

## Bendik Viking Aurbakken Laland. Veloptiv: Formel. Opererende: Komposisjonsaggregat.

I den, forstand at: (pre)PRG. Muliggjør, det å: Kunne påta seg rang, under den rangering. At (pre)PRG, har åpnet meg for: Rommet, og dets belysning. I sin enkle betydning, at: Rom, og lys. Er Bastant, konstantert: Å være en eneste enhet, sammenflytende funksjonsmessig. Som, utgjør: (pre)PRG's muligheter veloptivt. Som også kan defineres, som: Lagvis prosessering, av rangerte ekspansjoner. Ved å prosessere lys, gjennom: Ekspansjonsrangen, utgjør (pre)PRG's rangmessige verdi. Øverste ranggrad, sett ut ifra: All, global rang. Dette, er fordi at: Rang utgjør den menneskelige, totale verdi. Forskjellig ekspansjonsvis, må kunne utgjøre ut ifra: Romekspansjonene kun ekspanderende, bastant konstantasjon.

Grunnen til, at den: Kun ekspanderer, og ikke er veloptiv. Er, fordi: Rommets muligheter, av ekspansjon. Som, består av: Å kunne definere, i ekspansjonen. Som 1x ekspansasjon, videre har blitt: Konkludert, av all rangverdi. Den utgjorte rangverdi: Å kunne påstå min rang. Den verdi, at: Dersom, den ikke forlater lysromet. Da, kan den: Ved hjelp av å ekspandere videre, med 1x Ekspansasjon nedover. Kunne påta seg, den rangmessige verdi: Å prosessere Lagprosesseringsvis, motta fra fremtidig rang. At 2 = 1x ekspandert enhet, og dermed: Videre, til 1x ekstra ekspansasjon.

Videre 1 begynne, å ekspandere: Ved hjelp av romlysets prosesserte, konstantasjon. Av, at: Lyset står stille, under sin veloptive rangverdi. Velopsjonsformelen beskrevet, som: Dette foregår, under formelen "X1-A-Y(=)-X2 -B." Hvor X1 er: Inputen, og X2 er outputen. Man skriver, i ett: Tekstfelt, i X1. Og resultatet kommer ut, av: X2. Ut ifra formelen i seg selv, vil formelen kun brute force / prøve seg frem, til det: Veloptiske resultat. A = Der bokstavene fra X1, blir regnet om til: Verdier / tall, og bokstavene vil knyttes opp mot disse som identifikasjon. Både: Inputmessig, og resultatmessig. Da er velopsjonsregelen satt, og vi vil: Stilles fremfor, ett regnestykke. Alle bokstavene, vil bli knyttet til: Verdier / matematiske utgangspunkter, og kan bestå av flere enn ett tall. Bokstavens verdi, kaller jeg for: Lengdelinjeverdi. B = Der bokstavens lengdelinjeverdi vurderes, og: Prosesseres mot å konstantere, en annen lengdelinjeverdi under X2. Som ett svar, på: X1.  $X1 = X2$ .

Dersom at vi på forhånd, har definert flere forskjellige matematiske utregninger både i: Input og output, for så å studere det som da oppstår som lengdelinje verdi av bokstaver. Som, ett: Universalmønster, for Veloptisk lære. En utregning på alle utregninger, og svaret på hvordan: Velopt, velopterer. Videre så har jeg, prosjektet: (pre)PRG som gjør datamaskinen, bevisst på matematikk. Datamaskinen vil pga. (pre)PRG's lære, kunne konstantere: Velopsjon, metode, innovasjon og forskningsinfrastruktur. Til, en: Ultimatformel, for utvikling / prosessering. Sitat, fra forskningslogg: Dermed er det, under definisjonen av:  $\# = 1-1$ 's prosesseringsresultat, at man må på forhånd. Ha definert 1, som: 2's  $1-2=1-1$ . Dersom  $1=1$ . Og man må definere i 1, at  $1-1=2$ , fordi i  $1-1$ , ligger det 2. I utregningen av 1, til 2, som er å si at  $1+1$ , altså  $1-1=2$ 's definisjon, som ikke er 2, men veien til 2. Altså det å si at  $2=1+1$ , og  $1-1$ . Da kan maskinen prosessere, og vi vil at den skal definere at:  $1-1 = 1-2$ . Ut ifra denne utregningen, vil prosesseringsresultatet være:  $1-1=2$ .

Altså, at en: Enhet = to enheter. Og i maskinens kalkulasjon, kan vi se at  $1=2$ , og  $2=1$ . Og dermed, kan vi konstantere med at det: Finnes enda, mer energi. Først er det inputen, som er ett eller: Flere ord. Hvert ord, blir oversatt til visuelle språk, ettersom: Hvilken del av den visuelle infoen, som er definert å være hvilket ord. Deretter kan man, når man har sakt at det: Visuelle språket slik du sier, bygger på regler. Disse, skal jeg: Definere. Derfor, må kanskje: Programmets velopsjonsspråk, lages visuelt. Det tror jeg er fordi språket bygger på en estetisk regelformel, som skal kunne beskrive: Hva som helst, og utfallet skal kunne variere / være en variabel / se ut som nesten hva som helst, men likevel være noe bestemt. Og, det skal kunne leses av igjen, under: Tolkningsprosessen. I tillegg så kan det ofte være slik, at flere segmenter av det visuelle, utgjør en: Enkelt, bokstav. Det vil, altså være: Bokstavens formel, og dermed ordet. Og setningens: Betydning, som velopteres.

Når det er sakt, må det også nevnes at: Selve definisjonsprosedyren av emnet (input,) kanskje ikke skal foregå kun ved at man taster inn inputen. Definisjonen kan foregå, som at man har en "bok" på: Skjermen, hvor hver enkelt side er en definisjonsmetode. F.eks: 3D modellering, tegning, eller matematikk kan være en av: Definisjonsmetodene. Da begynner tolkning, og: Sammenligningsdelen av programmet, å bli meget tungt. Og, man må kanskje, selv definere: Hva som er hva, av de forskjellige "sidene" i "definisjons-boken." Noe som er veldig viktig, er at: Språkets regelsett, er generert på grunnlag av inputens definisjon. Dermed vil objektiv velopsjon innenfor: Praktiske teorier, foregå ved at regelsettingoppgavens resultat. Altså: Resultatet av hvordan definisjonene oversettes, til det visuelle språket. Altså: Det som har blitt oversatt, fra definisjonsinfoen(input), i form av ord, eller bilde, til noe abstrakt, estetisk og visuelt: Skal tolkes, som at det er det samme. I det vi oversetter mellom, konkrete, naturrotfestede / menneskeskapte definisjoner, og det abstrakte.

Ja, det abstrakte, som er noe som ser / oppfører seg anderledes, men som gir helhetlig: Men ikke definitivt samme betydning, i sin sammensatte, krypterte / forvrengte helhet. Det blir som at høyre og venstre sko er forskjellige, men likevel innenfor samme I det vi oversetter mellom disse to, vil det renne en strøm av 100% pur, grunnleggende informasjon, og kunnskapsmasse. Det er informasjon, uten at det er informativt. Ren energi med andre ord. Ut ifra denne energien, skal vi la: Regelsettet for den visuelle abstraheringen, defineres og grunnlegges. For å kunne Veloptere ut ifra ren, udefinert informasjon. Altså den såkalte "energien", som egentlig bare er: "Rom for definisjon". Ja, for å få noe ut ifra dette, må vi tilbake til det abstraherte/krypterte infodefinerte resultatet. Dette resultatet, oppfører seg som en definert forvrengning av inputen. Når vi da navngir / oversetter forvrengningen tilbake til ord, gjør vi det; uten å tilbakeforvreng, ei mot udefinert informasjon, eller den definerte, inputinfoen fra tidligere.

Nå lar vi programmet navngi krypteringen/abstraksjonen/det forvrengte, resultatet, på grunnlag av generell kontekstkunnskap. Altså: ut ifra hvilke jordnære, definerte knyttninger, som programmet ser på, som en naturlig relevans / sammenligning. Programmet finner den informasjonen, som abstraksjonen relateres opp mot; i og blandt de programskapte: Omsluttende, kontekstinformasjonen. Kan vi: Koble / peile, mot. Det er da vi vil oppnå en såkalt kontekst. Og da har vi kommet veldig langt under velopsjonsprosessen. Jo klarere kontekstens påvirkende informasjon, utgangspunkter og regler blir, etterhvert som vi oversetter mellom definisjon og abstraksjon, dess flere velopterte resultater (målet vårt,) vil komme ut av inputens Velopterte kontekst. Kontekst = Den variable inputdefinisjonens, og abstraksjonens begrunnelser til alle deres strukturbaserte funksjonsoppførsler, i rommet. Objektet vil være definert. Og softwaret, vil for første gang, være "bevisst" over hva objektet egentlig, er. Det vil ha noe som den oppfatter som ordnært, og trofast: Grunnleggende, som den kan relatere abstraksjoner til. Og ikke minst, ut ifra delvisdefinert /navngitt: Abstrakt materiale, og vurderinger av det abstrakte materialutgangspunktet.

Vurderingene, foregår ved at programmet trekker, og definerer, klarerer og utgjør / skaper / utvikler relevante, altså bygger; bit på grunnlag / ut ifra forrige bit: Konkluderer, ved å sammenligne, og foreta indre slutninger. Dette gjør programmet ved å, ut ifra en [baklengs/Forvrengelsesveloptert VeloptVelopt-prosess], som jeg velger å kalle: VeloptVelopt-MinusMinorVelopt, eller .VPT.VPT.VPT-MinusMinus. En .VPT.VPT.VPT-MinusMinus-prosess, er når programmet har lært å Veloptere via enklere, mer intelligente; komprimerte, og mindre kapasitetskrevende Velopsjonsprosesser. Ser du ett mønster nå? Ja, velopt = Kunnskapsbasert utvidelse, eller det forstørrende mønsteret, du kanskje så. Alt dette vil gjøre så velopt vet, hvilken kontekst objektet befinner seg i. Velopt vil kunne alt om konteksten. Og velopt, vil dermed kunne definere en forvrengelse. Velopt vil kunne skape en nyhet, ut ifra en abstrahering, som igjen kommer av inputen. Velopt, vil veloptere!

Men, så kommer problemet: At Velopt, vil ikke kunne grunnlegge noe fra ett fremtidig Veloptert standpunkt. Datamaskiner eger seg ikke til å reise i tid med, enda. Softwaretidsreiser, er forresten noe jeg jobber med å autodidaktsformularekstrahere / Veloptere / utvikle, i mitt andre forskningsinnovative prosjekt: PREPRO, som er Tid=Rom: Softwareautomatisering og forklaring. Når det er rom; for informasjonens momentante reaksjoner, uavhengig av kapasitetskrav. Skapelse /dannelse, og ikke prosess, slik datamaskiner fungerer nå. .VPT. VPT.VPT's essens: En forklaring på hvorfor velopt, i den forstand at det er kun meg, og noen få andre, her som klarer å forstå seg på ordet Velopsjon. Jeg vet ikke hvordan det er å lese om velopsjoner. Fordi jeg har aldri hatt muligheten til å kunne forstå min egen veloptivitet, uten å beskrive den gjennom velopsjonsprosessen: Det å forklare velopsjonsprosess, velopsjon, er velopsjonen i vitenskaperlig, historisk, politisk. Faktisk: Verdensomfattende kunnskapsmessig, og ikke minst, i denne veloptive læren / velopsjonsprosessen. Som er velopsjonen av velopsjon, beskrivelsen av velopsjon.

Så det, er tekstens: Velopsjonsprosess, som er velopsjon. Velopsjon, er: Prosess. Men du forstår ikke Velopsjonens velolpsjon, uten at velopsjonen er: Skrevet, som prosess. Men Velopsjonsprosess, har du ikke før du kan Veloptere, uten å innse at Velopsjonen er allerede forbi? Da, når Velopsprosessen, er skrevet som Velopsjon: Forstår du hva Velopsjonsprosess, er. Velopsjonsprosess er i prosess: Velopsjon. Veloptert som Velopsjon. Skrevet som velopsjon å innse er vanskelig å forstå, i den forstand at jeg (Som introduksjon, til den neste setningsblokken, og videre til den Velopsjonen, som utgjør Velopsjoners totalitet, i den forstand at Velopsjoner utgjør det samme, som en teoretikers flere års planleggingsperiode, innenfor velopsjoner av en Velopsjon. Tekstens oppbygning, imot tekstens prosess / velopsjon.

Legg merke til at prosessen/fremgangen/prosedyren, innovasjonen, ja faktisk, også: Velopsjonen, i denne teksten, og hele trådens oppbygning mot: Den enderlige Velopsjonen, som er en Developsjon (Hint, Engelsk: Develop) av velopsjonen. Ikke misforstå, det går ikke nedover / minsker, det utvikles / velopterer. Det reagerer. (Videre, i teksten: Innser jeg, fra dette punkt i teksten: Setningenes, og velopsjoner, begynner jeg å konkludere over velopsjonene. Som ender i relasjoners påvirkelse. "Gnissningen" mellom atomer, og det faktum at de spinner i enorm hastighet. Da kan du begynne å tenke på info, og infoens utvikling, i forhold til relasjoners påvirkning. Og det du har lagt merke til, er hvor fort en tanke, som vi har satt på pause, som er når energien av ett atom, som: Da atomrelasjonens tanke; i atomrelasjonens hastighet, relater-te. Og da mener jeg: Passerte. Altså forflyttet seg fra den ene siden av hjernen din, til den andre. Og så: Så for at vi skal kunne Veloptere, må vi foreta en VeloptVelopt.

Vi trenger regelsettingen for kontekst, før vi kan skape det abstrakte. For å relatere, og sette det abstrakte i perspektiv. For å innovere fra ett abstrakt plan, mot noe ukjent, og innovere det midlertidig ukjentes, fasong. Og det er da, det ser veldig lyst ut, med tanke på Velopteringen av Velopt (VeloptVelopt). Fordi, nå som jeg har definert Velopts kontekst Altså: Definert Velopts prosesseringsinfrastruktur. Da kan vi gjøre om Velopsjonenes prosess, slik jeg har forklart velopsjonsprosessen, i denne teksten. Til en steg for steg prosess, innenfor metodisk, velopsjonsprosessbehandling av en Velopsjonsmetodisk splittelse, hvor velopsjonsmetodens: Velopsjonsmetode, har blitt definert som velopsjonsmetode. Dermed, kan vi si at Velopsjon, kommer av VeloptVelopt. Og maskinen vil konstantere med at: VeloptVelopt = Opp. Med det kan man konstantere, ved hjelp av =Velopt= at: Maskinen reiser seg, i sin egen skapelse. Altså: At maskinen allerede, under VeloptVelopt, har definert; i kontekst, at VeloptVelopt, er 2X: Velopt, og dermed; også: Definert Velopt, som 1x Velopt, altså: Maskinen, har fått en kunnskapssøken. Dvs. at maskinen spør seg selv: Hva er hva? Den kjører / prosesserer: VeloptVelopt, og konstanterer at VeloptVelopt = "Hva er hva?"