain grew up not in Hannibal, Missouri but in St Louis, Missouri.”
 - grew up
 - Mark Twain (Mark, Twain, Mark Twain)
 - Hannibal
 - Hannibal, Missouri
 - St Louis
 - St Louis, Missouri
 - * Manual Index
 - “biased”
 - <https://wrdrd.github.io/docs/genindex>

Research Tools

Mendeley

Wikipedia: <https://en.wikipedia.org/wiki/Mendeley>

- Zotero is similar to Mendeley.

Zotero

Wikipedia: <https://en.wikipedia.org/wiki/Zotero>

See:

- Zotero
- Zotero and Schema.org RDFa
- *Mendeley* is similar to Zotero.

CKAN

Src: <https://github.com/ckan>

Src: git <https://github.com/ckan/ckan>

CKAN (*Comprehensive Knowledge Archive Network*) is an open source web application for cataloging data written in Python.

- There are a number of extensions for CKAN: <http://extensions.ckan.org/>
 - ckanext-extractor can automatically extract text and metadata from datasets (including PDF). <http://extensions.ckan.org/extension/extractor/>
- see also:
- * *LinkedReproducibility > StudyGraph: Document Nodes and Link Edges*
 - * *OAI-PMH, Fedora Commons*
- ckanext-datajson can generate data.gov JSON for datasets: <http://extensions.ckan.org/extension/datajson/>

DSpace

Wikipedia: <https://en.wikipedia.org/wiki/DSpace>

Homepage: <http://www.dspace.org/>

DSpace is an open source web application for creative works and their *XML* metadata written in Java.

- DSpace supports *OAI-PMH*.
- DSpace and *Fedora Commons* are now both part of DuraSpace.

Fedora Commons

Wikipedia: https://en.wikipedia.org/wiki/Fedora_Commons

Homepage: <http://fedorarepository.org/>

Docs: <http://fedorarepository.org/features>

Docs: <https://wiki.duraspace.org/>

Docs: <https://wiki.duraspace.org/display/FEDORA4x/Fedora+4.x+Documentation>

Fedora Commons (*Fedora Repository*, *Fedora*) is an open source web application for creative works and their *XML* metadata written in Java.

- <http://fedorarepository.org/features>
- Fedora supports *OAI-PMH*.
- Fedora can index metadata with other search engines (e.g. *Solr*, *ElasticSearch*)
- There are additional frontend web applications for Fedora:
 - *Hydra*
 - *Islandora*

- Fedora Commons is the database for a number of well-known institutional repositories (e.g. book and digital asset library catalogs).

Note: Fedora Commons (“Fedora”, “Fedora Repository”) is distinct from the Fedora Linux operating system.

Fedora Commons is a Java web application which runs in a WAR container on many operating systems.

Hydra

Homepage: <https://projecthydra.org/>

Src: <https://github.com/projecthydra>

Docs: <https://wiki.duraspace.org/display/hydra/The+Hydra+Project>

Hydra is an open source web application frontend for *Fedora Commons* written in Ruby

- *Solr*
- *Blacklight*

Blacklight

Homepage: <http://projectblacklight.org/>

Src: [git https://github.com/projectblacklight/blacklight](https://github.com/projectblacklight/blacklight)

Docs: <https://github.com/projectblacklight/blacklight/wiki>

Blacklight is an open source web application written in Ruby for providing a search interface to *Solr*.

- *Hydra* indexes *Fedora Commons* metadata with *Solr*; which can be displayed with Blacklight.

Islandora

Homepage: <http://islandora.ca/about>

Src: <https://github.com/Islandora>

Docs: <http://islandora.ca/documentation>

Hydra is an open source web application frontend for *Fedora Commons* written in PHP

- *Solr*
- Drupal (PHP)
- *Islandora* indexes *Fedora Commons* metadata with *Solr*; which can be displayed with the Islandora Drupal application.

OAI-PMH

Wikipedia: https://en.wikipedia.org/wiki/Protocol_for_Metadata_Harvesting

OAI-PMH (Open Metadata Institute Protocol for Metadata Harvesting) is an *XML* over *HTTP* standard for sharing metadata about creative works with Dublin Core (*DCMI* dcterms) and other schema.

- *Fedora Commons* supports OAI-PMH.
- *DSpace* supports OAI-PMH.

CHAPTER 15

Entrepreneurship

Wikipedia: <https://en.wikipedia.org/wiki/Entrepreneurship>

- *Books*
 - The Personal MBA: Master the Art of Business

<http://personalmba.com/>
<http://personalmba.com/best-business-books/>

See also:

- *Personal Finance*
- *Small Business Consulting*
 - *Solving*
 - *Business Modeling*
 - * *ROI*
 - *Business Plans #*
- *Investing*

SBA

Wikipedia: https://en.wikipedia.org/wiki/Small_Business_Administration

Homepage: <https://www.sba.gov/>

- <https://www.sba.gov/starting-managing-business>

- <https://www.sba.gov/tools/local-assistance>
- <https://www.sba.gov/starting-business/how-start-business/business-data-statistics>

Planning

Wikipedia: <https://en.wikipedia.org/wiki/Planning>

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Planning>

- https://en.wiktionary.org/wiki/best_laid_plans

See also:

- *Business Plans* (Pessimistic, Realistic, Optimistic)
- *Business Strategies*
- *Scenarios*
- *Team Building* > *The same page*

Plan for Failure

- https://en.wikipedia.org/wiki/Business_failure
- Many businesses fail.

What is the survival rate for new businesses?

About half of all new establishments survive five years or more and about one-third survive 10 years or more.

—https://www.sba.gov/sites/default/files/FAQ_March_2014_0.pdf (p.1)

- <http://www.bls.gov/bdm/bdmage.htm>
- <http://www.bls.gov/bdm/bdagefaq.htm#6>

- Many businesses are not *profitable* in their first year(s) of operation.

Contingency plan

Wikipedia: https://en.wikipedia.org/wiki/Contingency_plan

- *Business Continuity* > *Scenarios*

Exit strategy

Wikipedia: https://en.wikipedia.org/wiki/Exit_strategy

- Cash out
- Sell to: A, B, or C
- Find an **EXIT** sign [”plan your route”]

Plan for Success

- *Business Continuity*

Mission Statement

- https://en.wikipedia.org/wiki/Mission_statement

Project

- <https://en.wikipedia.org/wiki/Project>
- [https://en.wikipedia.org/wiki/Scope_\(project_management\)](https://en.wikipedia.org/wiki/Scope_(project_management))

Goal

Wikipedia: <https://en.wikipedia.org/wiki/Goal>

- https://en.wikipedia.org/wiki/Goal_setting
- https://en.wikipedia.org/wiki/SMART_criteria

Objective

Imagine Success

- https://en.wikipedia.org/wiki/Creative_visualization
- https://en.wikipedia.org/wiki/Guided_imagery

Start at the end

- https://en.wikipedia.org/wiki/Backward_induction
- https://en.wikipedia.org/wiki/Reverse_chronology

Profitability

Wikipedia: [https://en.wikipedia.org/wiki/Profit_\(accounting\)](https://en.wikipedia.org/wiki/Profit_(accounting))

Wikipedia: https://en.wikipedia.org/wiki/Rate_of_profit

Investopedia: <http://www.investopedia.com/terms/p/profitabilityratios.asp>

Investopedia: <http://www.investopedia.com/terms/r/reinvestment.asp>

- Profit \approx Income = Revenue - Expenses (*ROI*)
- There is an incentive to re-invest revenue into the business.

- <https://en.wikipedia.org/wiki/Break-even>
- https://en.wikipedia.org/wiki/Customer_profitability (*Smoothies*)
- https://en.wikipedia.org/wiki/Profitability_index

USA Entrepreneurship

- Choosing a name
 - DNS availability
 - Search result count
 - (Multi-) state business name registration
 - <https://www.sba.gov/starting-business/choose-register-your-business>
 - “Trade Name”
 - Scope
- Market Share / Market Size
 - [https://en.wikipedia.org/wiki/Market_\(economics\)#Size_parameters](https://en.wikipedia.org/wiki/Market_(economics)#Size_parameters)
 - https://en.wikipedia.org/wiki/Market_share
- Growth potential
 - number of customers
 - marginal customer value (see: *Smoothies*)
- Legal team factors to consider
 - Already-prepared, already-tested document library
 - Multi-state network
- Registering a business
 - State of incorporation
 - *Types of business entity*

Types of business entity

Wikipedia: https://en.wikipedia.org/wiki/Types_of_business_entity#United_States

No liability protection, no double-taxation (“pass-through taxation”):

- https://en.wikipedia.org/wiki/Types_of_business_entity#United_States
- https://en.wikipedia.org/wiki/Partnership_taxation_in_the_United_States
- https://en.wikipedia.org/wiki/Sole_proprietorship
- <https://en.wikipedia.org/wiki/Partnership>

Limited liability, no double-taxation (“pass-through taxation”):

- https://en.wikipedia.org/wiki/Limited_liability

- https://en.wikipedia.org/wiki/Limited_liability_partnership#United_States (“LLP”)
- https://en.wikipedia.org/wiki/Limited_liability_limited_partnership (“LLLLP”)
- https://en.wikipedia.org/wiki/Limited_liability_company (“LLC”)
- https://en.wikipedia.org/wiki/Articles_of_organization

Types of corporations:

- https://en.wikipedia.org/wiki/Types_of_business_entity#United_States
- https://en.wikipedia.org/wiki/Nonprofit_organization
- https://en.wikipedia.org/wiki/For-profit_corporation
- https://en.wikipedia.org/wiki/Articles_of_Incorporation
- https://en.wikipedia.org/wiki/Stock_certificate
- <https://en.wikipedia.org/wiki/Minutes>
 - [https://en.wikipedia.org/wiki/Agenda_\(meeting\)#Order_of_business](https://en.wikipedia.org/wiki/Agenda_(meeting)#Order_of_business)
 - https://en.wikipedia.org/wiki/Robert%27s_Rules_of_Order
 - *Digital Stand Up Meeting*
- https://en.wikipedia.org/wiki/S_corporation
 - some LLCs and LLPs can be taxed as S corporations
 - closely held
 - pass-through income taxation (according to shareholder %)
 - no aliens allowed
 - one class of stock
 - health insurance benefits and HSAs must be included as W-2 wages if the shareholder holds greater than 2% of the outstanding shares
 - https://en.wikipedia.org/wiki/S_corporation#Qualification_for_S_corporation_status
- https://en.wikipedia.org/wiki/C_corporation
 - taxed separately (business income tax, payroll tax; and then income tax)
 - https://en.wikipedia.org/wiki/Corporate_tax_in_the_United_States#Tax_rates
 - * <https://www.irs.gov/pub/irs-pdf/i1120.pdf>
 - <https://www.irs.gov/instructions/i1120/ch02.html#d0e2707>
 - (“Tax Rate Schedule” 2015 p.17)
 - Corporate income tax rates (2015 Form 1120): 15-35% (*)
 - Personal *Income tax* rates (2015 Form 1040): 10-39.6% (*)

```
>>> ((1.0 * (1-0.35)) * (1-0.396))
0.3926
>>> ((1.0 * (1-0.15)) * (1-0.10))
0.765
```

See also:

- *Information Systems > Accounting* (* “Accountant”)

- *Small Business Consulting > ROI*

Economics

<https://en.wikipedia.org/wiki/Microeconomics>

- <https://www.khanacademy.org/economics-finance-domain/microeconomics>

<https://en.wikipedia.org/wiki/Macroeconomics>

- <https://www.khanacademy.org/economics-finance-domain/macroecconomics>

https://en.wikipedia.org/wiki/Venture_capital

- <https://www.khanacademy.org/economics-finance-domain/core-finance/stock-and-bonds/venture-capital-and-capital-markets/v/raising-money-for-a-startup>

See also:

- *Investing > Financial Statement*
- *Investing > Capitalization table*
- *Investing > Equity*

Wikipedia: https://en.wikipedia.org/wiki/Personal_finance

- <https://www.reddit.com/r/personalfinance/wiki/index>

Dave Ramsey

Wikipedia: https://en.wikipedia.org/wiki/Dave_Ramsey

Homepage: <http://www.daveramsey.com/>

Twitter: <https://twitter.com/daveramsey>

- The 7 Baby Steps
<https://www.daveramsey.com/baby-steps/>
- Envelopes; physically separate envelopes
https://en.wikipedia.org/wiki/Envelope_system

Personal Budget

Wikipedia: https://en.wikipedia.org/wiki/Personal_budget

- <https://www.reddit.com/r/personalfinance/wiki/budgeting>
 - https://www.reddit.com/r/personalfinance/wiki/budgeting#wiki_tools.2Fsystems
 - * https://en.wikipedia.org/wiki/Envelope_system
 - <https://www.mint.com/how-mint-works/budgets>
 - * Spending budgets

* Savings goals

- Example / starter personal budget spreadsheets: https://www.reddit.com/r/personalfinance/wiki/tools#wiki_redditor-created_spreadsheets

Personal Income

Wikipedia: <https://en.wikipedia.org/wiki/Income>

- https://en.wikipedia.org/wiki/Personal_income_in_the_United_States

Occupational Outlook Handbook

Wikipedia: https://en.wikipedia.org/wiki/Occupational_Outlook_Handbook

Homepage: <http://www.bls.gov/ooh/>

- [...]
 - <http://www.bls.gov/ooh/computer-and-information-technology/home.htm>
 - *Software Development > Everyone Codes*

Personal Expenses

Wikipedia: <https://en.wikipedia.org/wiki/Expense>

Technically, an expense is an event in which an *Asset* is used up or a *Liability* is incurred. In terms of the *Accounting Equation*, expenses reduce owners' *Equity*.

—<https://en.wikipedia.org/wiki/Expense>

TODO: types of expenses

Personal Debt

Wikipedia: https://en.wikipedia.org/wiki/Consumer_debt

- https://en.wikipedia.org/wiki/Debt_consolidation
- https://en.wikipedia.org/wiki/Credit_counseling
- https://en.wikipedia.org/wiki/Predatory_lending
 - https://en.wikipedia.org/wiki/Predatory_lending#Abusive_or_unfair_lending_practices

Interest

Wikipedia: <https://en.wikipedia.org/wiki/Interest>

Compound interest

Wikipedia: https://en.wikipedia.org/wiki/Compound_interest

- https://en.wikipedia.org/wiki/Compound_interest#Compound_interest

Doubling time

Wikipedia: https://en.wikipedia.org/wiki/Doubling_time

Rule of 72

Wikipedia: https://en.wikipedia.org/wiki/Rule_of_72

Taxes

Wikipedia: <https://en.wikipedia.org/wiki/Tax>

See:

- Accountant (*Accounting*)

Income tax

Wikipedia: https://en.wikipedia.org/wiki/Income_tax

Property tax

Wikipedia: https://en.wikipedia.org/wiki/Property_tax

Capital gains tax

Wikipedia: https://en.wikipedia.org/wiki/Capital_gains_tax

Gift tax

Wikipedia: https://en.wikipedia.org/wiki/Gift_tax

- https://en.wikipedia.org/wiki/Gift_tax_in_the_United_States
 - In the USA, in 2015, you must pay gift tax on any amount over \$14,000 (per donor, per beneficiary).
 - * So, two parents, for example, must pay taxes on any amount over \$28,000.
 - Gift tax does not apply to e.g. trusts and, for example, direct payments for e.g. school tuition.
 - This affects eligibility for financial aid.
 - See also:
 - https://en.wikipedia.org/wiki/529_plan
 - https://en.wikipedia.org/wiki/Health_savings_account

Charitable contributions

Wikipedia: https://en.wikipedia.org/wiki/Charitable_contribution

Wikipedia: https://en.wikipedia.org/wiki/Charitable_organization

Wikipedia: [https://en.wikipedia.org/wiki/Charity_\(practice\)](https://en.wikipedia.org/wiki/Charity_(practice))

- In the USA, **charitable contributions to qualifying (501(c)(3)) non-profit organizations are tax deductible.**
 - https://en.wikipedia.org/wiki/Charitable_contribution_deductions_in_the_United_States
- [https://en.wikipedia.org/wiki/Charity_\(virtue\)#See_also](https://en.wikipedia.org/wiki/Charity_(virtue)#See_also)

Charity Navigator

Wikipedia: https://en.wikipedia.org/wiki/Charity_Navigator

Homepage: <http://www.charitynavigator.org/>

Twitter: <https://twitter.com/CharityNav>

AndroidApp: <https://play.google.com/store/apps/details?id=com.charitynavigator.mobile>

iOSApp: <https://itunes.apple.com/us/app/charity-navigator/id740191196>

- Top Ten Lists
<http://www.charitynavigator.org/index.cfm?bay=topten>
- Charity evaluation methodology
<http://www.charitynavigator.org/index.cfm?bay=content.view&cpid=32>
- Donate to Charity Navigator
<https://www.charitynavigator.org/index.cfm?bay=content.support>

Retirement

Wikipedia: <https://en.wikipedia.org/wiki/Retirement>

- https://en.wikipedia.org/wiki/Retirement_plans_in_the_United_States

Pension

Wikipedia: <https://en.wikipedia.org/wiki/Pension>

Defined benefit pension plan

Wikipedia: https://en.wikipedia.org/wiki/Defined_benefit_pension_plan

Social Security

Homepage: <https://www.ssa.gov/>

Wikipedia: https://en.wikipedia.org/wiki/Social_Security_Administration

Wikipedia: [https://en.wikipedia.org/wiki/Social_Security_\(United_States\)](https://en.wikipedia.org/wiki/Social_Security_(United_States))

- https://en.wikipedia.org/wiki/Social_Security_Wage_Base
- https://en.wikipedia.org/wiki/Social_Security_Trust_Fund
 - “The Trust Fund is required by law to be invested in non-marketable securities issued and guaranteed by the “full faith and credit” of the federal government. These securities earn a market rate of interest.”

401(k)

Wikipedia: [https://en.wikipedia.org/wiki/401\(k\)](https://en.wikipedia.org/wiki/401(k))

- 403(b) – non-profits
- 457(b) – governmental employers

401(k) match

Wikipedia: [https://en.wikipedia.org/wiki/401\(k\)#Technical_details](https://en.wikipedia.org/wiki/401(k)#Technical_details)

401(k) rollover

Wikipedia: [https://en.wikipedia.org/wiki/401\(k\)#Rollovers](https://en.wikipedia.org/wiki/401(k)#Rollovers)

IRA

Wikipedia: https://en.wikipedia.org/wiki/Individual_retirement_account

Roth IRA

Wikipedia: https://en.wikipedia.org/wiki/Roth_IRA

Healthcare

Wikipedia: https://en.wikipedia.org/wiki/Health_care

- https://en.wikipedia.org/wiki/Health_care_in_the_United_States

Health insurance

Wikipedia: https://en.wikipedia.org/wiki/Health_insurance

- https://en.wikipedia.org/wiki/Health_insurance_in_the_United_States
 - https://en.wikipedia.org/wiki/Health_insurance_in_the_United_States#Public_health_care_coverage
 - * https://en.wikipedia.org/wiki/Universal_health_care
 - https://en.wikipedia.org/wiki/Universal_health_coverage_by_country
 - https://en.wikipedia.org/wiki/List_of_countries_by_health_insurance_coverage
 - https://en.wikipedia.org/wiki/List_of_countries_by_total_health_expenditure_per_capita
 - * <https://westerner.github.io/opengov/us/healthcare#health-insurance-in-the-united-states>
 - https://en.wikipedia.org/wiki/Health_insurance_in_the_United_States#Private_health_care_coverage

Health Savings Account

Wikipedia: https://en.wikipedia.org/wiki/Health_savings_account

A Health Savings Account is a tax-advantaged savings account.

- An employer can contribute to a Health Savings Account pre-tax.
- An employee can contribute to a Health Savings Account pre-tax (*).
- Health Savings Account funds roll-over from year to year.

Long Term Care

Wikipedia: https://en.wikipedia.org/wiki/Long-term_care

Advance Care Directive

Wikipedia: https://en.wikipedia.org/wiki/Advance_healthcare_directive

“Advance Care Directive, Living Will, Personal Directive, Advance Directive, Advance Decision”

Estate planning

Wikipedia: https://en.wikipedia.org/wiki/Estate_planning

“Probate”

- [https://en.wikipedia.org/wiki/Administration_\(probate_law\)](https://en.wikipedia.org/wiki/Administration_(probate_law))
- https://en.wikipedia.org/wiki/Probate_court

Trust

Wikipedia: https://en.wikipedia.org/wiki/Trust_law

- https://en.wikipedia.org/wiki/United_States_trust_law
 - https://en.wikipedia.org/wiki/Uniform_Trust_Code
- https://en.wikipedia.org/wiki/United_States_trust_law#Types_of_Trusts
 - [#">https://en.wikipedia.org/wiki/Special_needs_trust #](https://en.wikipedia.org/wiki/Special_needs_trust)
 - https://en.wikipedia.org/wiki/Testamentary_trust
- <https://www.urbandictionary.com/define.php?term=trust+fund+baby>

Will and Testament

Wikipedia: https://en.wikipedia.org/wiki/Will_and_testament

- https://en.wikipedia.org/wiki/Will_and_testament#Requirements_for_creation
-

Estate tax

- https://en.wikipedia.org/wiki/Estate_tax_in_the_United_States
- <http://money.howstuffworks.com/personal-finance/personal-income-taxes/inheritance-tax1.htm>
 - Estate tax is paid by the estate (after debts are paid).

Inheritance tax

Wikipedia: https://en.wikipedia.org/wiki/Inheritance_tax

- Inheritance tax is paid by the beneficiaries. (“double taxation”)

Intestacy

Wikipedia: <https://en.wikipedia.org/wiki/Intestacy>

- If there is no will defined, the estate is “intestant”.
- https://en.wikipedia.org/wiki/Intestacy#United_States_and_Canada
 - State laws differ.
 - Not all states go by the Uniform Probate Code:
 - * https://en.wikipedia.org/wiki/Uniform_Probate_Code
 - <https://www.law.cornell.edu/uniform/probate>
 - <http://www.uniformlaws.org/shared/docs/probate%20code/upc%202010.pdf>
 - “SECTION 2-103. SHARE OF HEIRS OTHER THAN SURVIVING SPOUSE.”
 - It is more complicated than this, but, **roughly**, *if there is not a more specific will*, the first matching class takes everything (after debts are paid and then federal (and state) estate taxes):
 - .0. Surviving spouse
 - .a.1. decedents (children)
 - .a.2. parents (equally)
 - .a.3. parents’ decedents (e.g. siblings)
 - .a.4.A. 1/2 each to paternal/maternal grandparents
 - .a.4.B. 1/2 each to paternal/maternal grandparents’ decedents
 - .a.5. [... speak with an estate attorney ...]
 - .b. deceased spouse’s decedents (equally)
 - https://en.wikipedia.org/wiki/Uniform_Simultaneous_Death_Act

Note: If the default state order of succession applies (e.g. when there is no other will defined), for one matching class to gift all or part of their inheritance to others, they then must pay *both Inheritance tax and Gift tax* on any amount over the annual exclusion for the year.

Digital Will

Wikipedia: https://en.wikipedia.org/wiki/Digital_Will

Wikipedia: <https://en.wikipedia.org/wiki/Investment>

Warning: Like the rest of this documentation set, this is not to be taken as financial advice; or advice of any kind.

Basics

- <https://www.investor.gov/investing-basics>
- <http://www.fool.com/investing/basics/index.aspx>
- <http://www.investopedia.com/investing/investing-basics/>
- <http://www.investopedia.com/university/beginner/>
- http://www.wikinvest.com/site/Guide_to_Investing
- https://finance.yahoo.com/education/begin_investing

Cash flow

Wikipedia: https://en.wikipedia.org/wiki/Cash_flow

Revenue

Wikipedia: <https://en.wikipedia.org/wiki/Revenue>

- Gross Revenue

- Net Revenue: revenue after [returns, chargebacks, ...]
- <http://www.investopedia.com/ask/answers/102714/what-are-difference-between-gross-revenue-reporting-and-net-revenue-report.asp>

Accounting Equation

Wikipedia: https://en.wikipedia.org/wiki/Accounting_equation

Investopedia: <http://www.investopedia.com/terms/a/accounting-equation.asp>

- $\text{Assets} = \text{Liabilities} + \text{Equity}$
- $\text{Liabilities} = \text{Assets} - \text{Equity}$
- $\text{Equity} = \text{Assets} - \text{Liabilities}$

Equity

Wikipedia: [https://en.wikipedia.org/wiki/Equity_\(finance\)](https://en.wikipedia.org/wiki/Equity_(finance))

Investopedia: <http://www.investopedia.com/terms/e/equity.asp>

$$\text{Equity} = \text{Assets} - \text{Liabilities}$$

Asset

Wikipedia: <https://en.wikipedia.org/wiki/Asset>

Investopedia: <http://www.investopedia.com/terms/a/asset.asp>

Liability

Wikipedia: [https://en.wikipedia.org/wiki/Liability_\(financial_accounting\)](https://en.wikipedia.org/wiki/Liability_(financial_accounting))

Investopedia: <http://www.investopedia.com/terms/l/liability.asp>

Financial Statement

Wikipedia: https://en.wikipedia.org/wiki/Financial_statement

Investopedia: <http://www.investopedia.com/terms/f/financial-statements.asp>

Investopedia: <http://www.investopedia.com/university/financialstatements/>

-

Capitalization table

Wikipedia: https://en.wikipedia.org/wiki/Capitalization_table

Investopedia: <http://www.investopedia.com/terms/c/capitalization-table.asp>

A Capitalization table (*Cap table*) lists ownership stakes in a [new] venture.

There are a number of *SaaS web applications* for managing cap tables (and, most importantly, generating valuation reports).

- eShares

<https://esharesinc.com/>

- CapShares

<https://www.capshare.com/>

Option

Wikipedia: [https://en.wikipedia.org/wiki/Option_\(finance\)](https://en.wikipedia.org/wiki/Option_(finance))

Investopedia: <http://www.investopedia.com/terms/o/option.asp>

Investopedia: <http://www.investopedia.com/university/options/option.asp>

Stock Option

Wikipedia: [https://en.wikipedia.org/wiki/Option_\(finance\)](https://en.wikipedia.org/wiki/Option_(finance))

Investopedia: <http://www.investopedia.com/terms/s/stockoption.asp>

A stock option is a type of *Option*.

Employee Stock Option

Wikipedia: https://en.wikipedia.org/wiki/Employee_stock_option

SAFE

Homepage: <https://www.ycombinator.com/documents/>

Docs: https://www.ycombinator.com/docs/SAFE_Primer.rtf

Docs: https://www.ycombinator.com/docs/SAFE_Cap.rtf

Docs: https://www.ycombinator.com/docs/SAFE_Discount.rtf

Docs: https://www.ycombinator.com/docs/SAFE_Cap_Discount.rtf

Docs: https://www.ycombinator.com/docs/SAFE_MFN.rtf

- <https://blog.ycombinator.com/announcing-the-safe-a-replacement-for-convertible-notes>
- <https://www.quora.com/Y-Combinator/What-are-the-pros-and-cons-of-YCs-new-SAFE-securities-versus-convertible-notes>

- <https://www.cooleygo.com/documents/y-combinator-safe-financing-document-generator/>

Investing Strategies

Wikipedia: https://en.wikipedia.org/wiki/Template:Financial_markets_navigation

Investing Styles

Wikipedia: https://en.wikipedia.org/wiki/Investment_style

Investopedia: http://www.investopedia.com/terms/i/investing_style.asp

Investopedia:

Value Investing

Wikipedia: https://en.wikipedia.org/wiki/Value_investing

Investopedia: <http://www.investopedia.com/terms/v/valueinvesting.asp>

Investopedia: <http://www.investopedia.com/university/value-investing/>

See:

- *Financial Statement*

Buy and Hold

Wikipedia: https://en.wikipedia.org/wiki/Buy_and_hold

- *Value investing* is a Buy and Hold investing strategy / style.

Day trading

Wikipedia: https://en.wikipedia.org/wiki/Day_trading

- Daytrading is not *Buy and Hold* investing

Algorithmic trading

Wikipedia: https://en.wikipedia.org/wiki/Algorithmic_trading

See:

- *Quantopian (Quantopian Open)*

High Frequency Trading

Wikipedia: https://en.wikipedia.org/wiki/High-frequency_trading

Stock Market Crash

Wikipedia: https://en.wikipedia.org/wiki/Stock_market_crash

- https://en.wikipedia.org/wiki/Wall_Street_Crash_of_1929 (“Black Tuesday”)
 - https://en.wikipedia.org/wiki/Great_Depression
- https://en.wikipedia.org/wiki/Financial_crisis_of_2007%E2%80%932008
- https://en.wikipedia.org/wiki/2010_Flash_Crash

Recession

Wikipedia: <https://en.wikipedia.org/wiki/Recession>

- <https://en.wikipedia.org/wiki/Recession#Predictors>
- https://en.wikipedia.org/wiki/Recession#Stock_market

Quantitative analyst

Wikipedia: https://en.wikipedia.org/wiki/Quantitative_analyst

See also:

- *Data Science* (for more domain-independent analyses)

Tools

pandas

Wikipedia: [https://en.wikipedia.org/wiki/Pandas_\(software\)](https://en.wikipedia.org/wiki/Pandas_(software))

Homepage: <http://pandas.pydata.org/>

Src: <https://github.com/pydata/pandas>

PyPI: <https://pypi.python.org/pypi/pandas>

Docs: <http://pandas.pydata.org/pandas-docs/stable/>

Docs: <http://pandas.pydata.org/pandas-docs/stable/install.html>

Docs: <http://pandas.pydata.org/pandas-docs/stable/10min.html>

Docs: <https://pandas-docs.github.io/pandas-docs-travis/whatsnew.html>

Docs: <https://pandas-docs.github.io/pandas-docs-travis/contributing.html>

Pandas (Python data analysis) is a library for *Data Science* (and statistics).

- Pandas requires Cython, NumPy, and a number of additional libraries.

```
conda install pandas
```

– <https://github.com/westurner/notebooks/blob/gh-pages/install.sh>

- <http://pandas.pydata.org/pandas-docs/stable/ecosystem.html>

pandas-datareader

Src: <https://github.com/pydata/pandas-datareader>

Docs: <https://pandas-datareader.readthedocs.io/en/latest/>

- Yahoo! Finance
- Google Finance
- FRED
- Fama/French
- World Bank
- OECD
- Eurostat
- EDGAR Index

Quandl

Wikipedia: <https://en.wikipedia.org/wiki/Quandl>

Homepage: <https://www.quandl.com/>

Src: <https://github.com/quandl>

Docs: <https://www.quandl.com/docs/api>

quandl-python

Homepage: <https://www.quandl.com/tools/python>

PyPI: <https://pypi.python.org/pypi/Quandl>

Src: <https://github.com/quandl/quandl-python>

Docs: <https://www.quandl.com/help/python>

- quandl-python requires NumPy, which is really easy to install with Conda.

```
conda install numpy pandas requests pip
pip install quandl-python
```

QuantLib

Wikipedia: <https://en.wikipedia.org/wiki/QuantLib>

Homepage: <http://quantlib.org/>

Twitter: <https://twitter.com/quantlib>

Src: <https://github.com/lballabio/QuantLib>

Docs: <http://quantlib.org/docs.shtml>

Docs: <http://quantlib.org/install.shtml>

Quantlib is a library for *Quantitative analysis* written in C++.

- QuantLib-Python is one wrapper library / language binding for QuantLib.
- There are many language bindings for QuantLib:
<http://quantlib.org/extensions.shtml>

pyql

Src: <https://github.com/enthought/pyql>

pyql is a Cython package for working with parts of *QuantLib*.

Quantopian

Wikipedia: <https://en.wikipedia.org/wiki/Quantopian>

Homepage: <https://www.quantopian.com/>

Twitter: <https://twitter.com/quantopian>

Src: <https://github.com/quantopian>

Docs: <https://www.quantopian.com/help>

Docs: <https://www.quantopian.com/tutorials/getting-started>

Docs: <https://www.quantopian.com/tutorials/>

Quantopian is a crowd-sourced hedge fund.

- Quantopian offers free hosted IPython notebooks with *pandas*, *Zipline*, and minutely data from 2002 for algorithmic backtesting and live-trading.
 - <https://www.quantopian.com/posts/live-minutely-data-new-to-quantopian>
- You own your intellectual property.
- You can choose to share your research and algorithms with the community; who can clone and modify at will.
 - Sample Mean Reversion algorithm: <https://www.quantopian.com/algorithms/56f32bbf633c20776d000108>
 - 100 most-popular cloned algorithms: <https://www.quantopian.com/posts/community-algorithms-migrated-to-quantopian-2>

Quantopian Open

Homepage: <https://www.quantopian.com/open>

- <https://www.quantopian.com/open/rules>
- <https://www.quantopian.com/leaderboard>
- <https://www.quantopian.com/fund>

Zipline

Src: <https://github.com/quantopian/zipline>

PyPI: <https://pypi.python.org/pypi/zipline>

Docs: <http://www.zipline.io/>

Zipline is an *Algorithmic trading* library written in Python.

- Zipline does backtesting. (“How would this trading algorithm have performed based upon historical data?”)
- Zipline does live-trading. (“When to buy/sell?”)
- *Quantopian* hosts zipline (and other components).

qgrid

Src: <https://github.com/quantopian/qgrid>

PyPI: <https://pypi.python.org/pypi/qgrid>

Docs: <https://qgrid.readthedocs.io/en/latest/>

qgrid is “An Interactive Grid for Sorting and Filtering [*pandas*] DataFrames in IPython Notebook.

See Also

- *Data Science > Data Visualization*

Wikipedia: https://en.wikipedia.org/wiki/Educational_technology

Hashtags: #EdTech, #EduTech

STEM

STEM (*Science, Technology, Engineering, and Mathematics*) is a broad field of R&D fields.

- Science (*Science, Data Science*)
- Technology (*Computer Science, Information Systems, Art & Design, Education Technology*)
- Engineering (*Knowledge Engineering, Software Development, Project Management*)
- Mathematics (*Data Science*)

Why STEM?

- STEM combines fields for holism
- STEM combines fields for allocation efficiency
- **STEAM** includes *Art & Design*

Why CS?

- *Computer Science* tools are foundational to *Data Science*, which is foundational to *STEM*, which is foundational to *infrastructure*.

Publishing

<https://en.wikipedia.org/wiki/Publishing>

- Project Code and Artifact Repositories

- Version Control: Git, Mercurial
- Project Forge: GitHub, BitBucket, SourceForge
- Artifacts: built packages, reports, PDFs, data files
- DOI-granting repositories (“getting a citable identifier which resolves to a *URL*” (~like a shorturl)):
 - https://en.wikipedia.org/wiki/Digital_object_identifier (*URN*)
 - <https://guides.github.com/activities/citable-code/>
 - <https://figshare.com>
 - <https://zenodo.org/>
- Data Hosting (see: *Web Distribution*):
 - HTTP Object Storage (private/public *CDN*)
 - *BitTorrent*
 - * BitTorrent + *HTTP Web Seeds*
 - **It is faster to run the code next to the data.**
 - * *HDFS, Hadoop, Hive, Cassandra, Spark*, Mesos, ibis, *DDFS*
 - * *Accumulo* iterators
 - * *SQL (Relational Database)*
 - * *SPARQL (Linked Data)*
- Publishing Tools
 - *GitHub Pages*
 - *ReadTheDocs*
 - Sphinx
 - Jupyter Notebook
 - * nbformat JSON -> HTML, HTML slides, LaTeX, PDF, ePub, MOBI
 - * nbviewer
 - documentation-tools

GitHub Pages

Homepage: <https://pages.github.com/>

GitHub Pages serves webpages from git branches.

- GitHub Pages serves from the `gh-pages` branch of project repositories
- GitHub Pages serves from the `master` branch of user and organization repos
- GitHub Pages is backed by a *CDN*
- GitHub Pages get URLs like: <https://wrdrd.github.io/>
- GitHub Pages can have URLs like: <https://wrdrd.com/>

- Adding a `./CNAME` file to a repo causes GitHub Pages to redirect URLs to the CNAME (e.g. <https://wrdrd.com/>)
- If the *DNS domain name* does not resolve, GitHub Pages still redirects to the address in the `./CNAME` file
- pgs can also serve webpages from git branches like `gh-pages` and `master`

GitLab Pages

Homepage: <https://pages.gitlab.io/>

ReadTheDocs

Homepage: <https://readthedocs.org/>

Source: git <https://github.com/rtfd/readthedocs.io>

Source: git <https://github.com/rtfd/readthedocs-docker-images>

Docs: <https://read-the-docs.readthedocs.io/en/latest/>

ReadTheDocs is a free and *Open Source* web service for publishing Sphinx documentation sets which functions like a *Continuous Integration* build server and artifact publisher.

- ReadTheDocs can rebuild HTML, PDF, and ePub from Sphinx ReStructuredText every time a commit is made to a e.g. GitHub or BitBucket repository: <https://read-the-docs.readthedocs.io/en/latest/webhooks.html>
- ReadTheDocs can build and serve localized versions for one or more languages: <https://read-the-docs.readthedocs.io/en/latest/localization.html>
- ReadTheDocs can build and serve multiple versions or vcs revisions. By default, ReadTheDocs will try appending `/en/latest/`, so these links all redirect to the first link
 - <https://read-the-docs.readthedocs.io/en/latest/>
 - <https://read-the-docs.readthedocs.io/>
 - <https://read-the-docs.rtd.org/>

Self Directed Learning

“Self-Directed Learning with Online Resources:

An independent study of challenges, opportunities and strategies for encouraging feedback between tools and resources in online learning systems” (2012)

Source: git <https://github.com/westurner/self-directed-learning>

HTML: <https://westurner.github.io/self-directed-learning/>

HTML: <https://self-directed-learning.readthedocs.io/>

HTML (zip): <https://media.readthedocs.io/htmlzip/self-directed-learning/self-directed-learning.zip>

PDF: <https://media.readthedocs.io/pdf/self-directed-learning/latest/self-directed-learning.pdf>

ePub: <https://media.readthedocs.io/epub/self-directed-learning/latest/self-directed-learning.epub>

Report Process:

- [X] Goals: overbroad scope
- [X] Generate: 1 ream of paper; 1 pack of pens
- [o] Generate, Reduce, Clarify: Bookmarks, Zotero
- [X] Reduce, Clarify: match, cluster, re-sequence sheets (2D)
- [X] Products: transcribe handwritten sheets of paper as slides
- [o] Products: glossary
- [] Products: essay form
- [o] Tools: Sphinx, ReStructuredText, *ReadTheDocs*
- [o] <https://westurner.github.io/self-directed-learning/process.html>

Online Courses

- Course Catalogs
 - *Class Central*
- *Linked Curricula Graphs* (RDF, RDFa, Schema.org/Course)

Class Central

Homepage: <https://www.class-central.com/>

Source: git <https://github.com/dhawalshah/class-central/>

OpenBadges: []

Class Central aggregates lists of *Online Courses*.

Coursera

Wikipedia: <https://en.wikipedia.org/wiki/Coursera>

Homepage: <https://www.coursera.org/>

Courses: <https://www.coursera.org/courses>

OpenBadges: []

AndroidApp: <https://play.google.com/store/apps/details?id=org.coursera.android>

iOSApp: <https://itunes.apple.com/us/app/coursera/id736535961>

Source: <https://github.com/coursera>

Docs: <https://tech.coursera.org/app-platform/>

Coursera is a platform for *Online Courses*.

edX

Wikipedia: <https://en.wikipedia.org/wiki/EdX>

Homepage: <https://www.edx.org/>

Courses: <https://www.edx.org/course>

OpenBadges: []

Homepage: <https://open.edx.org/>

AndroidApp: <https://play.google.com/store/apps/details?id=org.edx.mobile>

iOSApp: <https://itunes.apple.com/us/app/edx/id945480667>

Source: <https://github.com/edx>

Source: git <https://github.com/edx/edx-platform>

Docs: <http://docs.edx.org/>

Docs: <https://edx.readthedocs.io/projects/edx-guide-for-students/en/latest/>

Docs: <https://edx.readthedocs.io/projects/edx-partner-course-staff/en/latest/>

Docs: <https://github.com/edx/edx-platform/wiki>

Docs: <https://github.com/edx/edx-platform/wiki/Sites-powered-by-Open-edX>

Docs: <https://open.edx.org/open-edx-rest-apis>

Docs: <https://open.edx.org/features-roadmap/all>

Twitter: <https://twitter.com/edXOnline>

edX is an *Open Source* platform for *Online Courses* written mostly in Python and Javascript.

Jupyter and edX

- Jupyter Notebook and *edX* are mostly written in Python
- It's possible to generate an edX course from Jupyter notebooks:
 - <http://mail.scipy.org/pipermail/ipython-dev/2015-February/015911.html>
 - https://github.com/topocm/topocm_content/

Udacity

Wikipedia: <https://en.wikipedia.org/wiki/Udacity>

Homepage: <https://www.udacity.com/>

Courses: <https://www.udacity.com/courses/>

OpenBadges: []

Docs: <https://www.udacity.com/wiki/>

Udacity is a platform for *Online Courses*.

- Udacity offers “Nanodegrees”: <https://www.udacity.com/nanodegree>

Jupyter and Learning

Jupyter Project is great for learning and education.

- Jupyter Notebook, JupyterHub
- Jupyter Notebook supports over 42 languages other than Python.
- Jupyter notebooks can be published as HTML, PDF, ePub, MOBI.
- Jupyter notebooks can be published as reveal.js HTML slide presentations.
- Jupyter notebooks can be published to and served directly from GitHub repos.
- Jupyter notebooks can be published as *edX* courses (*Jupyter and edX*)
- Jupyter notebooks can be structured into per-user, per-class, per-project Docker containers (and folders)
- Jupyter notebooks can be saved to and read from Google Drive:
<https://github.com/jupyter/jupyter-drive>
- Jupyter notebooks are great for taking notes:
<https://github.com/notablemind/notablemind>
- Jupyter notebooks should specify package dependencies (see: *Jupyter and Reproducibility*)
 - Jupyter notebooks can utilize code from ScipyStack packages (e.g. Pip python packages, conda, Anaconda)
- JupyterHub servers host Jupyter Notebook servers for one or more users; with authentication and optional Docker integration.

Learning Topics:

- *Computer Science*
- *Data Science > Data Learning*
- *Software Development*
- Python
- awesome-python-testing

Jupyter and Reproducibility

Jupyter Notebook, *Open Science*, and *Reproducibility*.

Lecture notes (in IPython Notebook format) on
Reproducible Science And Modern Scientific Software
<https://github.com/fperez/reprosw>

“Ten Simple Rules for Reproducible Computational Research”
<http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003285>

Rule 3: Archive the Exact Versions of All External Programs Used

- [] List required packages and extensions
 - watermark: datetime stamp, package versions
<https://github.com/rasbt/watermark>

- version_information: Python interpreter, and Python Package versions
https://github.com/jrjohansson/version_information
- [] List *instaleld* packages and extensions
 - Pip: `pip freeze`
 - Conda: `conda env export`
 - Dpkg: `dpkg-query -l, dpkg --get-selections, wajig list-installed`
- [] List reference and other maybe supported OS
 - OSX, Linux: `uname -a`
 - Windows: `systeminfo`
- [] List reference and other maybe supported platforms
 - CPU: i386, i686, x86-64, ARMv
 - GPU
 - RAM
 - osquery
 - Salt Grains
- [] Generate complete machine image (Backup, Restore, Virtualization)
 - Machine image process:
 - * [] Backup: Take snapshot
 - * [] Post-process: compress, add metadata, test decompression
 - * [] Archive: share/store/backup/upload/verify
 - Machine Imaging Tools:
 - * *clonezilla* (backup and restore partitions from CD/DVD, LAN, HTTP, SSH, PXE)
 - * *bup* (git-like backups for very many and very large files)
 - * *rsync, rsnapshot, rdiff*
 - Virtualization Machine Imaging Tools
 - * Docker Dockerfile and image
 - * Packer config and image
 - * Vagrant Vagrantfile and box

Jupyter and TDD

- The input/output feedback cycle of IPython and Jupyter notebooks captures the essence of *Test Driven Development*
- Jupyter notebooks can be tested with `runipy` and `ipython_nose`
- Jupyter notebooks can be tested and graded with `nbgrader`
- [awesome-python-testing](#) links to a number of testing concepts and Python tools

nbgrader

Source: git <https://github.com/jupyter/nbgrader>

Jupyter notebooks can be submitted and centrally graded with nbgrader.

Note: While it's possible to run tests of all code cells in a Jupyter notebook programmatically with runipy, **it's usually preferable to factor testable code into a module and a package** (e.g. Python Package, Conda package) and then reference those tested functions from within a Jupyter notebook.

JupyterHub Servers

- Sharing resources affords many challenges and opportunities
 - Timeshare resource exhaustion (CPU, RAM, Storage)
 - Security
- Principle of least privilege (“privilege separation”, *Confidentiality*)
https://en.wikipedia.org/wiki/Principle_of_least_privilege
- There are Docker containers for IPython Notebook, Jupyter Notebook, JupyterHub and supporting services.
 - <https://github.com/ipython/ipython/wiki/Install:-Docker>
- JupyterHub servers spawn and proxy to separate instances of Jupyter Notebook; which run on different ports, IPs, and/or containers.
 - <https://github.com/jupyter/jupyterhub/wiki/Spawners>
- JupyterHub servers authenticate users from a number of sources (local, OAuth, GitHub)
 - <https://github.com/jupyter/jupyterhub/wiki/Authenticators>

Knowledge Engineering

See: *Knowledge Engineering*

Linked Curricula Graphs

- <https://westerner.github.io/self-directed-learning/slides.html#knowledge-graph> (2012)
- <https://westerner.github.io/redditlog/#comment/ci3c1o3> (2014)
- [] Add *RDFa* to Course Catalog / Index HTML pages
 - [] [schema.org/Course: https://github.com/schemaorg/schemaorg/issues/195](https://github.com/schemaorg/schemaorg/issues/195)
- [] Link each component of the curriculum to a **Concept URI**
 - *Knowledge Engineering* lists a number of **Wikipedia Concept URIs**
Wikipedia (-> DBpedia RDF <- *LODCloud*)
 - * <https://en.wikipedia.org/wiki/DBpedia> – Wikipedia page for “DBpedia”

- * <https://dbpedia.org/page/DBpedia> – DBpedia page for “DBpedia”
- * <https://www.wikidata.org/wiki/Q465> – Wikidata page for DBpedia (“Q465”)
- A more local/structured vocabulary (with #permalink *URIs*) could also defined mappings from local *Concept URIs* to one or more *Wikipedia Concept URIs*
 - * *Common Core*
 - * *LRMI*
- [] Write tools to discover curriculum resources relevant to one or more concept *URIs*
- [] Write tools to sequence curriculum resources which have *URIs*
 - *Art & Design > Web Production*

OpenBadges

Homepage: <http://openbadges.org/>

Wikipedia: https://en.wikipedia.org/wiki/Mozilla_Open_Badges

Standard: <https://github.com/openbadges/openbadges-specification>

Docs: <https://wiki.mozilla.org/Badges>

Twitter: <https://twitter.com/openbadges>

- [] OpenBadges *JSON* Web Signatures and *Schema.org* (*RDFa*, *JSON-LD*):
<https://github.com/openbadges/openbadges-specification/issues/9>

OpenBadges Backpack

Homepage: <https://backpack.openbadges.org/backpack/>

Source: <https://github.com/mozilla/openbadges-backpack>

See also: *Team Building*, Jupyter

Wikipedia: https://en.wikipedia.org/wiki/Educational_assessment

Wikipedia: https://en.wikipedia.org/wiki/Standardized_test

- https://en.wikipedia.org/wiki/Educational_assessment#Controversy
- https://en.wikipedia.org/wiki/Teaching_to_the_test
- <https://en.wikipedia.org/wiki/Autodidacticism>

Educational Objectives

- https://en.wikipedia.org/wiki/Liberal_education
- https://en.wikipedia.org/wiki/Liberal_arts_education
- <https://en.wiktionary.org/wiki/well-rounded>

Educational Testing Objectives

- Placement
- Quality Control
- Baseline

USA Testing Standards

Wikipedia:

https://en.wikipedia.org/wiki/List_of_standardized_tests_in_the_United_States

- https://en.wikipedia.org/wiki/Education_in_the_United_States
- https://en.wikipedia.org/wiki/History_of_education_in_the_United_States

USA K-12 Curriculum Standards

Common Core

Wikipedia: https://en.wikipedia.org/wiki/Common_Core_State_Standards_Initiative

Homepage: <http://www.corestandards.org/>

Docs: <http://www.corestandards.org/Math/>

Docs: <http://www.corestandards.org/ELA-Literacy/>

High School Diploma

Wikipedia: https://en.wikipedia.org/wiki/High_school_diploma

GED

Wikipedia: https://en.wikipedia.org/wiki/General_Educational_Development

College Credit Highschool Courses

Wikipedia: https://en.wikipedia.org/wiki/Dual_enrollment

- <http://www.nacacnet.org/studentinfo/articles/Pages/EarningCollegeCredit.aspx>
- <https://bigfuture.collegeboard.org/pay-for-college/college-costs/getting-college-credit-before-college>

AP

Wikipedia: https://en.wikipedia.org/wiki/Advanced_Placement

Homepage: <https://apstudent.collegeboard.org/>

Homepage: <http://apcentral.collegeboard.com/>

- https://en.wikipedia.org/wiki/Advanced_Placement#Advanced_Placement_courses
- http://apcentral.collegeboard.com/apc/public/courses/teachers_corner/index.html
- Test grade: 1-5 (5 = “extremely well qualified”)
- e.g. AP Statistics
 - Wikipedia: https://en.wikipedia.org/wiki/AP_Statistics

- <http://apcentral.collegeboard.com/apstatistics>
- <https://apstudent.collegeboard.org/apcourse/ap-statistics>
- e.g. AP Computer Science
 - Wikipedia: https://en.wikipedia.org/wiki/AP_Computer_Science
 - http://apcentral.collegeboard.com/apc/public/courses/teachers_corner/4483.html
 - <https://apstudent.collegeboard.org/apcourse/ap-computer-science-a>

CLEP

Wikipedia: https://en.wikipedia.org/wiki/College_Level_Examination_Program

Homepage: <https://clep.collegeboard.org/>

- https://en.wikipedia.org/wiki/College_Level_Examination_Program#Currently_available_exams

IBDP

Wikipedia: https://en.wikipedia.org/wiki/IB_Diploma_Programme

- https://en.wikipedia.org/wiki/International_Baccalaureate
- *International*

Dual Enrollment

Wikipedia: https://en.wikipedia.org/wiki/Dual_enrollment

- <http://www.act.org/content/act/en/research/using-dual-enrollment.html>
- <http://research.collegeboard.org/publications/comparison-college-outcomes-ap-and-dual-enrollment-students-progress>

USA State Testing Standards

- https://en.wikipedia.org/wiki/List_of_state_achievement_tests_in_the_United_States
- https://en.wikipedia.org/wiki/List_of_standardized_tests_in_the_United_States
- <http://www.corestandards.org/standards-in-your-state/>

USA Higher Education Testing

Undergraduate

SAT

Wikipedia: <https://en.wikipedia.org/wiki/SAT>

Homepage: <https://collegereadiness.collegeboard.org/sat>

- CEEB (*College Entrance Examination Board*)

Wikipedia: https://en.wikipedia.org/wiki/College_Board

Homepage: <https://www.collegeboard.org/>

- Khan Academy SAT resources
 - <https://www.khanacademy.org/college-admissions/making-high-school-count/standardized-tests/v/overview-of-sat>
 - <https://www.khanacademy.org/sat>
 - <https://www.khanacademy.org/test-prep/sat>
 - <https://www.khanacademy.org/test-prep/sat/full-length-SAT-1>
 - <https://www.khanacademy.org/test-prep/sat/sat-math-practice>
 - <https://www.khanacademy.org/test-prep/sat/sat-reading-writing-practice>

ACT

Wikipedia: [https://en.wikipedia.org/wiki/ACT_\(test\)](https://en.wikipedia.org/wiki/ACT_(test))

Homepage: <http://www.act.org/>

- Khan Academy ACT resources
 - <https://www.khanacademy.org/college-admissions/making-high-school-count/standardized-tests/v/overview-of-act>

ASVAB

Wikipedia: https://en.wikipedia.org/wiki/Armed_Services_Vocational_Aptitude_Battery

Homepage: <http://official-asvab.com/>

- https://en.wikipedia.org/wiki/Armed_Services_Vocational_Aptitude_Battery#Composite_scores
- “asvab practice test”

Graduate

- https://en.wikipedia.org/wiki/Comprehensive_examination

GRE

Wikipedia: https://en.wikipedia.org/wiki/Graduate_Record_Examinations

Homepage: <https://www.ets.org/gre>

A GRE (*Graduate Record Examination*) is a standardized test for graduate school entrance.

- ETS (*Educational Testing Service*)

Wikipedia: https://en.wikipedia.org/wiki/Educational_Testing_Service

Homepage: <https://ets.org/>

GMAT

Wikipedia: https://en.wikipedia.org/wiki/Graduate_Management_Admission_Test

Homepage: <http://www.mba.com/us>

A GMAT (*Graduate Management Admission Test*) is a standardized test for graduate school entrance to a business program (e.g. an MBA (*Master of Business Administration*)).

- Khan Academy GMAT resources
 - <https://www.khanacademy.org/test-prep/gmat>
 - <https://www.khanacademy.org/test-prep/gmat/problem-solving>
 - <https://www.khanacademy.org/test-prep/gmat/data-sufficiency>

LSAT

Wikipedia: https://en.wikipedia.org/wiki/Law_School_Admission_Test

Homepage: <http://www.lsac.org/jd/lsat>

An LSAT (*Law School Admissions Test*) is a standardized test for law school entrance to a law school.

MCAT

Wikipedia: https://en.wikipedia.org/wiki/Medical_College_Admission_Test

Homepage: <https://students-residents.aamc.org/applying-medical-school/taking-mcat-exam/>

An MCAT (*Medical College Admission Test*) is a standardized test for medical school entrance.

- Khan Academy MCAT resources:
 - <https://www.khanacademy.org/test-prep/mcat>
 - <https://www.khanacademy.org/test-prep/mcat/critical-analysis-and-reasoning-skills-practice-questions>
 - <https://www.khanacademy.org/test-prep/mcat/biological-sciences-practice>
 - <https://www.khanacademy.org/test-prep/mcat/physical-sciences-practice>
 - <https://www.khanacademy.org/test-prep/mcat/social-sciences-practice>
 - <https://www.khanacademy.org/test-prep/mcat/biomolecules>
 - <https://www.khanacademy.org/test-prep/mcat/cells>

- <https://www.khanacademy.org/test-prep/mcat/organ-systems>
- <https://www.khanacademy.org/test-prep/mcat/physical-processes>
- <https://www.khanacademy.org/test-prep/mcat/chemical-processes>
- <https://www.khanacademy.org/test-prep/mcat/processing-the-environment>
- <https://www.khanacademy.org/test-prep/mcat/behavior>
- <https://www.khanacademy.org/test-prep/mcat/individuals-and-society>
- <https://www.khanacademy.org/test-prep/mcat/society-and-culture>
- <https://www.khanacademy.org/test-prep/mcat/social-inequality>

- https://en.wikipedia.org/wiki/Dental_Admission_Test
- https://en.wikipedia.org/wiki/Optometry_Admission_Test
- https://en.wikipedia.org/wiki/Pharmacy_College_Admission_Test
- <https://westurner.github.io/opengov/us/#gnuhealth>

Postgraduate

Wikipedia: https://en.wikipedia.org/wiki/Postgraduate_education

Dissertation / Thesis

Wikipedia: <https://en.wikipedia.org/wiki/Thesis>

- https://en.wikipedia.org/wiki/Thesis_statement
- https://en.wikipedia.org/wiki/Thesis_by_publication
- https://en.wikipedia.org/wiki/Category:Academic_publishing

See also:

- *Education Technology > Publishing*

Master's Degree

Wikipedia: https://en.wikipedia.org/wiki/Master%27s_degree

- https://en.wikipedia.org/wiki/Category:Master%27s_degrees
- https://en.wikipedia.org/wiki/List_of_master%27s_degrees_in_North_America
- https://en.wikipedia.org/wiki/Master%27s_degree_in_Europe

Doctorate

Wikipedia: <https://en.wikipedia.org/wiki/Doctorate>

A Doctorate Degree is for to teach and/or to work as a professional.

- https://en.wikipedia.org/wiki/Category:Doctoral_degrees
- https://en.wikipedia.org/wiki/List_of_fields_of_doctoral_studies_in_the_United_States

Honorary Degree

Wikipedia: https://en.wikipedia.org/wiki/Honorary_degree

CHAPTER 20

Art & Design

<https://en.wikipedia.org/wiki/Art>
<https://en.wikipedia.org/wiki/Design>

Media

<https://en.wikipedia.org/wiki/Media>

- [https://en.wikipedia.org/wiki/Media_\(communication\)](https://en.wikipedia.org/wiki/Media_(communication))
- https://en.wikipedia.org/wiki/List_of_artistic_media

Visual Composition

[https://en.wikipedia.org/wiki/Composition_\(visual_arts\)](https://en.wikipedia.org/wiki/Composition_(visual_arts))

- https://en.wikipedia.org/wiki/Rule_of_thirds
- [https://en.wikipedia.org/wiki/Headroom_\(photographic_framing\)](https://en.wikipedia.org/wiki/Headroom_(photographic_framing))
- https://en.wikipedia.org/wiki/Lead_room

Graphic Design

https://en.wikipedia.org/wiki/Graphic_design

Web Design

See: Web Design

BMP

https://en.wikipedia.org/wiki/BMP_file_format

File Extension: .bmp

GIMP

Wikipedia: <https://en.wikipedia.org/wiki/GIMP>

Homepage: <http://www.gimp.org/>

Download: <http://www.gimp.org/downloads/>

Source: <http://www.gimp.org/source/>

Source: git [git://git.gnome.org/gimp](https://git.gnome.org/gimp)

Docs: <http://www.gimp.org/docs/>

Docs: <http://docs.gimp.org/2.8/en/>

Docs: http://wiki.gimp.org/wiki/Main_Page

Docs: http://wiki.gimp.org/wiki/Hacking:Developer_FAQ

GIMP (GNU Image Manipulation Program) is an extensible *Open Source* raster image editor (*BMP*, *PNG*, *JPEG*, *GIF*, *ICO*, ..., GIMP .xcf).

- GIMP plugins are shared at <http://registry.gimp.org/popular>

GIF

Wikipedia: <https://en.wikipedia.org/wiki/GIF>

GIF is a file format for sequences of images (“animations”).

GIFs appear to load quickly because there is no media player or codec initialization time, but are actually less efficient than compressed video (e.g. *H.264*) because GIFs are not compressed and contain every frame of the animation.

Video codecs compress the changes between frames, in order to avoid basically GIF.

- https://en.wikipedia.org/wiki/Hampster_Dance (1990s)
‘ * <https://en.wikipedia.org/wiki/Category:Under-construction_templates> ‘ __
- https://en.wikipedia.org/wiki/Dancing_baby (1990s)
- https://en.wikipedia.org/wiki/Real_Ultimate_Power (2002; should’ve had GIFs)
- https://en.wikipedia.org/wiki/Dramatic_Chipmunk (2007; YouTube *tube site*)
- 2015: Google+, Twitter, Facebook add support for GIFs

ICO

Wikipedia: [https://en.wikipedia.org/wiki/ICO_\(file_format\)](https://en.wikipedia.org/wiki/ICO_(file_format))

ICO is a computer graphics file format for icons.

- ICO files store multiple resolutions in one file.
- *GIMP* can work with ICO files

Inkscape

Wikipedia: <https://en.wikipedia.org/wiki/Inkscape>

Homepage: <https://inkscape.org/en/>

Source: <https://launchpad.net/~inkscape.dev>

Source: `bzr lp:inkscape`

Download: <https://inkscape.org/en/download/>

Docs: <https://inkscape.org/en/learn/>

Docs: <https://inkscape.org/en/learn/books/>

Docs: <https://inkscape.org/en/doc/keys091.html>

Docs: <https://inkscape.org/en/doc/inkscape-man.html>

Inkscape is an *Open Source* vector graphics (*SVG XML*) editor.

SVG graphics can scale to very large and very small resolutions (e.g. 1080P, 4K, 8K, HD, 1920px, poster-size) and remain smooth.

- https://en.wikipedia.org/wiki/Scalable_Vector_Graphics
- http://wiki.inkscape.org/wiki/index.php/Extension_repository

JPEG

Wikipedia: <https://en.wikipedia.org/wiki/JPEG>

File Extension: `.jpg`, `.jpeg`

PNG

Wikipedia: <https://en.wikipedia.org/wiki/PNG>

File Extension: `.png`, `.png`

SVG

Wikipedia: <https://en.wikipedia.org/wiki/SVG>

File Extension: `.svg`

3D Modeling

https://en.wikipedia.org/wiki/3D_modeling
https://en.wikipedia.org/wiki/3D_rendering
https://en.wikipedia.org/wiki/3D_computer_graphics

- https://en.wikipedia.org/wiki/3D_computer_graphics_software

Blender

Wikipedia: [https://en.wikipedia.org/wiki/Blender_\(software\)](https://en.wikipedia.org/wiki/Blender_(software))
Homepage: <https://blender.org/>
Source: <https://developer.blender.org/diffusion/>
Source: git [git://git.blender.org/blender.git](https://git.blender.org/blender.git)
Download: <https://www.blender.org/download/>
Docs: <https://www.blender.org/manual/>
Docs: <https://www.blender.org/api/250PythonDoc/>

Blender is an *Open Source* 3D computer graphics software for movies, apps, and games written in C, C++, and Python.

- Blender has a scene graph over time t
- Blender has multi-camera tracking
- Blender has a Python API

NURBS

https://en.wikipedia.org/wiki/Non-uniform_rational_B-spline

NURBS (“Non-rational Uniform B-Splines”) are scale-invariant n-dimensional models.

- NURBS curves are sort of like *SVG* for *3D Modeling*: they scale losslessly from the very small to the very large.

OpenGL

Wikipedia: <https://en.wikipedia.org/wiki/OpenGL>

OpenGL is an API for 2D and 3D graphics on any platform which is implemented by all modern GPUs.

Rhinoceros 3D

Wikipedia: https://en.wikipedia.org/wiki/Rhinoceros_3D
Homepage: <http://www.rhino3d.com/>

Rhinoceros 3D is a commercial *NURBS 3D Modeling* tool for Windows and OSX.

SketchUp

Wikipedia: <https://en.wikipedia.org/wiki/SketchUp>

Homepage: <http://www.sketchup.com/>

Download: <http://www.sketchup.com/download>

Docs: <http://www.sketchup.com/learn>

- SketchUp extensions can be shared at <http://extensions.sketchup.com/>

WebGL

Wikipedia: <https://en.wikipedia.org/wiki/WebGL>

Homepage: <https://www.khronos.org/webgl/>

Standard: <https://www.khronos.org/registry/webgl/specs/1.0/>

Docs: <https://www.khronos.org/developers/reference-cards/>

Docs: <https://www.khronos.org/webgl/wiki/>

Docs: <http://caniuse.com/#search=webgl>

WebGL allows Web Browsers to run *OpenGL* programs with *HTML5* and Javascript.

- Lists of WebGL implementations:

https://www.khronos.org/webgl/wiki/User_Contributions

<https://en.wikipedia.org/wiki/Category:WebGL>

- WebGL Chrome Experiments:

<https://www.chromeexperiments.com/webgl>

See: *Data Visualization Tools*

Mediated Reality

Wikipedia: https://en.wikipedia.org/wiki/Computer-mediated_reality

- https://en.wikipedia.org/wiki/Computer-mediated_reality#/media/File:Viraugmixmodmediated_reality.png

Mixed Reality

Wikipedia: https://en.wikipedia.org/wiki/Mixed_reality

Augmented Reality

Wikipedia: https://en.wikipedia.org/wiki/Augmented_reality

Virtual Reality

https://en.wikipedia.org/wiki/Virtual_reality

Mediated reality mediums

Google Cardboard

Wikipedia: https://en.wikipedia.org/wiki/Google_Cardboard

Homepage: <https://www.google.com/get/cardboard/>

- Android Smartphone + Google Cardboard viewer
 - <https://www.google.com/get/cardboard/get-cardboard/>

Google Glass

Wikipedia: https://en.wikipedia.org/wiki/Google_Glass

Homepage: <https://www.google.com/glass/>

- Glasses frame + projective

HTC Vive

Wikipedia: https://en.wikipedia.org/wiki/HTC_Vive

Homepage: <https://www.htcvive.com/us/>

- Valve *Steam* VR

Oculus VR

Wikipedia: https://en.wikipedia.org/wiki/Oculus_VR

Homepage: <https://www.oculus.com/>

Oculus Rift

Wikipedia: https://en.wikipedia.org/wiki/Oculus_Rift

Homepage: <https://www.oculus.com/en-us/rift/>

Samsung Gear VR

Wikipedia: https://en.wikipedia.org/wiki/Samsung_Gear_VR

Homepage: <https://www.oculus.com/en-us/gear-vr/>

- Samsung Galaxy Android Smartphone + Samsung Ocular Gear VR Headset

Windows Holographic

Wikipedia: https://en.wikipedia.org/wiki/Windows_Holographic

Docs: <https://dev.windows.com/en-us/holographic>

Docs: <https://dev.windows.com/en-us/holographic/documentation>

- Windows 10

Microsoft HoloLens

Wikipedia: https://en.wikipedia.org/wiki/Microsoft_HoloLens

Homepage: <https://www.microsoft.com/microsoft-hololens/en-us>

Docs: <https://www.microsoft.com/microsoft-hololens/en-us/developers>

Photography

<https://en.wikipedia.org/wiki/Photography>

https://en.wikipedia.org/wiki/Outline_of_photography

https://en.wikipedia.org/wiki/Digital_photography

https://en.wikipedia.org/wiki/Photography_and_the_law

https://en.wikipedia.org/wiki/Digital_darkroom

- https://en.wikipedia.org/wiki/Category:Photo_software
- https://en.wikipedia.org/wiki/Category:Free_photo_software
- https://en.wikipedia.org/wiki/Image_editing

See: *Visual Composition*

Videography

<https://en.wikipedia.org/wiki/Videography>

<https://en.wikipedia.org/wiki/Cinematography>

See: *Visual Composition*

Video Production

https://en.wikipedia.org/wiki/Video_production

https://en.wikipedia.org/wiki/Video_editing

https://en.wikipedia.org/wiki/Non-linear_editing_system

<https://en.wikipedia.org/wiki/Post-production>

- https://en.wikipedia.org/wiki/Video_editing_software
- https://en.wikipedia.org/wiki/Comparison_of_video_editing_software
- https://en.wikipedia.org/wiki/Category:Video_editing_software
- https://en.wikipedia.org/wiki/Category:Free_video_software

See: *Visual Composition*, *Screen Captures*

Cinelerra

Wikipedia: <https://en.wikipedia.org/wiki/Cinelerra>

Homepage: <http://cinelerra.org/>

Source: `git git://git.cinelerra-cv.org/CinelerraCV.git`

Docs: <http://cinelerra-cv.org/docs.php>

Docs: http://cinelerra-cv.org/docs/cinelerra_cv_manual_en.html

CinePaint

Wikipedia: <https://en.wikipedia.org/wiki/CinePaint>

Homepage: <http://www.cinepaint.org/>

Homepage: <http://sourceforge.net/projects/cinepaint/>

Source: <http://sourceforge.net/p/cinepaint/code/>

CinePaint is an *Open Source* tool for hand-painting video frames.

FFmpeg

Wikipedia: <https://en.wikipedia.org/wiki/FFmpeg>

Homepage: <https://www.ffmpeg.org/>

Download: <https://www.ffmpeg.org/download.html>

Source: `git git://source.ffmpeg.org/ffmpeg.git`

Docs: <https://www.ffmpeg.org/documentation.html>

Docs: http://www.itbroadcastanddigitalcinema.com/ffmpeg_howto.html

FFmpeg is an audio/video transcoding library with support for very many codecs and a great command line interface (`ffmpeg`).

- FFmpeg can be used to rescale audio/video to a different resolution

- FFmpeg can be used to (re-)encode audio/video

FLV

Wikipedia: https://en.wikipedia.org/wiki/Flash_Video

FLV (“Flash Video”) is a media file container format for sharing media; especially over *HTTP*.

- FLV can be downloaded in segments (is “seekable”)
- FLV contains *MPEG-4* encoded content
- FLV supports *H.264 MPEG-4* video
- Most *Tube Sites* (natively) support FLV

HandBrake

Wikipedia: <https://en.wikipedia.org/wiki/HandBrake>

Homepage: <https://handbrake.fr/>

Download: <https://handbrake.fr/downloads.php>

HandBrake is an offline batch GUI audio/video transcoding tool.

- HandBrake can use *FFmpeg* for transcoding
- HandBrake helps with rescaling a video for e.g. mobile devices with limited bandwidth ahead-of-time

H.264

Wikipedia: https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC

H.264 (or *MPEG-4 AVC*) is a widely implemented video compression codec.

- Blu-ray discs are encoded with H.264
- HDTV, digital cable, and digital satellite are encoded with H.264
- Hulu, iTunes, Vimeo, VEVO, and YouTube all encode video with H.264

Note: Some mobile devices include hardware support for H.264 decoding, which can save CPU time and battery, resulting in smoother playback.

Lightworks

Wikipedia: <https://en.wikipedia.org/wiki/Lightworks>

Homepage: <http://www.lwks.com/>

Download: http://www.lwks.com/index.php?option=com_lwks&view=download&Itemid=206

Source:

Docs: http://www.lwks.com/index.php?option=com_lwks&view=download&Itemid=206&tab=4

Docs: http://www.lwks.com/index.php?option=com_content&view=article&id=162&Itemid=246

Lightworks is a professional video editing software tool.

- Lightworks is included in *DreamStudio*

LiVES

Wikipedia: <https://en.wikipedia.org/wiki/LiVES>

Homepage: <http://lives-video.com/>

Download: <http://lives-video.com/index.php?do=downloads>

Download: <http://lives-video.com/index.php?do=addons>

Project: <http://sourceforge.net/projects/lives/>

Source: svn <http://svn.code.sf.net/p/lives/code/trunk>

Docs: <http://lives-video.com/index.php?do=documentation>

Docs: http://lives-video.com/manual/LiVES_manual.html

Docs: <http://lives-video.com/index.php?do=tutorial-wiki>

Docs: <http://lives-video.com/doxygen/LiVES/files.html>

Video: <http://sourceforge.net/projects/lives/>

LiVES is an *Open Source* video editing and realtime *VJing* software tool written in C, Perl, and Python.

- LiVES has full undo and redo (“nondestructive editing”)
- LiVES has strong native support for networked streaming
- LiVES has support for *LADSPA*, *JACK*, *MIDI*

MPEG

Wikipedia: https://en.wikipedia.org/wiki/Moving_Picture_Experts_Group

Homepage: <http://mpeg.chiariglione.org/>

MPEG is a standards body formed by *ISO* and *IEC*.

- <https://en.wikipedia.org/wiki/MPEG-1> (MP3)
- <https://en.wikipedia.org/wiki/MPEG-2> (MP3, DVD, Digital Cable/Satellite)
- <https://en.wikipedia.org/wiki/MPEG-3>
- <https://en.wikipedia.org/wiki/MPEG-4>
 - https://en.wikipedia.org/wiki/MPEG-4_Part_14 (MP4, .mp4, .m4a)
 - https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC (*H.264*)
- *MPEG-DASH* (Adaptive bitrate streaming)

Screenwriting

<https://en.wikipedia.org/wiki/Screenwriting>

- [https://en.wikipedia.org/wiki/Rule_of_three_\(writing\)](https://en.wikipedia.org/wiki/Rule_of_three_(writing))
- [] fountainio, vim fountainio

See: *Small Business > Concept*

Fountain.io

Homepage: <http://fountain.io/>

Source: git <https://github.com/nyousefi/Fountain>

Docs: <http://fountain.io/faq>

Docs: <http://fountain.io/syntax>

Docs: <http://fountain.io/howto>

Docs: <http://fountain.io/apps>

Fountain.io is a Lightweight Markup Language similar to Markdown with extensions for writing formatted screenplays with e.g. Scene Headings, Characters, Action (scene description; cues), Dialogue.

- <http://fountain.io/apps> lists application plugins and integrations

Songwriting

<https://en.wikipedia.org/wiki/Songwriter>

<https://en.wikipedia.org/wiki/Lyrics>

Poetry

<https://en.wikipedia.org/wiki/Poetry>

- <https://en.wikipedia.org/wiki/Rhythm>
- [https://en.wikipedia.org/wiki/Metre_\(poetry\)](https://en.wikipedia.org/wiki/Metre_(poetry))
- https://en.wikipedia.org/wiki/Consonance_and_dissonance

Scorewriting

<https://en.wikipedia.org/wiki/Scorewriter>

https://en.wikipedia.org/wiki/Sheet_music

https://en.wikipedia.org/wiki/Comparison_of_scorewriters

LilyPond

Wikipedia: <https://en.wikipedia.org/wiki/LilyPond>

Homepage: <http://www.lilypond.org/>

Download: <http://www.lilypond.org/download.html>

Source: <http://download.linuxaudio.org/lilypond/source/?C=N;O=D>

Source: git <http://git.savannah.gnu.org/r/lilypond.git>

Docs: <http://www.lilypond.org/manuals.html>

Docs: <http://www.lilypond.org/doc/v2.19/Documentation/contributor-big-page.html>

LilyPond is an *Open Source* software tool for writing and engraving musical score compositions with support for *MIDI*, *MusicXML* written in C, Scheme, and Python

MuseScore

Wikipedia: <https://en.wikipedia.org/wiki/MuseScore>

Homepage: <https://musescore.org/>

Download: <https://musescore.org/en/download>

Source: git <https://github.com/musescore/MuseScore>

Docs: <https://musescore.org/en/handbook>

Docs: <https://musescore.org/en/development>

MuseScore is an *Open Source* software tool for writing and engraving musical score compositions with support for *MIDI*, *MusicXML* written in C++ and Qt.

- MuseScore can read/write to *MusicXML*, *MIDI*, GuitarPro
- MuseScore can write to PDF, SVG, PNG, PostScript
- MuseScore can save audio to WAV, FLAC, MP3, OGG
- There are MuseScore apps for iOS and Android
- MuseScore.com hosts sheet music: <https://musescore.com/sheetmusic>
 - <https://musescore.com/sheetmusic?instruments=7> (#guitar)

Musical Notation

https://en.wikipedia.org/wiki/Musical_notation

https://en.wikipedia.org/wiki/List_of_musical_symbols

https://simple.wikipedia.org/wiki/Sheet_music

https://en.wikipedia.org/wiki/Sheet_music

https://en.wikipedia.org/wiki/Lead_sheet

https://en.wikipedia.org/wiki/Chord_chart

MusicXML

Wikipedia: <https://en.wikipedia.org/wiki/MusicXML>

File Extension: .xml, .mxl

Homepage: <http://www.musicxml.com/>

Standard: <http://www.musicxml.com/for-developers/>

Standard: <http://www.musicxml.com/for-developers/musicxml-dtd/>

Standard: <http://www.musicxml.com/for-developers/musicxml-xsd/>

MusicXML is an *XML* standard for *Musical Notation*.

- *MIDI* can be represented in *MusicXML*.

Tablature

Wikipedia: <https://en.wikipedia.org/wiki/Tablature>

Tablature is *Musical Notation* mapped onto instrument positions (e.g. guitar frets) and spaced.

There are many tab formats; ASCII chord diagrams being the most classic.

Rosegarden

Wikipedia: <https://en.wikipedia.org/wiki/Rosegarden>

Homepage: <http://www.rosegardenmusic.com/>

Rosegarden is an *Open Source* digital audio workstation for score composition.

- *MIDI* and Hydrogen file import
- *MIDI*, *Csound*, *LilyPond* and *MusicXML* file export (including PostScript and PDF output file generation of score)

Tabledit

Wikipedia: https://en.wikipedia.org/wiki/Tabledit_Tablatore_Editor

Homepage: <http://www.tabledit.com/>

Download: <http://www.tabledit.com/download/>

Docs: <http://el-kay.com/tabledit/>

Docs: <http://www.tabledit.com/faq/>

Videos: <https://www.youtube.com/user/TablatoreEditor>

Tabledit is a musical *Tablature* (“tab”) *Musical Composition* tool with support for *MusicXML*, *LilyPond*, *MIDI*.

Musical Composition

https://en.wikipedia.org/wiki/Musical_composition

<https://en.wikipedia.org/wiki/Orchestration>

<https://en.wikipedia.org/wiki/Composer>

https://en.wikipedia.org/wiki/Music_theory

[https://en.wikipedia.org/wiki/Hook_\(music\)](https://en.wikipedia.org/wiki/Hook_(music))

Audio Synthesis

https://en.wikipedia.org/wiki/Audio_synthesis

<https://en.wikipedia.org/wiki/Synthesizer>

<https://en.wikipedia.org/wiki/Synth>

https://en.wikipedia.org/wiki/Comparison_of_audio_synthesis_environments

- https://en.wikipedia.org/wiki/Electronic_musical_instrument
- https://en.wikipedia.org/wiki/Musical_keyboard
- https://en.wikipedia.org/wiki/Electric_piano
- https://en.wikipedia.org/wiki/Digital_piano
- [https://en.wikipedia.org/wiki/Effects_unit \(“effects pedal”\)](https://en.wikipedia.org/wiki/Effects_unit_(%22effects_pedal%22))

Chuck

Wikipedia: <https://en.wikipedia.org/wiki/Chuck>

Homepage: <http://chuck.cs.princeton.edu/>

Homepage: <http://chuck.stanford.edu/>

Download: <http://chuck.cs.princeton.edu/release/>

Source: <http://chuck.cs.princeton.edu/release/files/>

Standard: <http://chuck.cs.princeton.edu/doc/language/>

Docs: <http://chuck.cs.princeton.edu/doc/>

Docs: <http://chuck.cs.princeton.edu/doc/program/>

Docs: <http://chuck.cs.princeton.edu/doc/program/otfp.html>

Chuck is an *Open Source* realtime, concurrent *Musical Composition Audio Synthesis* programming language.

- Chuck supports “On the Fly Programming”: code and loops can be changed while the Chuck program is running
- Chuck supports *JACK*
- Chuck works with *STK* Synthesis Toolkit

Csound

Wikipedia: <https://en.wikipedia.org/wiki/Csound>

Homepage: <http://csound.github.io/>

Project: <http://sourceforge.net/projects/csound/>

Source: git <https://github.com/csound/csound>

Docs: <http://csound.github.io/documentation.html>

Docs: <http://csound.github.io/docs/manual/index.html>

Docs: <http://booki.flossmanuals.net/csound/introduction/>

Csound is an *Open Source Musical Composition Audio Synthesis* programming library API.

- Csound works on Linux, OSX, Windows, Browsers (emscripten (Javascript)), Chrome Native Client (PNaCl), iOS, and Android
- Csound supports *MIDI*
- Csound ships with the OLPC XO laptops

Csound is simultaneously both ‘old school’ and ‘new school’.

—<http://booki.flossmanuals.net/csound/introduction/>

Hookpad

Homepage: <http://www.hooktheory.com/hookpad/>

Docs: <http://www.hooktheory.com/hookpad/docs>

Docs: <http://www.hooktheory.com/hookpad/shortcuts>

Hookpad is a web-based tool for *Musical Composition*

- <http://www.hooktheory.com/hookpad/new>
- <http://www.hooktheory.com/education>
- *Hooktheory I: Music Theory* (Book)

<http://www.hooktheory.com/music-theory-for-songwriting>

Hydrogen

Wikipedia: [https://en.wikipedia.org/wiki/Hydrogen_\(software\)](https://en.wikipedia.org/wiki/Hydrogen_(software))

Homepage: <http://hydrogen-music.org/hcms/>

Source: git <https://github.com/hydrogen-music/hydrogen>

Docs: <http://hydrogen-music.org/hcms/node/393>

Docs: <http://hydrogen-music.org/hcms/node/5>

Hydrogen is an *Open Source* drum machine software with support for *JACK*, ALSA, *MIDI*, and drumkits.

- Hydrogen is included in *DreamStudio*.

MIDI

Wikipedia: <https://en.wikipedia.org/wiki/MIDI>
https://en.wikipedia.org/wiki/MIDI_controller
https://en.wikipedia.org/wiki/DJ_digital_controller

STK

Wikipedia: https://en.wikipedia.org/wiki/Synthesis_Toolkit
Homepage: <https://ccrma.stanford.edu/software/stk/>
Download: <https://ccrma.stanford.edu/software/stk/download.html>
Source: git <https://github.com/thestk/stk/>
Docs: <https://ccrma.stanford.edu/software/stk/information.html>
Docs: <https://ccrma.stanford.edu/software/stk/classes.html>
Docs: <https://ccrma.stanford.edu/software/stk/tutorial.html>

STK is a realtime audio synthesis C++ library API for simulating physical and synthesized sounds.

- *ChucK* works with *STK*

VST

Wikipedia: https://en.wikipedia.org/wiki/Virtual_Studio_Technology
Homepage:

VST is a reusable audio transform API for *Audio Synthesis* and *Sound Production* tools.

- A VST plugin is a packaged module of instruments, effects, and/or *MIDI* code
- A VST host is a program which can run VST plugins
 - *Ardour*
 - *Audacity*
 - ACID, Adobe Premiere, Cakewalk, FL Studio, Sound Forge, Vegas

Sound Production

<https://en.wikipedia.org/wiki/Sound>
https://en.wikipedia.org/wiki/Sound_recording_and_reproduction
https://en.wikipedia.org/wiki/Audio_engineer

- https://en.wikipedia.org/wiki/Mixing_console

https://en.wikipedia.org/wiki/Comparison_of_free_software_for_audio
https://en.wikipedia.org/wiki/List_of_Linux_audio_software

https://en.wikipedia.org/wiki/Comparison_of_digital_audio_editors

https://en.wikipedia.org/wiki/Recording_studio

Acoustical Engineering

https://en.wikipedia.org/wiki/Acoustical_engineering

DJ

https://en.wikipedia.org/wiki/Disc_jockey

AAC

Wikipedia: https://en.wikipedia.org/wiki/Advanced_Audio_Coding

AAC is an audio compression standard designed to replace *MP3*.

AAC is a component of *MPEG-4* audio (“MP4”).

- AAC is a *lossy* audio compression algorithm
- iTunes files are in AAC format (MPEG-4 Audio, MP4, M4a)

Ardour

Wikipedia: [https://en.wikipedia.org/wiki/Ardour_\(software\)](https://en.wikipedia.org/wiki/Ardour_(software))

Homepage: <https://ardour.org/>

Ardour is an *Open Source* Digital Audio Workstation (DAW) for multi-track recording, editing, and mixing.

- Ardour supports *JACK*, *MIDI* sequencing (drum machines), AudioUnit, *LADSPA*, LV2, Mackie control protocol, OSC
- Ardour can sync one video in the timeline.
- Ardour integrates with external control surfaces (soundboard, mixing console, audio production surface).
- Ardour is included in *Ubuntu Studio*.

Audacity

Wikipedia: [https://en.wikipedia.org/wiki/Audacity_\(audio_editor\)](https://en.wikipedia.org/wiki/Audacity_(audio_editor))

Homepage: <http://audacityteam.org/>

Download: <http://audacityteam.org/download/>

Download: <http://audacityteam.org/download/plugins>

Source: git <https://github.com/audacity/audacity>

Docs: <http://audacityteam.org/help/documentation>

Docs: http://wiki.audacityteam.org/wiki/Audacity_Wiki_Home_Page

Docs: <http://wiki.audacityteam.org/wiki/Plugins>

Docs: http://wiki.audacityteam.org/wiki/Creating_your_own_Plug-in

Docs: <http://wiki.audacityteam.org/wiki/Category:Tutorial>

Audacity is an *Open Source* sound recording and editing software tool.

- Audacity supports LADSPA, and LV2
- Audacity is included in *DreamStudio*, *Ubuntu Studio*

FLAC

Wikipedia: <https://en.wikipedia.org/wiki/FLAC>

Homepage: <https://xiph.org/flac/>

Download: <https://xiph.org/flac/download.html>

Source: git <https://git.xiph.org/flac.git>

Docs: <https://xiph.org/flac/documentation.html>

FLAC is an *Open Source* lossless audio codec.

- *FLAC* files are, on average, like 10x (TODO ?) the file size of e.g. *MP3*, because they encode the full spectrum.

JACK

Wikipedia: https://en.wikipedia.org/wiki/JACK_Audio_Connection_Kit

Homepage: <http://jackaudio.org/>

Download: <http://jackaudio.org/downloads/>

Docs: <https://github.com/jackaudio/jackaudio.github.com/wiki>

Docs: <http://jackaudio.org/api/>

JACK is a low-latency (“realtime”) sound server for POSIX operating systems (Linux, OSX, Windows) written in C and C++.

- List of JACK-supporting applications: <http://jackaudio.org/applications/>
 - *Ardour*
 - *Audacity*
 - *LiVES*
 - *Mixxx*
 - *Blender*
 - *VLC*

LADSPA

Wikipedia: <https://en.wikipedia.org/wiki/LADSPA>

Homepage: <http://www.ladspa.org/>

LADSPA (“Linux Audio Developer’s Simple Plugin API”) is an *Open Source* standard for audio filters and effects written in C.

- *Ardour*, *Audacity*, *LiVES*, and *Rosegarden* all support LADSPA

Mixxx

Wikipedia: <https://en.wikipedia.org/wiki/Mixxx>

Homepage: <http://mixxx.org/>

Source: git <https://github.com/mixxxdj/mixxx>

Download: <http://mixxx.org/download/>

Download: <http://mixxx.org/download/#stable>

Docs: <http://mixxx.org/manual/latest/>

Docs: <http://mixxx.org/wiki/doku.php/>

Docs: http://mixxx.org/wiki/doku.php/hardware_compatibility#sound_cards

Docs: http://mixxx.org/wiki/doku.php/hardware_compatibility#controller_mappings

Mixxx is an *Open Source* sound recording, mixing, editing, beatmatching, and production software tool with *MIDI* and *HID DJ* controller support (for use with keyboards, analog turntables, CD players) written in C++, Qt, *XML*, and Javascript.

- Mixxx has dual and quad decks with scratchable waveforms and beatmatching indicators (“cue points”), time stretching, beat looping, EQ, crossfading
- Mixxx supports timecoded vinyl and CDs

MP3

Wikipedia: <https://en.wikipedia.org/wiki/MP3>

MP3 refers to one of two *MPEG* audio codecs.

- MP3 can be around a tenth the size of a raw *WAV* file
- MP3 can be CBR (“Constant Bitrate”) or VBR (“Variable Bitrate”) depending on the encoding parameters
- MP3 is a *lossy* audio compression algorithm
- Amazon Music (Amazon MP3) files are in MP3 format

See also: *AAC*, *FLAC*, *Ogg*, *SHN*

PulseAudio

Wikipedia: <https://en.wikipedia.org/wiki/PulseAudio>

Homepage: <http://www.freedesktop.org/wiki/Software/PulseAudio/>

Source: `git git://anongit.freedesktop.org/pulseaudio/pulseaudio`

Source: `http://cgkit.freedesktop.org/pulseaudio/pulseaudio/`

Download: `http://www.freedesktop.org/wiki/Software/PulseAudio/Download/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/Modules/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/Network/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/SystemWide/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/Developer/`

Docs: `http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/Developer/ModuleAPI/`

Docs: `http://freedesktop.org/software/pulseaudio/doxygen/`

Docs: `https://wiki.ubuntu.com/PulseAudio`

Docs: `https://wiki.archlinux.org/index.php/PulseAudio`

Docs: `https://wiki.debian.org/PulseAudio`

Docs: `https://wiki.gentoo.org/wiki/PulseAudio`

Docs: `http://mpd.wikia.com/wiki/PulseAudio`

PulseAudio is an *Open Source* realtime sound server.

- How can I use PulseAudio to stream music from my main PC to my LAN with multiple PCs with speakers?
- How can I use PulseAudio to share a single LINE-IN/MIC jack on the entire LAN?
- How can I use PulseAudio as an RTP based N:N multicast conferencing solution for the LAN?
- PulseAudio runs on FreeBSD, NetBSD, OpenBSD, Linux, Solaris, OSX, and Windows
- There are many PulseAudio modules for the PulseAudio ModuleAPI <http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/Modules/>
 - `module-native-protocol-unix`, `module-native-protocol-tcp` – PulseAudio native protocol
 - `module-rtp-send`, `module-rtp-recv` – RTP
 - `module-raop-discover`, `module-raop-sink` – AirPlay
 - `module-jack-sink`, `module-jack-source`, `module-jackdbus-detect` – *JACK*
 - `module-loopback`, `module-echo-cancel`, `module-equalizer-sink`, `module-ladspa-sink` *LADSPA*, `module-switch-on-connect`, `module-rescue-streams`, `module-device-restore`, `module-stream-restore`, `module-card-restore` – sink, source, filters
 - `module-pipe-sink`, `module-pipe-source`, `module-always-sink`, `module-null-sink`
 - `module-zeroconf-publish`, `module-zeroconf-discover` – ZeroConf / Bonjour peer discovery (local multi-cast LAN/WLAN segment)
 - `module-bluetooth-discover`, `module-bluetooth-policy`, `module-bluetooth-proximity` – Bluetooth A2DP audio peer discovery
 - `module-lirc` – Infrared remote volume control
 - `module-cli` – `pactl` commandline interface
- There are a number of volume control and mixer apps for PulseAudio.
- `pactl` – PulseAudio control CLI utility
- `pacmd` – PulseAudio runtime reconfiguration utility

- `pacat - cat` for PulseAudio servers (play (un)encoded audio)
- `pavucontrol` – PulseAudio taskbar widget (Gnome)
- `paprefs` – PulseAudio preferences config GUI (Gnome)
- `pax11publish` – PulseAudio / X11 GUI synchronization

```
man pulse-cli-syntax
man pactl
pactl info
pactl list
pactl load-module module-native-protocol-tcp
pacmd load-module module-native-protocol-tcp
```

A Bash script to supports pushing all system audio from one machine to another (e.g. speakers connected to a PC or a Pi on the other side of the room):

- <https://github.com/westurner/dotfiles/blob/master/scripts/pulse.sh>

WAV

Wikipedia: <https://en.wikipedia.org/wiki/WAV>

WAV (“Waveform Audio Format”) is a file format for storing an uncompressed audio bitstream.

- Like WAV, Audio CDs are encoded with PCM; but unlike WAV, Audio CDs are encoded with PCM and *Red Book audio*

Audio/Visual Production

https://en.wikipedia.org/wiki/Audiovisual_art

[https://en.wikipedia.org/wiki/Mashup_\(video\)](https://en.wikipedia.org/wiki/Mashup_(video))

https://en.wikipedia.org/wiki/Remix_culture

- https://en.wikipedia.org/wiki/Compression_artifact#Artistic_use

VJ

<https://en.wikipedia.org/wiki/VJing>

<https://en.wikipedia.org/wiki/DVJ>

https://en.wikipedia.org/wiki/Video_synthesizer

https://en.wikipedia.org/wiki/Music_visualization

https://en.wikipedia.org/wiki/VJing#Common_technical_setups

Advanced Visualization Studio

Wikipedia: https://en.wikipedia.org/wiki/Advanced_Visualization_Studio

Homepage: <http://www.1014.org/code/nullsoft/avs/>

Source: http://www.1014.org/code/nullsoft/avs/avs_src.zip

Advanced Visualization Studio is a GUI tool for developing music visualization presets for the Winamp media player.

Geiss

Homepage: <http://www.geisswerks.com/geiss/>

Docs: <http://www.geisswerks.com/geiss/shots.html>

Geiss is a Winamp plugin for music visualization.

Windows 95/98/ME/2000/XP/Vista, DirectX 3.0 or later, and a 200 MHz or faster processor. You can also run Geiss on Linux, using Wine.

To use the plugin, you must have Winamp... if you don't already use Winamp, just download the screen-saver version of Geiss.

libvisual

Wikipedia: <https://en.wikipedia.org/wiki/Libvisual>

Homepage: <http://libvisual.org/>

Source: git <https://github.com/Libvisual/libvisual>

Docs: <http://libvisual.org/docs/>

Docs: <http://libvisual.org/docs/annotated.html>

Docs: <https://github.com/Libvisual/libvisual/wiki>

Docs: <https://github.com/Libvisual/libvisual/wiki/Documentation>

Docs: <https://github.com/Libvisual/libvisual/wiki/Hacking-guide>

libvisual is a C API for music visualizations which enables integration and reuse with a number of media players.

- <https://github.com/Libvisual/libvisual/tree/master/libvisual-plugins/plugins/actor>
- <https://github.com/Libvisual/libvisual/tree/master/libvisual-plugins/plugins/input>
- <https://github.com/Libvisual/libvisual/tree/master/libvisual-plugins/plugins/morph>

MilkDrop

Wikipedia: <https://en.wikipedia.org/wiki/MilkDrop>

Homepage: <http://www.geisswerks.com/milkdrop/>

Source: <http://sourceforge.net/projects/milkdrop2/>

File Extension: `.milk`

Docs: http://www.geisswerks.com/about_milkdrop.html

Docs: <http://www.geisswerks.com/milkdrop/milkdrop.html>

Docs: http://www.geisswerks.com/milkdrop/milkdrop_preset_authoring.html

MilkDrop is a visualization tool which runs *MilkDrop presets*

- <https://en.wikipedia.org/wiki/MilkDrop>
- *Kodi* supports MilkDrop visualizations.

MilkDrop Presets

Docs: http://www.geisswerks.com/milkdrop/milkdrop_preset_authoring.html

- <https://github.com/xbmc/xbmc/tree/master/addons/visualization.milkdrop/presets>
- <http://www.theisozone.com/downloads/xbox/homebrew-apps/xbmc-visualizations/> (2007, ~9000 presets)
<http://forum.kodi.tv/showthread.php?tid=24616>
- http://ghostco.de/milkdrop_pack/
- <http://www.milkdrop.tk/>
- <http://forums.winamp.com/forumdisplay.php?f=84>
- http://www.geisswerks.com/milkdrop/favorite_presets_2013_12.zip

ProjectM

Homepage: <http://projectm.sourceforge.net/>

Project: <http://sourceforge.net/projects/projectm/>

Source: `git git://git.code.sf.net/p/projectm/code`

ProjectM is an *Open Source* implementation of *MilkDrop* with *OpenGL*.

- *MilkDrop Presets* work with ProjectM
- ProjectM is available for Linux, Unix
- ProjectM is available for iOS and Android

Resolume

Homepage: <http://resolume.com>

Download: <http://resolume.com/download/>

Docs: <http://resolume.com/download/>

Docs: <http://resolume.com/software/tutorials>

Docs: <http://resolume.com/manual/en/r4/start>

Resolume Arena and Avenue are realtime DVJ audiovisual production tools for live visuals.

- Resolume Arena is designed for **multi-projector setups** (“projection mapping”, “projection blending”)

VVVV

Wikipedia: <https://en.wikipedia.org/wiki/Vvvv>

Homepage: <http://vvvv.org/>

Download: <http://vvvv.org/downloads>

Docs: <http://vvvv.org/documentation/documentation>

Docs: <http://vvvv.org/documentation/devvvveloping>

vvvv (“v4”) is a realtime audiovisual production tool for Windows written in Delphi and C# .NET.

Web Production

M3U

Wikipedia: <https://git.xiph.org/flac.git>

M3U is an audio playlist format.

- An M3U file includes unqualified file *URLs* (e.g. `C:/music/file.mp3` or `/srv/music/file.mp3`)
- An M3U file includes HTTP *URLs* (e.g. “
- *VLC* can playback (read) *M3U* playlists.

PLS

Wikipedia: [https://en.wikipedia.org/wiki/PLS_\(file_format\)](https://en.wikipedia.org/wiki/PLS_(file_format))

PLS is an audio playlist format.

- ShoutCAST / IceCAST streams are shared in *PLS* format.
- *VLC* can playback (read) *PLS* playlists.

Popcorn Maker

Homepage: <https://popcorn.webmaker.org/>

Popcorn Maker is an *Open Source* tool for creating multitrack web media presentations.

- Popcorn Maker is a Mozilla Webmaker tool: <https://webmaker.org/en-US/tools>

Stream Annotation

Media codecs do not solve for commenting, other tools support commenting on points or ranges of timecodes (e.g. “this part is great!” or “who is that?”)

- *OpenAnnotation*
- SoundCloud, MixCloud

OpenAnnotation

OpenAnnotation is an *Open Source RDF* standard for threaded commenting on things with URLs and URIs.

OpenAnnotation *annotations* (comments with metadata) can be stored on (*synced with*) a central server, generally under a key, URN, or URI:

- Key: yA-NYmO8dDk
- URN: urn:x-youtubevideo:yA-NYmO8dDk
- URL: <https://www.youtube.com/watch?v=yA-NYmO8dDk>
- URL: https://www.youtube.com/watch?v=yA-NYmO8dDk&list=PLt_DvKGI_QLbqVMAiFRzts--Y9xZvxLDG&index=2
- URL: <https://youtu.be/yA-NYmO8dDk>
- ShortURL: <https://goo.gl/J4NJXs>
- Hypothesis is one implementation of an *OpenAnnotation* RDF server API (Python, Pyramid)
- There's a Javascript sidebar bookmarklet for Hypothesis *OA* comments.
- There's a Firefox sidebar extension for Hypothesis *OA* comments.
- There's a Javascript widget for Hypothesis *OA* comments:

```
<script async defer src="//hypothes.is/embed.js"></script>
<!-- show comments by default -->
<script>window.hypothesisConfig=function() {return{showHighlights:true};</script>
```

- There's a bookmarklet for Hypothesis *OA* comments: Hypothesis Bookmarklet

owl:sameAs: [Knowledge Engineering > Web Standards > OA](#)

XSPF

Wikipedia: https://en.wikipedia.org/wiki/XML_Shareable_Playlist_Format

Homepage: <http://xspf.org/>

Standard: <http://xspf.org/xspf-v1.html>

Docs: <http://xspf.org/applications/>

XSPF (*XML Shareable Playlist Format*) is an *Open Source* media playlist format for files with *URIs*, *URNs*, and *URLs*.

- XSPF understands `file://`-schema-prefixed local paths.
- XSPF understands `http://`-prefixed remote paths.
- XSPF is designed to be utilized with a **content resolver**; which CAN or COULD lookup alternate representations of resources from a given set of resource collections.
- “What a playlist is not” <http://xspf.org/xspf-v1.html#rfc.section.3.2>
- XSPF solves for all of the use cases of *M3U* and *PLS*.
- *VL*C can playback (read) *XSPF* playlists.

What cannot be done with XSPF?

- Cue points, crossfading, gap times

- [HTML content schema.org/description_HTML](http://schema.org/description_HTML)
- schema.org/CreativeWork

JSPF

Homepage: <http://xspf.org/jspf/>

JSPF (*JSON Shareable Playlist Format*) is an *Open Source* media playlist format for files with *URIs*, *URNs*, and *URLs*.

- [] Find/create *JSON-LD* @context for *JSPF*.

Web Distribution

https://en.wikipedia.org/wiki/Streaming_media

https://en.wikipedia.org/wiki/New_media

- New *Web Media (Web Content)* are shared through *Web Standards* like *TCP/IP*, *HTTP*, *HTML*, and *WebRTC*
- Streaming a video with seek and rewind is a bit different than downloading / streaming / buffering a file from beginning to end
 - *BitTorrent*, *HTTP Live Streaming*, and *MPEG-DASH* download segments of files over *HTTP* by specifying the `Content-Range: header` in GET requests
- Web scale audio / video requires less bandwidth when efficiently distributed through a file / stream distribution / caching / archiving network:
 - *CDN*
 - *BitTorrent*
 - *Named Data Networking*

CDN

Wikipedia: https://en.wikipedia.org/wiki/Content_delivery_network

https://en.wikipedia.org/wiki/Content_delivery_network#Notable_content_delivery_service_providers

<https://en.wikipedia.org/wiki/Cache#Computing>

A CDN (“Content Delivery Network”) is a service and infrastructure designed for serving *Web Content* efficiently and geodistributedly.

CDN Services:

- Akamai (OpenStack)
- *Amazon AWS* CloudFront
- CDNJS (free): <https://cdnjs.com/libraries>
- CloudFlare (free plan; optional DNS HTTPS static file compression)
- Fastly: *GitHub Pages*, PyPI

- Google: <https://developers.google.com/speed/libraries/>
- Rackspace CDN (OpenStack Swift)

CDN APIs

- OpenStack Poppy CDN API (+ OpenStack Swift)
<https://wiki.openstack.org/wiki/Poppy>
- libcloud Object Storage API:
https://libcloud.readthedocs.io/en/latest/supported_providers.html#object-storage
- *Ceph* Object Gateway: *Amazon AWS* S3 API and OpenStack Swift API
<http://ceph.com/docs/next/radosgw/>

BitTorrent

Wikipedia: <https://en.wikipedia.org/wiki/BitTorrent>
https://en.wikipedia.org/wiki/Comparison_of_BitTorrent_clients
https://en.wikipedia.org/wiki/Comparison_of_BitTorrent_tracker_software

BitTorrent is a file distribution protocol built on top of *HTTP*.

- A BitTorrent client downloads pieces of files over *HTTP* from **peers** discovered through a tracker or another peer discovery protocol (such as a *DHT*).
- BitTorrent magnet URIs open in an installed BitTorrent downloading program (e.g. BitTorrent, uTorrent, Transmission (Gnome), Deluge (Gnome))
- There are programs to retrieve the latest `.torrent` files from RSS and Atom feeds (e.g. a released [episode] of a series or topic)
- BitTorrent can be run headlessly (as a server daemon) e.g. with an HTTP API, SMTP pattern, web interface, or SSH for enqueueing torrents to download to a central location.
- BitTorrent does not preserve file permissions or extended attributes, but archives can.
- For a time, BitTorrent accounted for the largest slice of web traffic; but “now accounts for only 6.3% of total traffic in North America”.

<https://www.sandvine.com/trends/global-internet-phenomena/>

BitTorrent Torrent File A BitTorrent `.torrent` file includes:

- SHA-1 checksum hashes of all the pieces of all the files
- (optional) a *DHT* key (for trackerless peer discovery)
- (optional) a list of HTTP *BitTorrent Tracker* URLs
- (optional) a list of HTTP *Web Seeds*.

https://en.wikipedia.org/wiki/Torrent_file

BitTorrent Tracker A BitTorrent Tracker maintains a list of recently connected peers from which matching pieces can be downloaded.

BitTorrent Client A BitTorrent Client opens a *BitTorrent Torrent File*, registers with the given *BitTorrent Trackers*, looks for peers in a *DHT*, downloads the pieces of the files listed by choosing faster (more efficient, less costly, ~more local) peers to download from, and, according to the configuration, uploads pieces to other peers.

BitTorrent Seed A BitTorrent Seed is a complete, checksummed copy of all of the pieces of all the files in a *BitTorrent File* which a peer is sharing through a *BitTorrent tracker* or a *DHT*.

Web Seeding Web Seeding is when one or more *HTTP* servers serve default standby *seeds* (thus ensuring *Availability* when no-one is online).

https://en.wikipedia.org/wiki/BitTorrent#Web_seeding

Use Cases:

- <http://academictorrents.com/>
- <https://tug.org/mactex/MacTeX.pkg.torrent>
- <http://torrent.ubuntu.com:6969>
- <https://torrent.fedoraproject.org/>
- *debtorrent*
- <https://bundles.bittorrent.com/>

Note: “Scrambled” DRM content (which may or may not just work when I just copy the file to my other device) can be shared over BitTorrent all the same.

debtorrent

Homepage: <https://wiki.debian.org/DebTorrent>

Standard: <https://wiki.debian.org/DebTorrent/Protocol>

Source: `svn svn://svn.debian.org/debtorrent/debtorrent/trunk`

debtorrent downloads APT DEB packages with *BitTorrent* for Linux distributions like Ubuntu and/or Debian.

- debtorrent can efficiently transfer packages to many local or remote machines (instead of downloading the same file from the other side of the network multiple times)

```
sudo apt-get install debtorrent
echo 'deb debtorrent://localhost:9988/foo distro main' | \
sudo tee -a /etc/apt/sources.list
```

- <https://github.com/saltstack-formulas/apt-formula/blob/master/apt/transports/debtorrent.sls>

webtorrent

Source: `git https://github.com/feross/webtorrent`

webtorrent is a *BitTorrent* client written in Javascript that can run in browsers with *WebRTC* support or `node.js`.

- With `nodejs`, webtorrent can stream to AirPlay, Chromecast, *VLC*
- webtorrent can work with *HTML5* `<video>` tags

Named Data Networking

Wikipedia: https://en.wikipedia.org/wiki/Named_data_networking

Steam

Wikipedia: [https://en.wikipedia.org/wiki/Steam_\(software\)](https://en.wikipedia.org/wiki/Steam_(software))

Steam is a game and game content distribution system.

- Steam is part of SteamOS
- Steam registers DRM licenses with Steam for downloading files
- Steam does not yet download files from more local peers (e.g. everyone at the LAN party purchases the game)

Kerbal Space Program

Wikipedia: https://en.wikipedia.org/wiki/Kerbal_Space_Program

Homepage: <https://kerbalspaceprogram.com/>

Download: <http://store.steampowered.com/app/220200/>

Docs: <http://wiki.kerbalspaceprogram.com/wiki/>

Docs: http://wiki.kerbalspaceprogram.com/wiki/Version_history

Kerbal Space Program is a space flight simulator game.

KERBAL SPACE PROGRAM IS A MULTI-GENRE GAME WHERE THE PLAYERS CREATE THEIR OWN SPACE PROGRAM.

In KSP, you must build a space-worthy craft, capable of flying its crew out into space, without killing them. At your disposal is a collection of parts, which must be assembled to create a functional ship. Each part has its own function and will affect the way a ship flies (or doesn't).

- Kerbal Space Program is distributed over the *Steam* network
- Kerbal Space Program has many cool **mods**: <http://www.curse.com/ksp-mods/kerbal>

Tube Sites

Fueled by the dramatic growth of YouTube, the term “Tube Site” refers to any of a number of sites serving (seekable) video; initially with Flash Video *FLV* and now with *HTML5* and *HTTP Live Streaming* and *MPEG-DASH*.

Features

- GIF thumbnails
- seekable thumbnails
- download links
- download for my device links
- threaded comments, reviews, upvotes, downvotes

- *Stream Annotation*
- playlists this item is contained in
- featured playlists
- related content

HTTP Live Streaming

Wikipedia: https://en.wikipedia.org/wiki/HTTP_Live_Streaming

Standard: <http://tools.ietf.org/html/draft-pantos-http-live-streaming>

File Extension: .m3u8

Homepage: <https://developer.apple.com/streaming/>

Docs: <https://developer.apple.com/library/ios/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/Introduction/Introduction.html>

HTTP Live Streaming (HLS) is a draft *IETF* standard for streaming playlists of audio and video at various resolutions over *HTTP*.

- https://en.wikipedia.org/wiki/HTTP_Live_Streaming#Supported_players_and_servers
- HLS downloads and buffers file segments.
- HLS specifies optional AES Digital Rights Management (“scrambling”)
- *nginx-rtmp-module* can host *HLS* streams.
- *VLC* can play *HLS* streams.
- Safari and OSX and iOS have native support for *HTML5* HLS.

MPEG-DASH

Wikipedia: https://en.wikipedia.org/wiki/Dynamic_Adaptive_Streaming_over_HTTP

Standard: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=65274

Standard: <http://webstore.ansi.org/RecordDetail.aspx?sku=ISO/IEC+23009-1:2014>

MPEG-DASH is an *ISO* standard for streaming audio and video content with adaptive bitrates over *HTTP*.

- MPEG-DASH downloads and buffers file segments.
- MPEG-DASH selects from a number of possible resolutions according to available bandwidth and processing capacity.
- MPEG-DASH specifies optional *HTML5 Encrypted Media Extensions* DRM
- MPEG-DASH segments may contain MPEG-2 files in an MP4 container or other combinations of codecs and containers
- It currently costs \$265 USD to download the MPEG-DASH standard, which is not necessary to enjoy MPEG-DASH adaptability.
- *nginx-rtmp-module* can host *MPEG-DASH* streams.
- *VLC* can play *MPEG-DASH* streams.
- Chromecast, YouTube, NetFlix support *HTML5 MPEG-DASH*

nginx-rtmp-module

Source: <https://github.com/arut/nginx-rtmp-module>

Docs: <https://github.com/arut/nginx-rtmp-module/wiki>

Docs: <https://github.com/arut/nginx-rtmp-module/wiki/Installing-on-Ubuntu-using-PPAs>

nginx-rtmp-module is an *Open Source* module for the nginx web server which implements *RTMP*, *HTTP Live Streaming*, and *MPEG-DASH* for hosting streaming audio and video written in C.

- nginx-rtmp-module uses *FFmpeg* for online audio/video transcoding
- nginx-rtmp-module can record streams to seekable Flash Video *FLV* files
- nginx-rtmp-module works with *H.264*, *AAC*, *MPEG-DASH* audio / video compression / decompression codecs

Video Chat

<https://en.wikipedia.org/wiki/Videoconferencing>

<https://en.wikipedia.org/wiki/Videotelephony>

Video chat: what might've been an exercise in *Collaboration Engineering*:

- Goals, Patterns:
 - *Seven Layer Model of Collaboration*, (Goals, Patterns, Activities, Products, Techniques, Tools, Scripts)
 - *Six Patterns of Collaboration* (Generate, Reduce, Clarify, Organize, Evaluate, Build Consensus)
 - *ThinkLets* (Patterns, Techniques, Scripts)
- Goals: *Digital Stand Up Meeting* (Pattern: *Three Questions*)
- Tools: *RTMP* (Flash conferencing tools)
- Tools: *WebRTC* (newer Browsers with WebRTC support or WebRTC plugins)

Models for interactive video chat:

- Video Relay Server (*RTMP*, *nginx-rtmp-module*)
 - Clients capture, encode, and push video to server
 - Server [re]encodes stream and pushes video to other client(s)
- Browser-to-Browser (*WebRTC*)
 - Clients connect in a mesh and relay data streams
 - Clients can be browsers or hosted applications (e.g. *SaaS*)
 - *WebRTC* is also useful for centralized *Web Distribution*

Media Tools

<http://distrowatch.com/search.php?category=Multimedia>

DreamStudio

Homepage: <http://www.celeum.com/dreamstudio/>

Source: <http://sourceforge.net/projects/dreamstudio/>

Download: http://sourceforge.net/projects/dreamstudio/files/Ceum_DreamStudio_2015/

Download: <http://sourceforge.net/projects/dreamstudio/files/latest/download>

DistroWatch: <http://distrowatch.com/table.php?distribution=dreamstudio>

Ceum DreamStudio is a distribution of Ubuntu Linux with lots of great packages and a kernel tuned for media production.

- DreamStudio includes software with a diverse set of software licenses.

Ubuntu Studio

Wikipedia: https://en.wikipedia.org/wiki/Ubuntu_Studio

Homepage: <http://ubuntustudio.org/>

Download: <https://ubuntustudio.org/download/>

DistroWatch: <http://distrowatch.com/table.php?distribution=ubuntustudio>

Docs: <https://help.ubuntu.com/community/UbuntuStudio>

Docs: <https://ubuntustudio.org/tour/audio/>

Docs: <https://ubuntustudio.org/tour/graphics/>

Docs: <https://ubuntustudio.org/tour/video/>

Docs: <https://ubuntustudio.org/tour/photography/>

Docs: <https://ubuntustudio.org/tour/publishing/>

Ubuntu Studio is a distribution of Ubuntu Linux with lots of great packages and a kernel tuned for media production.

- Ubuntu Studio includes software with a diverse set of all Free and *Open Source* software licenses.

VLC

Wikipedia: https://en.wikipedia.org/wiki/VLC_media_player

Homepage: <https://www.videolan.org/vlc/>

Download: <https://www.videolan.org/vlc/#download>

Source: `git git://git.videolan.org/vlc.git`

Docs: <https://www.videolan.org/videolan/mirrors.html>

Docs: https://wiki.videolan.org/Developers_Corner

Docs: <https://wiki.videolan.org/GetTheSource/>

Docs: <https://wiki.videolan.org/VLC/>

Docs: <https://wiki.videolan.org/Documentation>

Docs: https://wiki.videolan.org/Documentation:Play_HowTo

Docs: https://wiki.videolan.org/Documentation:Streaming_HowTo

Docs: https://wiki.videolan.org/Documentation:Streaming_HowTo_New/

Docs: <https://wiki.videolan.org/Documentation:Modules/>

Home A/V

Standards

DIAL

Wikipedia: https://en.wikipedia.org/wiki/Discovery_and_Launch

Homepage: <http://www.dial-multiscreen.org/>

Standard: <http://www.dial-multiscreen.org/dial-protocol-specification>

DIAL (*Discovery and Launch*) is a standard for discovering media devices and launching media resources built upon *SSDP*, *UPnP*, and *HTTP*.

- A DIAL Server (a “1st screen device”; e.g. a TV or streaming device) receives DIAL requests from a DIAL client.
- A DIAL Client (a “2nd screen device”; e.g. a phone or a PC) sends DIAL requests to a DIAL server.
- DIAL Clients discover DIAL devices with *DIAL Service Discovery*.
- DIAL Clients interact with DIAL devices over a *DIAL REST Service API*.

DLNA

Wikipedia: https://en.wikipedia.org/wiki/Digital_Living_Network_Alliance

Homepage: <http://www.dlna.org/>

DLNA: Digital Living Network Alliance DLNA is a standards group and a standard for networked devices; particularly audio/video media devices like standalone media players and networked stereos.

Google Cast SDK

Wikipedia: <https://en.wikipedia.org/wiki/Chromecast>

Homepage: <https://developers.google.com/cast/>

Source: <https://github.com/googlecast>

Docs: https://en.wikipedia.org/wiki/List_of_apps_with_Google_Cast_support

Docs: <https://developers.google.com/cast/>

Docs: <https://developers.google.com/cast/docs/developers>

Docs: <https://developers.google.com/cast/docs/reference/>

Google Cast SDK powers *Chromecast* devices.

- A Cast sender application pushes a URI to a Cast receiver application.
- The Cast receiver application then retrieves the resource/app identified by the URI; saving the battery of the Cast sender application.
 - Bluetooth Audio (A2DP,) and Chromecast **full device audio/video cast** require audio [recompression and] transmission, which drains a mobile device battery.

SSDP

Wikipedia: https://en.wikipedia.org/wiki/Simple_Service_Discovery_Protocol

Standard: <https://tools.ietf.org/html/draft-cai-ssdp-v1>

SSDP (*Simple Service Discovery Protocol*) is a networking protocol for discovering services over HTTPU (*HTTP* over UDP) unicast and multicast; without DNS or DHCP.

UPnP

Wikipedia: https://en.wikipedia.org/wiki/Universal_Plug_and_Play

Homepage: https://en.wikipedia.org/wiki/Universal_Plug_and_Play

Standards: <http://openconnectivity.org/upnp/specifications>

Universal Plug and Play is a group of standards for networked devices built on *HTTP*, *XML*, and SOAP web services.

- https://en.wikipedia.org/wiki/Comparison_of_UPnP_AV_media_servers
- https://en.wikipedia.org/wiki/List_of_UPnP_AV_media_servers_and_clients

Zeroconf

Wikipedia: <https://en.wikipedia.org/wiki/Zeroconf>

Zeroconf/Bonjour are standards for zero-configuration networking; wherein networked devices assume local IP addresses if necessary, and broadcast their capabilities over multicast DNS.

- Devices must be on the same LAN/WLAN segment (!?)
 - [] Otherwise forwarding / ebttables ?
 - * [] pulseaudio-dlna

10-foot interfaces

Gaming Consoles

[...]

- *Steam* Machine (SteamOS)
- NVIDIA Shield (AndroidTV)
- XBox
 - *XBMC* (now *Kodi*) started out as a an application for modded XBoxen way back in 2002.
- XBox 360 (joysticks tied to visualizations)
- XBox One
- PS3
- PS4

Standalone Media Devices

Pros:

- Upgradeable
- Interchangeable

Cons:

- Extra HDMI, Power
- Remote Control Integration

Android TV

Wikipedia: https://en.wikipedia.org/wiki/Android_TV

- Android TV is originally based on Android 5.0 (“Lollipop”)
- Android TV supports the *Chromecast* API
- Android TV supersedes Google TV (which was also Android-based)

Chromecast

Wikipedia: <https://en.wikipedia.org/wiki/Chromecast>

Homepage: <https://www.google.com/chromecast/>

- 1: 2013, 2: 2015
- In: Power: MicroUSB
- Configure WiFi w/ Android/iOS device and *Chromecast* app
- Cast content from Android, computer (Chrome, ChromeOS), iOS to Chromecast
- Out: *HDMI*

Chromecast Audio

Wikipedia: <https://en.wikipedia.org/wiki/Chromecast>

Homepage: <https://www.google.com/chromecast/speakers/>

Chromecast Audio turns just about any stereo with an AUX input into a web-connected audio player.

- In: Power: MicroUSB
- Configure WiFi w/ Android/iOS device and *Chromecast* app
- Cast content from Android, computer (Chrome, ChromeOS), iOS to Chromecast Audio
- Out: 3.5mm Headphone Jack
- Out: Optical Audio Out

Apple TV

Wikipedia: https://en.wikipedia.org/wiki/Apple_TV

Homepage: <https://www.apple.com/tv/>

Docs: <https://www.apple.com/tv/compare/>

- 1: 2007, 2: 2010, 3: 2012/2013, 4: 2015
- Extremely simple remote
- Supports Bluetooth Keyboards, iOS devices

Kodi

Wikipedia: [https://en.wikipedia.org/wiki/Kodi_\(software\)](https://en.wikipedia.org/wiki/Kodi_(software))

Kodi (formerly XBMC) is a “10 foot” media playback application designed for use with a remote.

Kodi addons are written in Python and *XML*.

- Kodi runs on Windows
- Kodi runs on OSX
- Kodi runs on Linux
- Kodi runs on Android
- Kodi runs on iOS
- Kodi runs on *Apple TV*
- Kodi runs on FreeBSD
- Kodi runs on *ARM*, MIPS, PowerPC, *x86*, x86-64
- Kodi supports *MilkDrop* audio visualizations.

MythTV

Wikipedia: <https://en.wikipedia.org/wiki/MythTV>

MythTV (2002) is a networked DVR server and protocol standard with a number of client implementations.

- MythTV supports [remote] tuner cards
- MythTV supports TV listings
- MythTV supports IR remotes, timeshifting, Linux, Windows

Plex

Wikipedia: [https://en.wikipedia.org/wiki/Plex_\(software\)](https://en.wikipedia.org/wiki/Plex_(software))

Homepage: <https://plex.tv/>

Download: <https://plex.tv/downloads>

Plex Media Server Plex Media Server is a free app which runs on a computer or a NAS device and serves media to Plex client apps.

Plex Home Theater Plex Home Theater is a free *10-foot interface* client app for a computer attached to a TV (e.g. with *VGA*, *DVI*, or *HDMI*).

Plex Pass Plex Pass is an optional premium subscription service upgrade.

<https://plex.tv/subscription/about>

- Plex is/was originally a fork of *Kodi*
- Plex works with Windows
- Plex works with Windows Phone
- Plex works with OSX
- Plex works with Linux
- Plex works with Android
- Plex works with *Chromecast*
- Plex works with FreeBSD
- Plex works with iOS
- Plex works with *Apple TV*
- Plex works with Xbox 360
- Plex works with Xbox One
- Plex works with PS3
- Plex works with PS4
- Plex works with *Roku*

Roku

Wikipedia: <https://en.wikipedia.org/wiki/Roku>

Homepage: <https://www.roku.com/>

Store: <https://channelstore.roku.com/>

- Through the Roku Channel Store, Roku supports Google Play, Netflix, PBS, HBO Go, HBO Now, Epix, Showtime, and a number of other channels:
- Roku supports Amazon Instant VOD
- Roku apps are written in a Roku-specific language called “BrightScript”
- Roku OS is derived from Linux
- The Roku remotes are all very simple and easy.

- The Roku 3 remote has a headphone jack for private listening (*Wi-Fi Direct*)

seeAlso:

- *Art & Design*
- *Web Design*
- responsive web design

Human-Computer Interaction (HCI)

Wikipedia: https://en.wikipedia.org/wiki/Human%E2%80%93computer_interaction

WikipediaCategory: https://en.wikipedia.org/wiki/Category:Human%E2%80%93computer_interaction

- https://en.wikipedia.org/wiki/Outline_of_human%E2%80%93computer_interaction
 - History of human-computer interaction: https://en.wikipedia.org/wiki/Outline_of_human%E2%80%93computer_interaction#History_of_human-computer_interaction

Q1. How long have we been doing this? Q2. How do we make this better?

Usability

Wikipedia: <https://en.wikipedia.org/wiki/Usability>

WikipediaCategory: <https://en.wikipedia.org/wiki/Category:Usability>

- https://en.wikipedia.org/wiki/Usability_engineering

Experience Design (XD)

Wikipedia: https://en.wikipedia.org/wiki/Experience_design

User Experience (UX)

Wikipedia: https://en.wikipedia.org/wiki/User_experience

Wikipedia: https://en.wikipedia.org/wiki/User_experience_design

User Interface (UI)

Wikipedia: https://en.wikipedia.org/wiki/User_interface_design

Wikipedia: https://en.wikipedia.org/wiki/User_interface

- https://en.wikipedia.org/wiki/Heuristic_evaluation

Information Design

Wikipedia: https://en.wikipedia.org/wiki/Information_design

Wikipedia: https://en.wikipedia.org/wiki/Team_building

- https://en.wikipedia.org/wiki/Leadership_development
- https://en.wikipedia.org/wiki/Organization_development
- https://en.wikipedia.org/wiki/Tuckman's_stages_of_group_development
- “From Comfort Zone to Performance Management” White (2008) http://www.academia.edu/460313/From_Comfort_Zone_to_Performance_Management
- <https://en.wikipedia.org/wiki/Kaizen>

Personal Development

Wikipedia: https://en.wikipedia.org/wiki/Personal_development

- https://en.wikipedia.org/wiki/Life_skills#See_also
- <https://en.wikipedia.org/wiki/Virtue>

Golden Rule

Wikipedia: https://en.wikipedia.org/wiki/Golden_Rule

- <https://en.wikipedia.org/wiki/Empathy>
- <https://en.wikipedia.org/wiki/Compassion>

Emotional Intelligence

Wikipedia: https://en.wikipedia.org/wiki/Emotional_intelligence
https://en.wikipedia.org/wiki/Industrial_and_organizational_psychology

https://en.wikipedia.org/wiki/Contrasting_and_categorization_of_emotions

<https://en.wikipedia.org/wiki/Motivation>

- https://en.wikipedia.org/wiki/Theory_X_and_Theory_Y
- https://en.wikipedia.org/wiki/Theory_Z

https://en.wikipedia.org/wiki/Psychological_manipulation

https://en.wikipedia.org/wiki/The_Five_Dysfunctions_of_a_Team

[https://en.wikipedia.org/wiki/Coping_\(psychology\)](https://en.wikipedia.org/wiki/Coping_(psychology))

https://en.wikipedia.org/wiki/Defence_mechanisms#Level_4:_Mature

https://en.wikipedia.org/wiki/Positive_psychology#Broad_theories

https://en.wikipedia.org/wiki/Happiness#Research_results

Collaboration Engineering

<https://en.wikipedia.org/wiki/Collaboration>

“Collaboration Engineering: Foundations and Opportunities” de Vreede (2009) <http://aisel.aisnet.org/jais/vol10/iss3/7/>

Collaboration Engineering focuses on mission-critical collaborative tasks. A mission-critical task is one that creates substantial value, or that reduces the risk of a substantial loss of value for organizational stakeholders. Collaboration Engineering further focuses on processes for mission-critical tasks that are recurring and must be executed frequently.

Seven Layer Model of Collaboration

“A Seven-Layer Model of Collaboration: Separation of Concerns for Designers of Collaboration Systems” Briggs (2009) <http://aisel.aisnet.org/icis2009/26/>

1. Goals
2. Products
3. Activities
4. Patterns of Collaboration
5. Techniques
6. Tools
7. Scripts

Six Patterns of Collaboration

“Defining Key Concepts for Collaboration Engineering” Briggs (2006) <http://aisel.aisnet.org/amcis2006/17/>

- Generate
- Reduce
- Clarify
- Organize
- Evaluate
- Build Consensus

ThinkLets

“ThinkLets: Achieving Predictable, Repeatable Patterns of Group Interaction with Group Support Systems (GSS)” http://www.academia.edu/259943/ThinkLets_Achieving_Predictable_Repeatable_Patterns_of_Group_Interaction_With_Group_Support_Systems_GSS_

<https://scholar.google.com/scholar?q=thinklets>

Collaborative Software

https://en.wikipedia.org/wiki/Collaborative_software

https://en.wikipedia.org/wiki/Collaborative_software#Classification

Table 22.1: Classifying Collaborative Software

Features	X_n	X_1	X-2
Core capabilities			
Data contributions			
Data relationships			
Data limits			
Data persistence			
Actions			
Synchronicity			
Identifiability			
Access Controls			
Roles			
Awareness indicators			
Interruption alerts			
Instructions			

- “Toward a Taxonomy of Groupware Technologies” (2008)

<https://books.google.com/books?id=ONRsCQAAQBAJ&pg=PA309>

<https://www.worldcat.org/title/groupware-design-implementation-and-use-14th-international-workshop-criwg-2008-omaha-ne-usa-september-14-18-2008-rev-oclc/760695691>

- “Classification of Collaboration Technology” (2013)

<https://books.google.com/books?id=gTR0AgAAQBAJ&pg=PA44>
<https://www.worldcat.org/title/collaboration-systems-concept-value-and-use/oclc/867930692>

See also:

- *Schema.org*
 - <http://schema.org/Action>
 - <http://schema.org/ReactAction>
 - <http://schema.org/comment> -> <http://schema.org/Comment>
 - <http://schema.org/review> -> <http://schema.org/Review>
 - <http://schema.org/InteractionCounter>
 - <http://schema.org/docs/full.html>

The same page

Wiktionary: https://en.wiktionary.org/wiki/on_the_same_page

“On the same page”

- [https://www.google.com/search?q="on+the+same+page"](https://www.google.com/search?q=)
- [https://www.bing.com/search?q="on+the+same+page"](https://www.bing.com/search?q=)
- [https://search.yahoo.com/search?p="on+the+same+page"](https://search.yahoo.com/search?p=)
- <https://wrdrd.github.io/docs/consulting/team-building#the-same-page> (Sphinx, Git
 - Source, Edit, History, Annotate sidebar links
 - <https://pypi.python.org/pypi/sphinxcontrib-srclinks>

Group Support Systems

https://en.wikipedia.org/wiki/Electronic_meeting_system

https://en.wikipedia.org/wiki/List_of_collaborative_software

- https://en.wikipedia.org/wiki/Etherpad#Etherpad_Lite
 - https://en.wikipedia.org/wiki/Operational_transformation

Text Templates

- Collaboration with Etherpad Lite and Thinklets: “As ____, we need to ____ because ____” (copy from the *Prompts* at the top)

Collaboration Checklist

- Read <https://wrdrd.github.io/docs/consulting/team-building>
- Collaboration Plan (*Seven Layer Model of Collaboration*)
 1. Goals
 2. Products
 3. Activities
 4. Patterns of Collaboration
 5. Techniques
 6. Tools
 7. Scripts
- Collaborate
 - *Generate* new list(s) of {problems, solutions}
 - *Reduce*
 - *Clarify*
 - *Organize*
 - *Evaluate* options from [ordered] set { A, B, [or] C }
 - * @usernameX: A, B, C because (e.g. code compiles, tests pass, docs look great)
 - Relevant Resources with URLs:
 - <https://gist.github.com/westurner/5704379>
 - <https://github.com/westurner/ldpnotes>
 - * @usernameY: C, B, A because (e.g. code compiles, tests pass, docs look great)
 - Relevant Resources with URLs:
 -
 - * @usernameZ: criteria:x = 400, criteria:y = 100, criteria:z = 5
 - * Build Consensus
 - Runoff Voting
 - $A = (3+1) = 4$
 - $B = (2+2) = 4$
 - $C = (1+3) = 4$
 - Weighted Criteria
 - criteria:x = 400

- criteria:y = 100
- criteria:z = 5
- A = {x, y}, B = {y,z}, C = {x,y,z}
- A = 500, B = 105, C = 505

See also:

- *Software Engineering > Digital Stand Up Meeting*
- *Collaboration Plan*

A collaboration plan for successfully working together to set and achieve team objectives.

Contact Information: Email, Phone

Shared Folder: .

Mailing List: .

Project Site: .

Sharing Files

Is this the newest version of [...]

Email attachments can be hard to find.

Versioned shared storage folders are so much easier to work with.

An email with a link to the (updated) files is helpful.

Mailing List

I forgot to CC [..]

I forgot to reply-all

Please sift through these n forwarded messages

For bringing new team members onboard, it's much easier to archive a mailing list and share links to specific emails and threads than to be forwarding and remembering to CC people.

As team correspondence, the context for a mailing list may be a bit less informal.

One continuous thread may very well be sufficient and easier to scroll through. For this reason it's helpful to put some forethought into the subject of the initial email message.

Keeping in mind the DRY principle and that it's possible to cross-post (x-post) a link to another message, some other ways to organize threads:

- One thread per team (e.g. design, development, marketing)
- One thread per objective (e.g. objectives, updates, followup)
- One thread per meeting (e.g. scheduling, planning, minutes, followup)

To keep volume down and everyone on the same page, sometimes it's easier to address everyone at the same time.

```
To: mailing-list@groups.google.com
From: wes.turner@gmail.com
Subject: Meeting 1/23/14

Great meeting. Here are a few minutes:

1.
2.
3.

@A
For next week, could you [...]

@B
The [...] look great, thanks.
```

Project Site

Where was that link to the [...]

Not strictly necessary, but helpful as a central collaboration point.

Links to:

- contact information
- shared folders
- mailing lists
- team calendar
- works in progress
- roadmap with dates and links to deliverables

Re: Document-based collaboration

Coming from a software development perspective, where:

- all changes are stored in version control
- all changes are tagged with an author
- everyone has access to central storage of every version
- each change affects a concrete set of files
- anyone can create a patch or submit a pull request to share their suggested modifications

Emailing document attachments is a wasteful way to collaborate.

> What version am I working on?

> Who wrote this?

> When was this change proposed?

> What did it look like before the merge?

Online office suites with live collaboration can be much more productive in that it's possible to review how something developed; but I like my distributed version control for good reason.

Who better to work out effective methods for online collaboration than software developers?

To this end, I propose a few simple processes for avoiding time wastage on document presentation:

- Contact Information Page (derived from a mailing list thread)
- Mailing List for archiving and relaying team correspondence
- Shared Storage with web interface and access control
- Project Site with WYSIWYG editor (derived from and driving mailing list activity)
 - Contact Information
 - Goals / Objectives / Roadmap
 - Team Policies / Procedures
 - * scheduling a meeting
 - * sending progress notifications
 - * sharing meeting minutes
 - Shared Storage Folder
 - Calendar Widget
 - * drawing from a calendaring system with iCal that I can overlay onto my other calendars

WRD R&D

Source: <https://github.com/wrdrd/docs>

Docs: <https://wrdrd.readthedocs.org/en/latest/>

Docs: <https://wrdrd.com/docs/>

Lead

- Wes Turner <wes@wrdrd.com>

Contributors

- Wes Turner <wes@wrdrd.com>

CHAPTER 25

License



WRD R&D Documentation is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

CHAPTER 26

Tools

Homepage: <https://westurner.github.io/tools/>

Source: git <https://github.com/westurner/tools>

wrdrd

A Python package with a collection of scripts for web consulting.

Installation

At the command line:

```
pip install -e git+https://github.com/wrdrd/docs#egg=wrdrd
```

Usage

wrdrd.tools contains a number of command line utilities.

domaintool

```
# Retrieve DNS and Whois information for a domain
domaintool wrdrd.github.io

# Print usage
domaintool --help
```

crawl

```
# Install requirements
pip install -r requirements.txt
```

```
# Download NLTK libraries
python -m nltk.downloader all

# Crawl a website
crawl -c http://wrdrd.github.io/

# Print HTML to stdout
crawl --html http://wrdrd.github.io/

# Print text to stdout
crawl --text http://wrdrd.github.io/

# Print usage
crawl --help
```

stripsinglehtml

```
# Strip Sphinx (single) HTML markup
stripsinglehtml ./_build/singlehtml/index.html > __index.html

# Print usage
stripsinglehtml --help
```

wrdrd.tools package

Submodules

wrdrd.tools.crawl module

wrdrd.tools.domain module

wrdrd.tools.domain.**check_google_dkim**(*domain*, *prefix*=*'google'*)

Check a Google DKIM DNS TXT record

Parameters

- **domain** (*str*) – DNS domain name
- **prefix** (*str*) – DKIM s= selector ('DKIM prefix')

Returns 0 if OK, 1 on error

Return type int

<https://support.google.com/a/answer/174126>

<https://admin.google.com/AdminHome?fral=1#AppDetails:service=email&flyout=dkim>

Note: This check function only finds “v=DKIM1” TXT records; it defaults to the default google prefix and **does not validate DKIM signatures**.

<http://dkim.org/specs/rfc4871-dkimbase.html#rfc.section.3.6.2.1>

<http://dkim.org/specs/rfc4871-dkimbase.html#rfc.section.A.3>

`wrdrd.tools.domain.check_google_dmarc` (*domain*)

Check a Google DMARC DNS TXT record

Parameters `domain` (*str*) – DNS domain name

Returns 0 if OK, 1 on error

Return type int

<https://support.google.com/a/answer/2466580>

<https://support.google.com/a/answer/2466563>

`wrdrd.tools.domain.check_google_domain` (*domain*, *dkim_prefix*='google')

Check DNS MX, SPF, DMARC, and DKIM records for a Google Apps domain

Parameters

- `domain` (*str*) – DNS domain

- `dkim_prefix` (*str*) – DKIM prefix (<prefix>._domainkey)

Returns nonzero returncode on failure (sum of returncodes)

Return type int

<https://support.google.com/a/answer/2716802>

`wrdrd.tools.domain.check_google_mx` (*domain*)

Check Google MX DNS records

Parameters `domain` (*str*) – DNS domain name

Returns 0 if OK, 1 on error

Return type int

https://support.google.com/a/topic/2716885?hl=en&ref_topic=2426592

`wrdrd.tools.domain.check_google_spf` (*domain*)

Check a Google SPF DNS TXT record

Parameters `domain` (*str*) – DNS domain name

Returns 0 if OK, 1 on error

Return type int

<https://support.google.com/a/answer/178723?hl=en>

`wrdrd.tools.domain.dig_all` (*domain*)

Get all DNS records with dig

Parameters `domain` (*str*) – DNS domain

Returns dig output

Return type str

`wrdrd.tools.domain.dig_dnskey` (*zone*)

Get DNSSEC DNS records with dig

Parameters `zone` (*str*) – DNS zone

Returns dig output

Return type `str`

`wrdrd.tools.domain.dig_mx` (*domain*)

Get MX DNS records with dig

Parameters `domain` (*str*) – DNS domain

Returns dig output

Return type `str`

https://en.wikipedia.org/wiki/MX_record

`wrdrd.tools.domain.dig_ns` (*domain*)

Get DNS NS records with dig

Parameters `domain` (*str*) – DNS domain

Returns dig output

Return type `str`

`wrdrd.tools.domain.dig_spf` (*domain*)

Get SPF DNS TXT records with dig

Parameters `domain` (*str*) – DNS domain

Returns dig output

Return type `str`

https://en.wikipedia.org/wiki/Sender_Policy_Framework

`wrdrd.tools.domain.dig_txt` (*domain*)

Get DNS TXT records with dig

Parameters `domain` (*str*) – DNS domain

Returns dig output

Return type `str`

`wrdrd.tools.domain.domain_tools` (*domain*)

Get whois and DNS information for a domain.

Parameters `domain` (*str*) – DNS domain name

Returns nonzero returncode on failure (sum of returncodes)

Return type `int`

`wrdrd.tools.domain.main` (**args*)

wrdrd.tools.domain main method

Parameters `args` (*list*) – commandline arguments

Returns nonzero returncode on failure (sum of returncodes)

Return type `int`

`wrdrd.tools.domain.nslookup` (*domain, nameserver=''*)

Get nslookup information with nslookup (resolve a domainname to an IP)

Parameters

• `domain` (*str*) – DNS domain

• `nameserver` (*str*) – DNS domain name server to query (default: `''`)

Returns nslookup output

Return type str

`wrdrd.tools.domain.whois` (*domain*)

Get whois information with whois

Parameters `domain` (*str*) – DNS domain

Returns whois output

Return type str

wrdrd.tools.stripsinglehtml module

Credits

WRD R&D

Source: <https://github.com/wrdrd/docs>

Docs: <https://wrdrd.readthedocs.org/en/latest/>

Docs: <https://wrdrd.com/docs/>

Lead

- Wes Turner <wes@wrdrd.com>

Contributors

- Wes Turner <wes@wrdrd.com>

History

Complete project history can be found at <https://github.com/wrdrd/docs/commits>.

0.1.0 (2014-1-11)

- First release.

0.2.0 (2014-10-14)

- BUG: Fix bugs in `crawl`, `domaintool`, and `stripsinglehtml`
- DOC: Add docstrings
- ENH,BUG: Switch to *sarge*
- ENH: Add *structlog*
- ENH: Add `wrdrd.tools.domain:check_google_dkim`
- DOC: Create `docs/wrdrd/readme.rst`

License

wrdrd is licensed under the terms of the New BSD (3-clause) license:

```
Copyright (c) 2014, WRD R&D
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
```

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the <organization> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

```
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

CHAPTER 28

Indices and tables

- `genindex`
- `modindex`
- `search`

W

`wrdrd.tools`, 330

`wrdrd.tools.domain`, 330

Symbols

0 (Year), 184
10-foot interfaces, 306
1000BASE-T, 69
1000BASE-X, 69
100BASE-T, 69
100BASE-TX, 69
10BASE-T, 68
10GBASE-T, 69
3D Modeling, 275
3G, 71
401(k), 241
401(k) match, 241
401(k) rollover, 241
40GBASE-T, 69
4G, 71
5 Star Linked Open Data, 190
5 Star Open Data, 190
5G, 71
802.11, 70
8P8C, 66

A

A&D, 271
AAC, 289
Accounting, 92
Accounting Equation, 248
Accumulo, 168
ACME Protocol, 33
Acoustical Engineering, 289
ACT, 268
Advance Care Directive, 243
Advanced Placement, 266
Advanced Visualization Studio, 293
Agile, 41
Agile Modeling, 44
AGP, 58
Algorithmic trading, 250
Amazon Payments, 93

Ampere, 123
Amps, 123
and Inference, 137
and metadata rows, 223
Android TV, 307
AP, 75, 266
Apache Avro, 177
Apache Beam, 169
Apache Hive, 170
Apache Lucene, 182
Apache Parquet, 171
Apache Solr, 182
Apache Spark, 171
ApacheNutch, 182
API, 78
App, 78
AppEngine, 84
Apple Pay, 93
Apple TV, 307
Application, 78
AppScale, 85
Ardour, 289
ARM, 56
ARP, 20, 73
Arrays, 148
Art & Design, 271
ASCII, 135
ASCIIMathML, 104
Asset, 248
Astronomical year numbering, 184
ASVAB, 268
ATA, 64
Audacity, 289
Audio Synthesis, 286
Audio/Visual Production, 293
Augmented Reality, 277
Availability, 90
Avro, 177
AWS, 87

B

Backups, 91
Backward induction, 233
BDD, 45
Beam, 169
Before Present, 185
Behavior Driven Development, 45
BFS, 150
BI, 94
BigTable, 169
Binary Prefixes, 118
Bisection, 53
Bit, 125
BitTorrent, 299
BitTorrent Client, **299**
BitTorrent Seed, **300**
BitTorrent Torrent File, **299**
BitTorrent Tracker, **299**
Blacklight, 229
Blazegraph, 166
Blender, 276
Blueprints, 166
BMP, 274
Bokeh, 112
Boolean Algebra, 138
Bootstrap, 26
BP, 185
Branching & Merging, 53
Brand, 11
Branding, 11
Breadth-first search, 150
Bridge, **74**
BSP, 174
btrfs, 156
Bug Bounty Program, 31
Bulk Synchronous Parallel, 174
Bup, 147
Business, 7
Business Books, 7
Business Continuity, 90
Business Entity Types, 234
Business Intelligence, 94
Business Metrics, 10
Business Modeling, 8
Business Plans, 9
Business Relationships, 9
Business ROI, 8
Business Strategy, 8
Business Systems, 91
Buy and hold, 250
Byte, 125
bzip2, 152

C

Calculus, 105
Candela, 125
Capital gains tax, 239
Capitalization Table, 248
Cash flow, 247
Cassandra, 169
CAT-5, 67
CAT-5e, 67
CAT-6, 67
CAT-6A, 67
Causality, 106
CD, 52
CDMA, 71
CDN, 298
Celeum DreamStudio, 303
Celsius, 124
Ceph, 159
Character encoding, 134
Character map, 134
Character set, 134
Charitable contributions, 240
Charity Navigator, 240
check_google_dkim() (in module wrdrd.tools.domain), 330
check_google_dmarc() (in module wrdrd.tools.domain), 331
check_google_domain() (in module wrdrd.tools.domain), 331
check_google_mx() (in module wrdrd.tools.domain), 331
check_google_spf() (in module wrdrd.tools.domain), 331
Checksums, 153
Chromecast, 307
Chromecast Audio, 307
ChucK, 286
CI, 54
CIFS, 159
Cinelerra, 280
CinePaint, 280
Citation Metadata, 225
CKAN, 215, 227
Class Central, 258
CLEP, 267
Cloud Application Layers, 80
Cloud Computing, 80
Clouds, 80
Code Golf, 40
Codelabels, 50
Codeset, 134
Cognitive Bias, 101
Collaboration Checklist, 317
Collaboration Engineering, 314
Collaboration Plan, 318
Collaborative Software, 315

- College Credit Highschool Courses, 266
 - Color Scheme, 11
 - Common Core, 266
 - Common Era, 185
 - Common Weakness Enumeration, 32
 - Composition (Visual), 273
 - Compound interest, 238
 - Compression Algorithms, 152
 - Computer Engineering, 54
 - Computer Science, 37
 - Computer Science Courses, 40
 - Computer Science Curricula, 37
 - Computer Science Exercises, 40
 - Computer-mediated Reality, 277
 - Concept, 11
 - Confidentiality, 90
 - Configuration Management, 90
 - Content, 24
 - Content Delivery Network, 298
 - Contingency plan, 232
 - Continuous Delivery, 52
 - Continuous Deployment, 54
 - Continuous Integration, 54
 - Control Characters, 134
 - Conversion, 10
 - Conversion Marketing, 10
 - Coordinated Universal Time, 186
 - Copy, 11
 - Copy (marketing), 11
 - CORBA, 175
 - Coursera, 258
 - CPU, 56
 - CRC, 153
 - Criteria, **38**
 - CRM, 94
 - CS, 37
 - Csound, 286
 - CSS, 25, 194
 - CSV, 198, 223
 - CSVW, 198, 223
 - CSVW and Units, 117
 - CUDA, 60
 - cup, 128
 - Curricula, **38**
 - Customer Relationship Management, 94
 - CWE, 32
 - Cyclical Redundancy Check, 153
- ## D
- D3, 113
 - D3.js, 113
 - Data Analysis, 106
 - Data Device Bus, 62
 - Data Dredging, 107
 - Data Engineering, 142
 - Data Grid, 181
 - Data Learning, 107
 - Data Mining, 107
 - Data Science, 94
 - Data Science Techniques, 109
 - Data Science Theory, 101
 - Data Science Tools, 108
 - Data Structures, 148
 - Data Visualization, 110
 - Data Visualization Tools, 110
 - Databases, 161
 - datasciencemasters.org, 95
 - Day, 122
 - Day trading, 250
 - Daylight Saving Time, 188
 - DBpedia, 210
 - DCMI, 204
 - dcterms, 204
 - dctypes, 204
 - DDFS, 159
 - debtorrent, 300
 - Decimal Time, 184
 - Deep Learning, 108
 - Defined benefit pension plan, 241
 - Deis, 86
 - Depth-first search, 150
 - Description Logic, 141
 - Descriptive Statistics, 106
 - DFS, 150
 - DHCP, **22, 73**
 - DHT, 173
 - DIAL, 305
 - dig_all() (in module wrdrd.tools.domain), 331
 - dig_dnskey() (in module wrdrd.tools.domain), 331
 - dig_mx() (in module wrdrd.tools.domain), 332
 - dig_ns() (in module wrdrd.tools.domain), 332
 - dig_spf() (in module wrdrd.tools.domain), 332
 - dig_txt() (in module wrdrd.tools.domain), 332
 - Digital Stand Up Meeting, 41
 - Digital Will, 244
 - Disc Drives, 65
 - Discovery and Launch, 305
 - Dissertation, 270
 - Distributed Algorithms, 172
 - Distributed Computing Problems, 172
 - Distributed Computing Protocols, 174
 - Distributed Databases, 168
 - Distributed Hash Table, 173
 - Distributed Revision Control, 53
 - DJ, 289
 - DL, 141
 - DLNA, 305
 - DNS, **22**

DNS Configuration, 30
Doctorate, 270
DOM, 24
Domain Name, 22
domain_tools() (in module wrdrd.tools.domain), 332
Doubling time, 239
dram, 128
DreamStudio, 303
Drizzle, 163
DSpace, 228
Dual Enrollment, 267
DVCS, 53
DVI, 60

E

EARL, 204
Economics, 236
EdTech, 254
Education Technology, 254
Educational Testing, 263
EdX, 258
Effort Estimation, 43
ElasticSearch, 181
Emotional Intelligence, 313
Employee Stock Option, 249
Entailment, 142
Enterprise Service Bus, 176
Entrepreneurship, 230
Equity, 248
eSATA, 63
eSATAp, 64
ESB, 176
Escape Sequences, 135
Estate planning, 243
Estate tax, 243
Ethernet, 67
Ethernet Crossover Cable, 68
ETL, 109
Everyone Codes, 41
Existential quantification, 140
Exit strategy, 232
Experience Design (XD), 311
ExpressCard, 58
ext2, 156
ext3, 156
ext4, 156
Extended Knowledge Organization System, 205
Extreme Programming, 43

F

FAT, 156
FAT12, 156
FAT16, 156
FAT32, 156

Favicon, 12
Fedora Commons, 228
FFmpeg, 280
Fiber, 72
Fiber to the x, 72
Fibre Channel, 72
File Locking, 147
File Structures, 146
File Systems, 154
Filesystems, 154
Financial Statement, 248
First-order Logic, 140
FLAC, 290
Fluid Ounce, 128
FLV, 281
FOAF, 205
FOL, 140
Folders and Labels, 225
Fonts, 12
Foot, 127
Fountain.io, 283
Friend of a Friend, 205
FUSE, 158
Fuzzy Logic, 139

G

Gallon, 129
Gaming Consoles, 306
Gateway, 73
GED, 266
Geiss, 294
GIF, 274
Gift tax, 239
Gigabit, 66
Gigabit Ethernet, 69
GIMP, 274
Git File Structures, 146
Git Object, 146
Git Packfile, 147
Git Reference, 147
GitHub Pages, 256
Given-When-Then, 45
GlusterFS, 159
GMAT, 269
GNUCash, 92
Goal, 49, 233
Golden Rule, 313
Google Apps, 91
Google Cardboard, 278
Google Cast SDK, 305
Google Checkout, 93
Google Cloud, 88
Google Glass, 278
Google Wallet, 93

GPT, 155
 GPU, 59
 Gram, 121, 130
 Graph Databases, 165
 Graphic Design, 11, 273
 Graphs, 149
 GraphX, 172
 GRE, 268
 Gremlin, 167
 GSM, 72
 gzip, 152

H

H.264, 281
 Hadoop, 170
 Hadoop distributed filesystem, 160
 HandBrake, 281
 Handling Money, 92
 Hard drives, 65
 Hash Functions, 153
 Haystack, 181
 HBase, 170
 HDFS, 160
 HDMI, 61
 Health Insurance, 242
 Health Savings Account, 242
 Healthcare, 242
 Heroku, 87
 Hertz, 123
 HFS+, 157
 High School Diploma, 266
 Hive, 170
 HLS, 302
 Hoare Logic, 140
 Home A/V, 304
 Honorary Degree, 271
 Hookpad, 287
 Hour, 122
 HTC Vive, 278
 HTML, 196
 HTML4, 196
 HTML5, 197
 HTTP, 33, 193
 HTTP in RDF, 194
 HTTP Live Streaming, 302
 HTTP Strict Transport Security, 33, 194
 HTTP STS, 33, 194
 HTTP/2, 196
 HTTP2, 196
 HTTPS, 33, 194
 Hub, 74
 Human-Computer Interaction (HCI), 311
 Hydra, 229
 Hydrogen, 287

Hz, 123

I

IaaS, 87
 IBDP, 267
 ICO, 274
 IDE (drive interface), 64
 IEC, 192
 IETF, 192
 IFTT, 16
 Imagine Success, 233
 Imperial Distance Units, 127
 Imperial Mass Units, 130
 Imperial Units, 126
 Imperial Volume Units, 127
 Inch, 127
 Income tax, 239
 Individual Retirement Account, 241
 Industry Units, 131
 Inference, 142
 Information, 77
 Information Design, 312
 Information Retrieval, 183
 Information Security, 90
 Information System, 77
 Information Systems, 75
 Information Systems Glossary, 77
 Information Technology, 77
 Information Theory, 104
 Infrastructure, 77
 Infrastructure-as-a-Service, 87
 Inheritance tax, 244
 Inkscape, 275
 Integrity, 90
 Interest, 238
 International Atomic Time, 183
 Intestacy, 244
 Investing, 245
 Investing Strategies, 250
 Investing Styles, 250

IP, 20

IP Address, 20
 IPv4, 21
 IPv4 Address, 21
 IPv6, 21
 IPv6 Address, 21
 IRA, 241
 IS, 75
 ISA, 57
 Islandora, 229
 ISO, 193
 ISO8601, 189
 iso8601, 189
 ISO9660, 157

J

JACK, 290
Joule, 124
JPEG, 275
JSON, 198
JSON-LD, 203
JSON-RPC, 177
JSON-WSP, 179
JSPF, 298
Jupyter and edX, 259
Jupyter and Learning, 259
Jupyter and Reproducibility, 260
Jupyter and TDD, 261
JupyterHub Servers, 262

K

Kanban, 43
Keg, 129
Kelvin, 124
Kerbal Space Program, 301
Kilogram, 121
Knowledge Engineering, 131
Kodi, 308

L

LADSPA, 290
Language API, 78
LaTeX, 104
Layout Graphics, 14
LDP, 208
Letsencrypt, 34
Liability, 248
libvisual, 294
License (Docs), 323
Lightworks, 281
LilyPond, 283
Linear Algebra, 105
Linked Curricula Graphs, 262
Linked Data, 190
Linked Data Platform, 208
Linked Open Vocabularies, 217
Linked Reproducibility, 102
LinkedReproducibility, 218
Lists, 148
Liter, 120
Litre, 120
LiVES, 282
Location Based Services, 17
LODCloud, 218
Logic, 137, 138
Logical Volume Manager, 155
Logo, 12
Logo (graphical), 12
Logotype, 12

Long Now Dates, 183
Long Term Care, 242
LOV, 217
LSAT, 269
LTE, 72
Lucene, 182
LVM, 155

M

M3U, 296
MAC address, 20
Machine Learning, 107
Magnet URI, 23
Mailing List Collaboration, 319
main() (in module wrdrd.tools.domain), 332
Maintainability, 80
Many-valued Logic, 138
MapReduce, 173
Marketing, 9
Marketing Mix, 9
Mass noun, 117
Master's Degree, 270
Math, 102
Math Courses, 103
Mathematical Notation, 103
Mathematics, 102
MathJax, 104
MathML, 104
Matplotlib, 111
Matrices, 148
Matrix, 148
Mayavi, 112
MBR, 155
MCAT, 269
MD5, 153
Media, 273
Media Tools, 303
Mediated Reality, 277
Mediated Reality Mediums, 278
Memory, 57
Mendeley, 227
Mesh Wireless, 70
Message Passing, 175
Meta-analysis, 102
Meter, 120
Metre, 120
Metric Prefixes, 119
Metric System, 118
Metrics, 10
Micro HDMI, 61
Microsecond, 121
Microsoft HoloLens, 279
MIDI, 287
Mile, 127

Milkdrop, 294
 MilkDrop Presets, 295
 Millisecond, 121
 Mini HDMI, 61
 Minute, 121
 Mission, 48
 Mission Statement, 233
 Mixed Reality, 277
 Mixxx, 291
 Mobile Broadband, 71
 Modem, **73**
 Modus ponens, 139
 Mole, 125
 Monitor, 55
 month, 122
 MP3, 291
 MPEG, 282
 MPEG-DASH, 302
 MPI, 176
 mSATA, 63
 Multiheaded Display, 59
 MuseScore, 284
 Musical Composition, 285
 Musical Notation, 284
 MusicXML, 284
 MySQL, 164
 MythTV, 308

N

N-Triples, 201
 N3, 201
 Named Data Networking, 300
 nbgrader, 261
 Neo4j, 167
 Network File Systems, 158
 Network Filesystems, 158
 Network Interface, **73**
 Network interfaces, 65
 Network Route, **74**
 Network Time Protocol, 190
 NetworkX, 150
 NFS, 160
 nginx-rtmp-module, 302
 NIC, 66, **74**
 NLES, 279
 Non-blocking algorithm, 173
 Nonparametric Statistics, 106
 Northbridge, 55
 NoSQL, 165
 NoSQL Databases, 165
 Notation3, 201
 nslookup() (in module wrdrd.tools.domain), 332
 NTFS, 157
 NTP, 190

NURBS, 276
 Nutch, 182

O

OA, 206
 OAI-PMH, 229
 Object Relational Mapping, 162
 Objective, 49, 233
 Occupational Outlook Handbook, 238
 Oculus Gear VR, 278
 Oculus Rift, 278
 Oculus VR, 278
 OMG, 193
 Online Courses, 258
 Open Science, 101
 Open Source, 3
 Open Source Licensing, 5
 Open Web Standards, **24**
 OpenAnnotation, 206, 296
 OpenBadges, 263
 OpenBadges Backpack, 263
 OpenGL, 276
 OpenUp, 43
 Operating System, **78**
 Operating System API, **78**
 Option, 249
 ORM, 162
 Ounce, 130
 OWASP, 32
 OWL, 209

P

PaaS, 84
 pandas, 251
 pandas-datareader, 252
 Parallel ATA, 64
 Parametric Statistics, 106
 Parquet, 171
 PATA, 64
 Paxos, 173
 Payments, 93
 PayPal, 93
 PCI, 58
 PCI-e, 58
 PCMCIA, 58
 Pension, 241
 Personal Debt, 238
 Persistent Storage, 64
 Personal Budget, 237
 Personal Development, 313
 Personal Expenses, 238
 Personal Finance, 236
 Personal Income, 238
 Photography, 279

Photography Checklist, 13
PhysX, 60
Pint, 129
Plan for Failure, 232
Plan for Success, 232
Planning, 232
Planning Poker, 43
Platform-as-a-Service, 84
Plex, 308
Plotly, 112
PLS, 296
PNG, 275
Poetry, 283
Popcorn Maker, 296
Post-production, 279
PostgreSQL, 164
Pound, 130
PPC, 56
Predicate Logic, 139
prefix.cc, 217
Presto, 171
Probabilistic Logic, 139
Processor, 56
Profitability, 233
Project, 49, 233
Project Euler, 103
Project Management, 40
Project Site, 320
ProjectM, 295
Property tax, 239
Propositional Calculus, 139
Propositional Logic, 139
Protocol Buffers, 178
PROV, 210
Publishing, 255
PulseAudio, 291
pyql, 253
PyQtGraph, 113

Q

QB, 204
qgrid, 113, 254
Quandl, 252
quandl-python, 252
Quantitative analyst, 251
Quantlib, 252
Quantopian, 253
Quantopian Open, 253
Quart, 129
QUDT, 212
QUDT Quantities, 212
QUDT Units, 213
QuickBookx, 92
Quicken, 92

R

Rackspace Cloud, 89
Raft, 174
RAID, 154
RAM, 57
RCS, 52
RDF, 199
RDF and Units, 117
RDF Data Cube, 204
RDF Databases, 168
RDF Interfaces, 199
RDF Schema, 203
RDF Triplestores, 168
RDF/XML, 201
RDFa, 202
RDFa Core Context, 202
RDFHDT, 216
RDFJS, 216
RDFLib, 217
RDFS, 203
ReadTheDocs, 257
Reasoning, 137, 142
Refactoring, 52
Regression Analysis, 106
Relation Algebra, 162
Relational Algebra, 162
Relational Databases, 162
Reliability, 91
Representational State Transfer, 179
Reproducibility, 102
Requirements Management, 44
Requirements Modeling, 44
Requirements Traceability, 46
Research, 224
Research Tools, 227
Resolume, 295
Responsible Disclosure, 31
Resource-Oriented Architecture, 179
REST, 179
RESTful API, 79
Retirement, 240
Revenue, 247
Revision Control, 52
Rhinoceros 3D, 276
ROA, 179
ROI, 8
Roku, 309
Rosalind, 103
Rosegarden, 285
Roth IRA, 242
Router, 75
RS-232, 65
RTMP, 195
Rubric, 38

Rule of 72, 239
RUP, 43

S

S3, 160
SaaS, 83
SAFE, 249
Sales, 10
Sales Information Systems, 94
SANS/CIS Consensus Audit Guidelines:, 31
SAT, 267
SATA, 63
SBA, 231
Scenarios, 91
Schema.org, 207
Schema.org RDF, 207
Schema.org TopBraid RDF, 208
Science, 101
Scientific Method, 102
Scorewriting, 283
Screen Captures, 26
Screenwriting, 282
Scrum, 43
SCSI, 64
Seaborn, 111
Search Engine Indexing, 181
Search engines, 226
Second, 121
Semantic Web, 191
Semantic Web Schema Resources, 217
Semantic Web Standards, 191
Semantic Web Tools, 215
Semantically Interlinked Online Communities, 206
Serial ATA, 63
Set Theory, 138
Seven Layer Model of Collaboration, 314
SHA, 153
SHACL, 206
SI Amount Units, 124
SI Base Units, 119
SI Bit, 125
SI Byte, 125
SI Data Units, 125
SI Distance Units, 120
SI Electric Current Units, 123
SI Frequency Units, 123
SI Gram, 121
SI Kilogram, 121
SI Litre, 120
SI Luminous Intensity Units, 125
SI Mass Units, 120
SI Meter, 120
SI Minute, 121
SI Power Units, 124
SI Prefixes, 119
SI Second, 121
SI Temperature Units, 124
SI Time Units, 121
SI Units, 118
SI Volume Units, 120
SigmaJS, 113
Simple Knowledge Organization System, 205
Simple Service Discovery Protocol, 305
SIOC, 206
Six Patterns of Collaboration, 314
SketchUp, 276
SKOS, 205
Small Business Consulting, 6
SMART, 49
SMB, 161
Smoothies, 100
SOA, 178
Social Media, 14
Social Media Images, 12
Social Security, 241
Software Development, 35
Software Development Effort Estimation, 43
Software-as-a-Service, 83
SOHO, 73
Solr, 182
Solving, 7
Songwriting, 283
Sound Production, 288
Spark, 171
SPARQL, 208
SQL, 163
SQL Databases, 162
SQLite, 164
Square, 93
SSD, 65
SSDP, 305
SSHFS, 158
Stand Up Meeting, 41
Standalone Media Devices, 306
Start at the end, 233
Statistical Inference, 106
Statistics, 105
STEAM, 255
Steam, 301
STEM, 255
STK, 288
Stock Option, 249
Strategy (Business), 8
Stream Annotation, 296
StructuredPremises, 221
StudyGraph, 219
SVG, 275
Swift, 161

Switch, 74
Symbols, 133
System, 77
System Bus, 57
Systematic Review, 102

T

T568A, 67
T568B, 67
Tablature, 285
TablEdit, 285
tablespoon, 128
tar, 152
Taxes, 239
TCP, 20
TDD, 44
Team Building, 312
Team Site, 320
teaspoon, 127
Technical Debt, 51
Ten Simple Rules, 95
Test Automation, 53
Test Driven Development, 44
Text Templates, 316
The Four Ps, 10
The same page, 316
Thesis, 270
ThinkLets, 315
Three C's, 46
Three Questions, 41
Three-valued Logic, 139
Three.js, 113
Thrift, 178
TIA, 66
Time Standards, 183
Time Zones, 186
Ton, 130
Tool Support for Units, 117
Topological Sorting, 151
Torrent file structure, 147
Transaction Processing, 92
Trees, 151
TriG, 202
Triplestores, 168
TriX, 201
Trust (financial), 243
Tube Sites, 301
Turtle, 202
Twitter, 14
Typeface, 12
Types of Business Entity, 234
Typography, 12

U

Ubuntu Studio, 304
Udacity, 259
UI, 312
UI / UX Design, 310
Unicode, 136
Unified Process, 43
Units, 115
Universal quantification, 140
Unix Time, 184
UP, 43
UPnP, 306
URI, 23, 192
URIs for Units, 217
URL, 22, 192
URN, 23, 192
US customary units, 130
US Daylight Saving Time, 188
US Time Zones, 187
USA Entrepreneurship, 234
USA Higher Education Testing, 267
USA K-12 Curriculum Standards, 266
USA State Testing Standards, 267
USA Testing Standards, 265
Usability, 311
USB, 62
USB Hub, 63
USB Keyboard, 63
USB Mouse, 63
USB Type-C, 63
Use Case, 46
User Experience (UX), 312
User Interface (UI), 312
User Story, 45
UTC, 186
UTF-8, 137
UUID, 23

V

Value Investing, 250
VCS, 52
Vega, 112
Version Control, 52
VGA, 60
Video Card, 58
Video Chat, 303
Video Editing, 279
Video Production, 279
Videography, 279
Vincent, 112
Virtual Reality, 277
Virtual Studio Technology, 288
Virtuoso, 164
Virtuoso Universal Server, 164

VisPy, 112
 Visual Composition, 273
 Visualizing Data Science, 110
 VJ, 293
 VLC, 304
 Voice Searches to Test, 18
 Volt, 124
 VST, 288
 vvvs, 295

W

W3C, 193
 WAMP, 180
 Waterfall, 40
 Watt, 124
 WAV, 293
 Web Analytics, 10
 Web API, **78**
 Web Application, **19**
 Web Browser, **23**
 Web Content, 24
 Web Design, 24
 Web Development, 18
 Web Development Checklist, 27
 Web Development Requirements, 34
 Web Distribution, 298
 Web Hosting, 30
 Web Hosting Service, **20**
 Web Layout, 25
 Web Layout Graphics, 14
 Web Names, 192
 Web Ontology Language, 209
 Web Page, **19**
 Web Production, 296
 Web Seeding, **300**
 Web Service, **79**
 Web site, **19**
 Web Standard, **24**
 Web Standards, 192
 WebDAV, 161
 WebGL, 277
 Webhook, **79**
 WebRTC, 196
 WebSec, 31
 WebSec Forums, 32
 Website, **19**
 WebSocket, 195
 webtorrent, 300
 Week, 122
 whois() (in module wrdrd.tools.domain), 333
 Whoosh, 183
 Wi-Fi Direct, 70
 WiFi, **75**
 Wikidata, 215

Will and Testament, 243
 Windows Holographic, 279
 Wireless, 69
 Wireless mesh network, 70
 Wordmark, **12**
 Work Breakdown Structure, 48
 WRD R&D Documentation, 1
 WRDRD Docs, 1
 wrdrd.tools (module), 330
 wrdrd.tools.domain (module), 330
 WS- Web Services, 178
 WSDL, 179

X

x86, 56
 x86_64, 57
 Xapian, 183
 XHTML, 196
 XKOS, 205
 XML, 197
 XML-RPC, 176
 XP, 43
 XSD, 197
 XSPF, 297

Y

Yard, 127
 year, 123
 Year Zero, 184

Z

Zapier, 16
 Zeroconf, 306
 ZIP, 153
 Zipline, 254
 Zotero, 227