

Verklarende terminologielijst voor de Software Engineering

A

abnormal end: abend *abnormaal einde*

De beëindiging van een proces voordat dat proces geheel is afgewerkt.

abort (to) *afbreken*

Een proces afbreken voordat het volledig is afgewerkt.

absolute address *absoluut adres*

Een adres dat permanent aan een eenheid of geheugenplaats is toegewezen en dat de eenheid of de geheugenplaats identificeert zonder dat daarvoor een vertaling of berekening nodig is.

absolute assembler *absoluut assembleerprogramma*

Een assembleerprogramma dat uitsluitend absolute code genereert.

absolute code *absolute code*

Code waarin alle adressen absoluut zijn.

absolute coding *absolute codering*

Een coderingsmethode waarin gebruik wordt gemaakt van instructies die absolute adressen bevatten.

absolute expression *absolute uitdrukking*

Een uitdrukking voorafgaande aan het moment waarop het assembleren van een programma plaatsvindt. Dit moment wordt niet beïnvloed door het verplaatsen van het programma. Een absolute uitdrukking kan een absoluut adres in een assembleertaal weergeven.

absolute instruction *absolute instructie*

1. Een instructie in de definitieve uitvoerbare vorm.
2. Een computerinstructie waarin alle adressen uit absolute adressen bestaan.

absolute loader *absoluut laadprogramma*

Een programma dat een ander programma in het hoofdgeheugen kan plaatsen. Dit begint bij het initiële adres dat aan de code door het assembleerprogramma of de compiler is toegewezen en dat de adressen in de code niet verder wijzigt of aanpast.

absolute load module *absolute laadmodule*

Een combinatie van werkmodules die op een gespecificeerd adres in het werkgeheugen wordt uitgevoerd.

absolute machine code *absolute machinecode*

Machinecode die steeds in vaste geheugenplaatsen moet worden geladen en niet mag worden verplaatst. Dit in tegenstelling tot verplaatsbare machinecode.

absolute pathname *absolute padnaam*

Een padnaam die de informatie bevat over de wijze waarop een bestand kan worden gevonden. Het zoeken naar het bestand vangt aan bij de hoofddirectory waarna de boom van de directory wordt afgezocht.

abstract class *abstracte klasse*

1. Een klasse die geen directe exemplaren (instantiaties) bevat maar waarvan de afgeleide klassen wel directe exemplaren bezitten.
2. Een klasse die één of meer abstracte methoden bevat en als gevolg daarvan nimmer kan worden geïnstantieerd. Abstracte klassen zijn zodanig gedefinieerd dat andere klassen deze uit kunnen breiden en ze concreet kunnen maken door de abstracte methoden te implementeren.
3. In de object-georiënteerde programmering is dit een klasse die uitsluitend als een ouder is geconstrueerd en waarvan deelklassen kunnen worden afgeleid maar die zelf niet geschikt is om te worden geïnstantieerd. Deze klasse wordt vaak gebruikt om onvolledige verzamelingen met kenmerkende eigenschappen te abstraheren. Deze kunnen dan onderling worden gedeeld door een groep bloedverwante deelklassen waaraan verschillende variaties van de ontbrekende delen kunnen worden toegevoegd.

abstract data type: ADT *abstract gegevenstype*

1. Een klasse gegevensstructuren die met behulp van een verzameling bewerkingen in plaats van hun fysieke representaties worden gedefinieerd. Een voorbeeld hiervan is een klasse in de object-georiënteerde programmering.
2. Een gegevenstype waarvan de eigenschappen van het gegeven en de daarop uit te voeren bewerkingen zijn gespecificeerd, zonder dat het daarbij van belang is hoe het gegeven wordt gerepresenteerd of hoe de bewerkingen zullen worden geïmplementeerd.
3. De term heeft betrekking op een klasse van objecten waarvan het gedrag door een serie waarden en een reeks bewerkingen is gedefinieerd.

abstraction *abstractie*

1. Een beschouwing van een probleem waarin essentiële informatie, die relevant is voor een bepaald doel, wordt geëxtraheerd en de rest van de informatie wordt genegeerd.
2. Het proces met behulp waarvan een abstractie wordt gevormd.
3. Een optiek (gezichtspunt of invalshoek) van een object die zich richt op de informatie die voor een bepaald doel van belang is en die de rest van de informatie buiten beschouwing laat.
4. Het proces waarin de optiek als genoemd onder punt 3 wordt geformuleerd.

abstract machine *abstracte machine*

1. Een representatie van de karakteristieke eigenschappen van een proces of een machine.
2. Een module die de invoer van processen verzorgt en dit doet alsof het een machine is.

abstract method *abstracte methode*

Een methode die geen implementatie kent.

abstract syntax notation no. 1: ASN.1 *abstracte syntax notatie nr.*

1

Een internationale standaard voor de beschrijving van gegevens. Deze standaard kan de uitwisseling van gegevens tussen applicatieprogramma's vereenvoudigen.

abstract-type and schema-definition language: ASDL *abstract type en schemadefinitietaal*

Deze taal is in het kader van een ESPRIT-project ontwikkeld en vormt de basis voor de generatie van op talen gebaseerde opmaakprogramma's (editors) en omgevingen. In de taal zijn een systeem voor object-georiënteerde types, syntax-gedreven vertaalschema's en een doeltaalinterface gecombineerd.

abstract window toolkit: AWT *abstracte venster-gereedschapskist*

Een verzameling componenten van grafische gebruikersinterfaces (GUI's) die zijn geïmplementeerd met behulp van autochtone platformversies van de componenten. Deze componenten bevatten een deelverzameling van functionele eigenschappen die alle autochtone platformen gemeenschappelijk hebben. Deze gereedschapskist is grotendeels vervangen door de schommelserie (swing set) van componenten.

acceptance criteria *acceptatiecriteria*

1. De criteria waaraan een softwareproduct moet voldoen voor het voltooiën van een succesvolle testfase of de criteria waarmee een softwareproduct aan de eisen voor een goede afleveringsprocedure moet voldoen.

2. De criteria waaraan een systeem of een component moet voldoen om door een gebruiker, klant of andere gemachtigde entiteit te kunnen worden geaccepteerd.

acceptance testing *acceptatietesten*

1. Een formele testprocedure of testreeks die wordt uitgevoerd om te bepalen of een systeem al dan niet aan de acceptatiecriteria voldoet en om de klant in staat te stellen te bepalen of deze het systeem al dan niet kan accepteren.
2. Een formele testprocedure die wordt uitgevoerd om een gebruiker, klant, of andere gemachtigde entiteit in staat te stellen om te bepalen of een systeem of een component aan vastgelegde acceptatiecriteria voldoet en met behulp waarvan de gebruiker, klant, of andere gemachtigde kan bepalen of het systeem al dan niet kan worden geaccepteerd.

access-control mechanism *toegangsbesturingsmechanisme*

Hardware- of softwarefuncties, operationele procedures, of beheerprocedures die zijn ontwikkeld ten behoeve van een gemachtigde toegang en het voorkomen van een niet-gemachtigde toegang tot een computersysteem.

accessibility *toegankelijkheid*

De mate waarin software het selectieve gebruik of het onderhoud van de componenten ervan vergemakkelijkt.

accuracy *nauwkeurigheid*

1. Een toestand van iets dat geen fouten bevat.
2. Een kwalitatieve inschatting van het ontbreken van fouten, dan wel een inschatting van datgene dat met een kleine fout correspondeert.
3. Een kwantitatieve maat van de grootte van een fout. Deze dient bij voorkeur als een functie van de relatieve fout te worden uitgedrukt. Een grote waarde van deze maat komt overeen met een kleine fout.
4. Een kwantitatieve inschatting van het ontbreken van fouten.

activation *activering*

Het kopiëren van de definitieve vorm van methoden en opgeslagen gegevens in een executeerbare adresruimte. Op deze wijze kunnen methoden op opgeslagen informatie worden toegepast en uitgevoerd.

active DBMS *actief databasemanagementsysteem*

Een conventioneel of passief databasemanagementsysteem dat gecombineerd is met hulpmiddelen waarmee events kunnen worden gedetecteerd en toestanden kunnen worden gevolgd. De eventbehandeling is vaak een op regels gebaseerde verwerking zoals die welke bij expertsystemen wordt toegepast.

active object *actief object*

Een object dat een eigen besturingsmechanisme bezit.

active partition *actieve partitie*

De partitie waarmee op een bepaald moment wordt gewerkt.

active redundancy *actieve redundantie*

Het gebruik in een fouttolerant systeem van redundante elementen die gelijktijdig werken om in een dergelijk systeem fouten te voorkomen of het systeem zichzelf, bij of na het optreden van fouten, te kunnen doen herstellen.

actor *acteur*

Een object in de object-georiënteerde programmering dat de vorm van een gelijktijdig werkend proces (concurrent process) heeft.

actual address *werkelijk adres*

1. Een adres dat de identiteit van een geheugenpositie of -eenheid bepaalt zonder gebruik te maken van een tussenverwijzing.
2. Een adres dat door de ontwerper van het programma of systeem aan een geheugenpositie of -eenheid wordt toegekend.

actual argument (parameter) 1. *actueel argument*; 2. *actuele argumentparameter*

In een programmeertaal heeft de term betrekking op een taalobject dat in een procedure-oproep verschijnt en dat verband houdt met de corresponderende formele parameter die wordt gebruikt bij het uitvoeren van de procedure.

actual code *werkelijke code*

Instructies die met gebruikmaking van absolute adressen zijn geschreven.

actual coding *absolute codering*

Een coderingsmethode waarbij gebruik wordt gemaakt van instructies die absolute adressen bevatten.

actual outcome *actueel resultaat*

Het actuele gedrag van een object indien dit onder gespecificeerde condities wordt getest.

actual parameter list *actuele parameterlijst*

De argumenten die in een specifieke methode-oproep zijn gespecificeerd.

Ada *Ada*

Een hogere programmeertaal waarvan de ontwikkeling financieel door het Amerikaanse Ministerie van Defensie (DOD) werd ondersteund. De taal bevat een mechanisme waarmee meervoudige taken kunnen worden uitgevoerd en bevat een groot aantal functies die nuttig kunnen zijn bij de ontwikkeling van software.

Ada programming support environment: APSE *programmeringsondersteuningsomgeving voor Ada*

Een omgeving die gereedschappen bevat die op maat voor Ada zijn ontwikkeld en waarmee de ontwikkeling en het onderhoud van grote softwaresystemen kan worden ondersteund.

adaptability *aanpassingsvermogen*

Het gemak waarmee software kan worden gemodificeerd, bijvoorbeeld om het aan nieuwe eisen te kunnen laten voldoen. Zie ook flexibiliteit (flexibility).

adaptable user interface *aanpasbare gebruikersinterface*

Een gereedschapskist van Oracle met behulp waarvan applicaties kunnen worden geschreven die tussen verschillende venstergeoriënteerde systemen kunnen worden uitgewisseld. De gereedschapskist bevat een interface op het oproepniveau en een systeemelementmanager (resource manager) en editor voor een groot aantal grafische gebruikersinterfaces (GUI's). Daaronder vallen onder andere die van de MacIntosh, Windows, en het X Window systeem.

adaptation *aanpassing*

Het proces waarin een softwaresysteem of -service wordt gemodificeerd met het doel de functie ervan op een andere wijze of met andere gegevens dan oorspronkelijk de bedoeling was, te laten werken.

adaptation data *aanpassingsgegevens*

Gegevens die worden gebruikt om een programma aan te kunnen passen aan een bepaalde installatieplaats of aan bepaalde condities van de operationele omgeving.

adaptation parameter *aanpassingsparameter*

Een variabele die een bepaalde waarde krijgt toegewezen om een programma aan een zekere installatieplaats of aan gegeven condities van de operationele omgeving aan te kunnen passen.

adaptive communication environment: ACE *adaptieve communicatieomgeving*

Een C++-bibliotheek die in communicatieomgevingen kan worden gebruikt.

adaptive maintenance *adaptief onderhoud*

Software-onderhoud dat wordt uitgevoerd om een computerprogramma in een zich gewijzigde omgeving te kunnen laten werken.

additive attributes *aanvullende attributen*

Attributen of kenmerken waarvoor geen vooraf bepaalde waarden of eigenschappen bestaan en die, indien nodig, altijd aan de lijst met gespecificeerde attributen of kenmerken moeten worden toegevoegd.

address *adres*

1. Getallen, tekens, of tekengroepen die een bepaalde eenheid of geheugenplaats identificeren.
2. Een adres voor elektronische post dat de naam van de geadresseerde, de naam van de dienstverlener (service provider), eventueel een symbolische notatie van het instituut of het bedrijf waartoe de geadresseerde behoort, en een landencode bevat. De naam wordt gevolgd door het '@'-teken en de overige notaties worden door de punt (.) gescheiden.

address (to) *adresseren*

Naar een eenheid of geheugenplaats, die door een identificerend getal, tekens of groep tekens wordt geïdentificeerd, verwijzen.

address field *adresveld*

Elk deel van een computerinstructie dat adressen of informatie waarmee adressen kunnen worden bepaald, bevat. Zie ook adresdeel (address part).

address format *adresformaat*

1. Het aantal en de rangschikking van adresvelden in een computerinstructie.
2. Het aantal en de rangschikking van elementen binnen een adres, zoals de elementen die nodig zijn om een bepaald kanaal, eenheid, sector van een schijf en record in een magnetisch schijfengeheugen te kunnen identificeren.

addressing exception *adresseringsexceptie*

Een uitzonderingstoestand die optreedt indien een programma een adres berekent dat buiten de grenzen van het toelaatbare geheugen-gebied ligt.

address modification *adresmodificatie*

Elke rekenkundige, logische of syntactische bewerking die op een adres van toepassing is of kan zijn.

address part *adresdeel*

Het gedeelte van een instructie dat een adres of een deel van een adres bevat.

address space *adresruimte*

1. De volledige reeks adressen die in een computerprogramma kunnen worden aangeroepen of gebruikt, of waarover een programmeur kan beschikken. Opmerking: In sommige systemen kan dit de reeks fysieke geheugenlocaties zijn die een programma, los van andere programma's, kan gebruiken tezamen met de reeks virtuele adressen waarnaar geheugenplaatsen kunnen verwijzen en die ook toegankelijk zijn voor andere programma's.
2. Het gebied in het hoofdgeheugen dat door één gebruiker kan worden geadresseerd.
3. Het virtuele geheugen dat aan een bepaalde taak wordt toegewezen.
4. Het gebied op een beeldscherm of in de buffer van een besturingseenheid.
5. Het aantal geheugenplaatsen dat een centrale verwerkingseenheid kan adresseren.

a depository of development documents: ADDD *bewaarplaats voor ontwikkelingsdocumenten*

Een software-ontwikkelingsomgeving van het Duitse Gesellschaft für Mathematik und Datenverarbeitung (GMD) die vrij beschikbaar is. De omgeving is ten behoeve van een bepaald Duits project ontwikkeld.

ad hoc testing *ad hoc testen*

Testen die worden uitgevoerd zonder dat daarbij gebruik wordt gemaakt van een erkende ontwerpstechniek van testprocedures of -methodes.

advanced computing environment: ACE *geavanceerde rekenomgeving*

Een consortium dat een open architectuur heeft opgesteld die gebaseerd is op de MIPS R4000 chip. De naam van deze architectuur is ARC (advanced RISC computing) of ARCS (advanced RISC computing specification). De architectuur zal met behulp van de bedrijfs-systemen OS/2 of Open Desktop moeten kunnen werken.

advanced DOS *geavanceerd disk-besturingssysteem*

DOS-opdrachten en -faciliteiten die door ervaren systeemprogrammeurs en gebruikers die hun eigen programma's ontwikkelen, testen en foutvrij maken, worden toegepast.

advanced interactive executive: AIX *geavanceerd interactief bedrijfssysteem*

IBM's Unix-versie. Deze versie vormt de basis van de standaard van de Open Software Foundation.

advanced network systems architecture: ANSA *geavanceerde architectuur voor een netwerksysteem*

Een softwarebus die gebaseerd is op een model voor gedistribueerde systemen en die binnen het kader van het Europese ESPRIT-project is ontwikkeld.

advanced research projects agency network: ARPANET *ARPANET*

Een wijd lokaalnetwerk (WAN) van het Amerikaanse Ministerie van Defensie. Dit netwerk werd in 1968 operationeel en kan als de voorloper van Internet worden beschouwd.

advanced RISC computing (specification): ARC(S) *geavanceerde RISC-architectuur*

De term verwijst naar de standaard hardware-architectuur van de adaptieve communicatieomgeving (ACE). Hierin worden de basiseisen voor de hardware waarmee een ACE-compatibel systeem kan worden gegenereerd, gespecificeerd.

advanced software environment: ASE *geavanceerde software-omgeving*

Een object-georiënteerd applicatie-ondersteuningssysteem van Nixdorf.

Aegis *Aegis*

Een gereedschap voor de computer-ondersteunde software-ontwikkeling (CASE) en bestemd voor het wijzigingsbeheer van een ontwikkelingsproject. Het is een deel van het Gnu-bedrijfssysteem.

afferent *afferent*

De term heeft betrekking op een gegevens- of besturingsstroom van een ondergeschikt module naar een superieur module in een software-systeem.

aggregate expression *geaggregeerde uitdrukking*

Een reeksuitdrukking of een structuuruitdrukking.

aggregation *aggregatie*

Een speciale vorm van overdrachtelijke associaties waarin een groep objecten één enkele semantische entiteit vormt. Bewerkingen die op een dergelijke groep worden uitgevoerd, planten zich meestal voort naar de objecten van de groep.

algebraic language *algebraïsche taal*

Een wetenschappelijke programmeertaal waarin de opdrachten zodanig zijn gestructureerd dat ze op de structuur van algebraïsche uitdrukkingen lijken. Voorbeelden van dit soort talen zijn Algol en Pascal.

Algol *Algol*

Een hogere programmeertaal die voornamelijk wordt (werd) gebruikt om computerprogramma's met behulp van algoritmen uit te drukken.

Algol'60 *Algol'60*

Een hogere programmeertaal die ontworpen is voor toepassingen waarbij numerieke berekeningen en logische processen nodig zijn.

Algol'68 *Algol'68*

Een uitbreidbare hogere programmeertaal die meer mogelijkheden bevat dan Algol'60, maar niet als een uitbreiding van Algol'60 kan worden gezien. Algol'68 is het geesteskind van wijlen prof. dr. ir van Wijngaarden en heeft model gestaan voor de hogere programmeertalen PL/1 en Ada.

algorithm *algoritme*

1. Een gespecificeerd aantal nauwkeurig omschreven regels waarmee een probleem in een beperkt aantal stappen kan worden opgelost. Een algoritme bevat, bijvoorbeeld, een volledige beschrijving van een rekenkundig proces om de waarde van de zevende-machts wortel van een getal tot op een vastgestelde nauwkeurigheid te kunnen bepalen.
2. Elke serie bewerkingen waarmee een gespecificeerde taak kan worden uitgevoerd.

algorithmic *algoritmisch*

Deze term heeft betrekking op methoden om problemen waarin algoritmen voorkomen, op te lossen.

algorithmic language *algoritmische taal*

Een kunstmatige taal, zoals Algol, die ontworpen is om daar algoritmen in uit te drukken.

allocated baseline *toegewezen voetlijn*

In configuratiebeheer heeft de term betrekking op de initieel goedgekeurde specificaties die de ontwikkeling van configuratie-items, die deel uitmaken van een op een hoger niveau geplaatst configuratie-item, sturen.

allocated configuration identification *toegewezen configuratie-identificatie*

De op een bepaald moment goedgekeurde specificaties in configuratiebeheer die de ontwikkeling sturen van configuratie-items die deel uitmaken van een op een hoger niveau geplaatst of aanwezig configuratie-item. Elke specificatie definieert de functionele eigenschappen en kenmerken die vanuit de configuratie-items die op een hoger niveau zijn geplaatst, aan de lagere-niveau configuratie-

items zijn toegewezen. Dit definieert de testen die nodig zijn om aan te kunnen tonen dat de toegewezen functionele eigenschappen en kenmerken zijn gerealiseerd, dat de benodigde interface-eisen met andere geassocieerde configuratie-items zijn afgebakend, en dat de ontwerpbeperkingen (indien daar sprake van is) zijn vastgelegd.

allocation *toewijzing*

1. Het proces waarin de eisen, behoeften, middelen, systeemelementen en andere entiteiten over de componenten van een systeem of programma worden verdeeld.
2. Het resultaat van de verdeling zoals onder punt 1 van deze definitie is beschreven.

alpha test *alfatest*

Een gesimuleerde of werkelijke operationele evaluatie in een ruimte van de afnemer van de software. Bij deze vorm van testen zijn de ontwikkelaars van de software niet betrokken.

American National Standards Institute: ANSI *ANSI*

Het Amerikaanse standaardisatie-instituut dat verantwoordelijk is voor de ontwikkeling en distributie van Amerikaanse standaarden in verschillende toepassingsgebieden waaronder computers en communicaties. ANSI is lid van de ISO.

American standard code for information interchange: ASCII *ASCII*

In deze standaard zijn de 7-bit numerieke codes gedefinieerd die in een computer worden gebruikt om letters, cijfers, leestekens en andere bijzondere tekens en besturingssymbolen weer te geven. Enkele uitzonderingen daargelaten (bijvoorbeeld de mainframes van IBM) gebruiken alle computers deze interne codering.

analysis *analyse*

1. Het methodisch onderzoek van een probleem en het ontleden van dit probleem in kleinere delen voor verder gedetailleerd onderzoek.
2. Het deel van het proces van de software-ontwikkeling waarin de eisen voor het te ontwikkelen product worden gedefinieerd.

analyzers *analysemiddelen*

Softwaregereedschappen die worden gebruikt om broncode te schatten, te meten of te analyseren. Voorbeelden hiervan zijn pad-analyseprogramma's, timing-analyseprogramma's en complexiteit-analyseprogramma's.

Andrew file system: AFS *Andrew bestandssysteem*

Het gedistribueerde bestandssysteem van het Andrew-project. Dit is door de OSF overgenomen als deel van de gedistribueerde rekenomgeving (distributed computing environment: DCE) van OSF.

Andrew project *Andrew project*

Een project voor de ontwikkeling van een gedistribueerd systeem ten behoeve van de ondersteuning van het gebruik van computers voor educatieve en researchdoeleinden aan de Carnegie Mellon Universiteit.

Andrew toolkit *Andrew gereedschapskist*

Een overdraagbare gebruikersinterface die als een onderdeel van het Andrew project is ontwikkeld. De gereedschapskist draait onder het X Window systeem en wordt met behulp van X11R5 (versie 11, vrijgave 5 van het X protocol) gedistribueerd.

an executable (text)editor: aXe *uitvoerbare teksteditor*

Een teksteditor voor het X Window systeem.

ANNA *ANNA*

Een specificatietaal van de Stanford Universiteit (Palo Alto, Californië) waarmee Ada-programma's formeel kunnen worden gespecificeerd. ANNA heeft een specificatie-analyseprogramma en een systeem dat de consistentie van de programma's controleert.

annealing *ontharding*

Een techniek die bij elke minimalisatie of elk leerproces dat gebaseerd is op succesievelijke bijwerkstappen, kan worden toegepast. Die bijwerkstappen (update steps) zijn of willekeurig of deterministisch, waarbij de lengte van de bijwerkstap verhoudingsgewijs overeenkomt met een willekeurig gekozen verzameling parameters die de rol van temperatuur in een onthardingsproces vervullen. In analogie met het ontharden van metalen wordt de temperatuur hoog gehouden gedurende de vroege stadia van het proces teneinde een snellere minimalisatie of sneller leerproces te verkrijgen waarna de temperatuur wordt gereduceerd ten behoeve van een grotere stabiliteit.

annotation *annotatie*

Toegevoegd commentaar, opmerkingen, of nadere verklaringen.

anomaly *anomalie*

Alles wat in de documentatie of werkwijze van software wordt aangetroffen dat van de verwachtingen, die op eerder geverifieerde softwareproducten of referentiedocumenten zijn gebaseerd, afwijkt.

anonymous file transfer program *anoniem bestandsoverdrachtprogramma*

Een bestandsoverdrachtprogramma waarmee bestanden vanuit computers via Internet naar andere computers kunnen worden verplaatst zonder dat de computer waarnaar de bestanden worden overgedragen of de

naam van diegene voor wie de bestanden bestemd zijn, bekend behoeft te zijn.

ANSI/SPARC architecture *ANSI/SPARC architectuur*

Een uit lagen bestaand model van een database-architectuur dat uit een fysiek schema, een conceptueel schema, en een gebruikersoptiek bestaat.

anticipatory buffering *anticiperende buffering*

Een techniek die voor het tijdelijk opslaan (het bufferen) van gegevens wordt gebruikt en waarin deze gegevens in anticipatie op de noodzaak tot het gebruik van de gegevens in een buffer worden opgeslagen.

anticipatory paging *anticiperende paginerig*

Een opslagtoewijzingstechniek waarin pagina's vanuit een hulpgeheugen naar het hoofdgeheugen worden getransporteerd. Dit geschiedt in anticipatie op het gebruik van deze pagina's.

APL *APL*

APL staat voor: A Programming Language. Dit is een hogere programmeertaal met een speciale syntaxis en tekenverzameling. De taal is ontworpen voor het oplossen van uiteenlopende problemen en toepassingen zoals voorkomen in de administratieve gegevensverwerking, het systeemontwerp, wiskundige en wetenschappelijke berekeningen, database toepassingen en het onderwijs in bepaalde vakken.

Apollo *Apollo*

Naam van het bedrijf dat aan de wieg van de ontwikkeling en toepassing van wetenschappelijke en engineering werkstations stond. Het bedrijf is overgenomen door Hewlett Packard.

AppKit *AppKit*

Een verzameling objecten die door de bouwer van applicaties voor de NeXTstep-omgeving kunnen worden gebruikt.

Apple *Apple*

Apple Computer Inc. is het bedrijf dat de MacIntosh-computers vervaardigt. De eerste serie Apple computers zorgden voor een doorbraak in de markt voor computers voor persoonlijk gebruik. Niet in de laatste plaats vanwege de populaire gebruikersinterface. Deze interface kan als de voorloper van die van het Windows-systeem van Microsoft worden beschouwd.

Apple open collaboration environment: AOCE *open samenwerkingsomgeving van Apple*

Een aantal softwareproducten geschikt voor elektronische post, gids- en adressenbestanden, speciale services enzovoort.

applet *applet*

Een programma dat in de programmeertaal Java is geschreven. Een applet kan op een Java-compatibele webbrowser (web-snuffelprogramma) draaien. Voorbeelden van dergelijke browsers zijn Yahoo, Webcrawler, Explorer, HotJava en Netscape Navigator.

Appletalk *Appletalk*

Het particuliere lokale netwerkprotocol van Apple Computers dat door dit bedrijf voor haar serie MacIntosh-processoren is ontwikkeld. Huidige implementaties van Appletalk zijn beschikbaar op Localtalk en Ethertalk.

appliances *hulpmiddelen*

In een netwerk opgenomen eenheden zoals printers, Java-terminals, en cliënten. Deze worden gebruikt om applicaties met behulp van de Java-management-applicatieprogramma-interface (JMAPI) te bouwen.

application *applicatie*

1. Een programma waarmee een concreet karwei kan worden gerealiseerd, anders dan het beheren en roosteren van taken, het verzorgen van interacties enzovoort. Applicatieprogramma's verzorgen in de praktijk voorkomende problemen en taken, zoals administratieve taken en wetenschappelijke berekeningen.
2. De functionaliteit(en) van eindgebruikers. Deze worden verzorgd door één of meer programma's die uit een verzameling met elkaar communicerende objecten bestaat.
3. Een softwaresysteem of -product dat direct met een niet uit software opgebouwd systeem communiceert (mensen, robots, productieapparatuur enzovoort).

application binary interface: ABI *binaire interface van een applicatie*

De interface met behulp waarvan een applicatieprogramma toegang kan verkrijgen tot een besturingssysteem (operating system) en andere diensten. De interface is zodanig geconstrueerd dat het de overdracht van gecompileerde binaire applicaties tussen systemen met dezelfde ABI mogelijk maakt.

application development cycle: AD/Cycle *applicatieontwikkelingscyclus*

Een verzameling producten die compatibel zijn met de Systems Applications Architecture (SAA) van IBM of door deze firma worden ondersteund en die bestemd zijn voor de ontwikkeling van programma-tuur. De producten draaien op werkstations die een centrale vergaarbak (repository) op een mainframe aan kunnen roepen. De

stadia van de cyclus omvatten het opstellen van eisen en behoeften, analyse en ontwerp, de vervaardiging van de applicatie, de implementatie en het testen, alsmede het onderhoud. De gebruikte technologieën zijn onder andere codegeneratoren, op kennis gebaseerde systemen, talen en foutenzoek- en diagnoseprogramma's.

application environment specification: AES *applicatie-omgevingspecificatie*

Een set specificaties van de Open Software Foundation (OSF) bestemd voor de programmering en gebruikersinterfaces. Het is ontwikkeld met het doel de ontwikkelaars van een consistente applicatieomgeving op verschillende soorten apparatuur te kunnen voorzien. AES bevat een bedrijfssysteem (gebruikscommando's en programma-interfaces), een gebruikersomgeving (Motif) en netwerkservices.

application generator *applicatiegenerator*

1. Een soort gereedschap dat softwarespecificaties of -eisen gebruikt om daarmee automatisch applicatiesoftware te genereren. Dit omvat de broncode van de software en programmabesturingsinstructies. Een applicatiegenerator kan een geschikt hulpmiddel binnen een bepaalde software-ontwikkelingsomgeving zijn of het kan als een afzonderlijk hulpmiddel worden gebruikt.
2. Een codegenerator die programma's produceert die één of meer problemen in een specifiek applicatiegebied op kunnen lossen.

application integration architecture: AIA *applicatie integratie architectuur*

De open-standaard specificaties van Digital Equipment Corporation.

application objects *applicatieobjecten*

Applicaties en hun componenten die binnen een object-georiënteerd systeem worden beheerd.

application-oriented language *applicatiegerichte taal*

Een computertaal met mogelijkheden of notaties die vooral op één specifiek applicatiegebied van toepassing zijn; bijvoorbeeld, een taal voor de computer-ondersteunde instructie in het ontwerpen van apparatuur.

application portability architecture: APA *applicatieoverdrachts-architectuur*

Het project van DEC voor overdraagbare applicatiesoftware.

application program interface: API *applicatieprogramma-interface*

1. Een term voor de interface waarmee een applicatieprogramma toegang verkrijgt tot het besturingssysteem en andere diensten. De interface is op broncode-niveau gedefinieerd.
2. De specificatie van de wijze waarop een programmeur die een applicatie schrijft, het gedrag en de toestand van klassen en objecten kan benaderen.

application software *applicatiesoftware*

Software die zodanig is ontwikkeld dat het tegemoet komt aan de specifieke eisen van een bepaalde gebruiker, zoals software ten behoeve van de navigatie van schepen, software voor de procesbesturing enzovoort.

application visualization system: AVS *applicatie-visualisatiesysteem*

Een portabel modulair en op Unix gebaseerd grafisch pakket dat door een consortium leveranciers wordt ondersteund. Dit consortium bestaat onder andere uit Convex, DEC, IBM, HP, SET Technologies, Stardent en WaveTracer.

Archie *Archie*

Een op een server geïmplementeerde database en vraag- en antwoord-systeem (query systeem) dat door de McGill University School of Computer Sciences wordt aangeboden. Het systeem verwerkt op afstand ingediende vragen betreffende software die in globale archieven aanwezig en bereikbaar is en zorgt er voor dat de gevraagde informatie via een bestandsoverdrachtprogramma bij de aanvrager wordt gedeponneerd.

architectural design *architectonisch ontwerp*

1. Het proces waarin de definitie van een verzameling hardware- en softwarecomponenten en hun interfaces wordt gedefinieerd met het doel het raamwerk voor de ontwikkeling van een computersysteem vast te leggen.
2. Het resultaat van het onder punt 1 van deze definitie geschetste proces.

architecture *architectuur*

1. De organisatorische structuur van een systeem, component of goed.
2. De structuur die tussen componenten en hun onderlinge relaties bestaat en de principes en richtlijnen die het ontwerp en de evolutie van de componenten in de tijd regelen. Het is raadzaam de meer specifieke termen, zoals domeinarchitectuur, softwaresysteemarchitectuur enzovoort, te gebruiken. Er bestaan verder verschillende soorten architecturen: strategische architecturen, ondernemingsarchitecturen, standaardisatiearchitecturen, logische en fysieke architecturen enzovoort.

architecture-neutral distribution format: ANDF *architectuur-neutraal distributieformaat*

Een OSF-standaard die ontwikkeld is/wordt ten behoeve van de distributie van software. Programma's worden, voordat ze worden gedistribueerd, in dit formaat gecompileerd. Daarna worden hiervan uitvoerbare versies voor de lokale doelsystemen geproduceerd.

argument *argument*

1. Een grootheid die in een verklaring of commando na de opdracht verschijnt. Zo bestaat het commando: 'verwissel arg1 arg2' uit de opdracht 'verwissel' en twee argumenten: 'arg1' en 'arg2'. Het uiteindelijke resultaat van dit commando kan betekenen dat de twee argumenten in hun geheugenlocaties met elkaar worden verwisseld.
2. Een gegevensitem dat in een procedure- of methode-oproep wordt gespecificeerd. Een argument kan een variabele, een uitdrukking, een constante, of een symbolische naam zijn.
3. Een onafhankelijke variabele; bijvoorbeeld, de variabele m in de vergelijking $E = mc^2$.
4. Een specifieke waarde van een onafhankelijke variabele; bijvoorbeeld, de waarde $m = 24$ kg.
5. Een parameter die wordt uitgewisseld tussen een oproepend en een opgeroepen programma.
6. Een parameter die wordt uitgewisseld tussen een oproepend programma en een deelprogramma of opdrachtfunctie.
7. Een lijst met uitdrukkingen en argumenten die deel uitmaakt van een procedureverwijzing.

argument list *argumentenlijst*

Een tussen haakjes geplaatste lijst met één of meer argumenten die door komma's van elkaar zijn gescheiden en die volgt op de naam van een ingangsconstante, een ingangsvariabele en een algemene naam of een ingebouwde functienaam. Deze lijst wordt aan de parameters van het ingangspunt in bepaalde programmeertalen (zoals PL/1) doorgegeven.

arithmetic expression *rekenkundige uitdrukking*

1. Een uitdrukking die rekenkundige bewerkingen en opdrachtelementen bevat en die tot één enkele waarde kan worden gereduceerd.
2. Een conditionele uitdrukking die een combinatie vormt van rekenkundige termen, bewerkingstekens en dubbele haakjes.
3. Een uitdrukking die een willekeurige combinatie van gegevensnamen, numerieke lettersymbolen en figuurlijke constanten bevat die aan elkaar zijn gekoppeld door één of meer rekenkundige bewerkingstekens en wel op zodanige wijze dat de uitdrukking als geheel tot één enkele waarde kan worden gereduceerd.

arithmetic function *rekenkundige functie*

Een functie die één van de standaard rekenkundige bewerkingen, zoals optellen, aftrekken, vermenigvuldigen, delen enzovoort, vertegenwoordigt.

arithmetic instruction *1. rekenkundige instructie; 2. rekenkundige opdracht*

Een opdracht of instructie waarin het bewerkingsteken een rekenkundige bewerking vertegenwoordigt.

arithmetic operator *rekenkundig bewerkingsteken*

1. Een bewerkingsteken dat kan worden gebruikt in een absolute, een verplaatsbare of een rekenkundige uitdrukking en dat de bewerkingen aangeeft die op de termen binnen een uitdrukking moeten verricht.
2. Een symbool dat het systeem een rekenkundige bewerking laat verrichten.

arithmetic statement *rekenkundige opdracht*

Zie rekenkundige instructie (arithmetic instruction).

array *1. lijst; 2. tabel; 3. reeks; 4. matrix*

- 1a. Een verzameling gegevensitems die allen van hetzelfde type zijn en waarin de plaats van elk item op unieke wijze door een geheel getal is gespecificeerd.
- 1b. Een geordende reeks gegevens die door één enkele naam wordt geïdentificeerd.
2. Een rangschikking van elementen in één of meer dimensies.
- 3a. Een reeks van één of meer waarden.
- 3b. Een geordende verzameling gegevens, structuren of objecten die alle identieke kenmerken bezitten.
4. Een n -dimensionale geordende verzameling gegevensitems die door één enkele naam en één of meer indices worden geïdentificeerd en wel op zodanige wijze dat elk element van deze verzameling afzonderlijk kan worden geadresseerd. Voorbeelden zijn een matrix, een tabel, of een vector.

array declaration *reeksomschrijving*

Een nauwkeurige specificatie van de structuur en de grootte van een reeks.

array element *reeks-element*

Een gegeven in een reeks dat wordt geïdentificeerd door de reeksnaam gevolgd door een achtervoegsel dat de positie van het element binnen de reeks aanduidt.

array expression *reeksuitdrukking*

Een uitdrukking die na evaluatie een waarde, de waarde van de reeks of de reekswaarde, oplevert.

array file *reeksbestand*

Een bestand dat reekselementen bevat.

array index *reeksindex*

Het getal van een element in een reeks of het veld dat het getal dan wel de relatieve positie van een element van een reeks bevat.

array name *reeksnaam*

De naam van een geordende reeks gegevens.

array of structures *structuurreeks*

Een geordende verzameling identieke structuren die worden gespecificeerd door een dimensiekenmerk aan de naam van een structuur toe te kennen.

Artifex *Artifex*

Een Italiaanse software-ontwikkelingsomgeving voor de ontwikkeling van grote event-gedreven gedistribueerde systemen. Het bevat een automatische codegenerator en functies die geschikt zijn voor het snel vervaardigen van prototypen.

artificial intelligence *kunstmatige intelligentie*

Een onderdeel van de informatietechnologie dat zich bezig houdt met de concepten en methoden van symbolische redeneringen met behulp van computertechnieken, alsmede de symbolische representatie van de kennis die bij dergelijke redeneringsprocessen wordt gebruikt.

assemble (to) *assembleren*

Het vertalen van een computerprogramma dat in een assembleertaal is geschreven in de equivalente machinetaal.

assemble-and-go *assembleer en verwerk*

Een werkwijze waarin geen pauzes of onderbrekingen tussen de assemblage, de montage (het linken), het laden en de uitvoering van een computerprogramma voorkomen. Het proces wordt in één vloeiende stroom afgewerkt.

assembled origin *geassembleerde oorsprong*

Het adres van de initiële opslaglocatie dat door een assembleerprogramma, een compiler of een montageprogramma aan een computerprogramma wordt toegewezen.

assembler *assembleerprogramma*

Zie assembleerprogramma (assembly program).

assembler language *assembleertaal*

1. Een taal die opdrachten in een mnemonische vorm bevat en waarbij van een overeenkomst tussen de instructie- en gegevensopmaak van de computer waarvoor de taal wordt gebruikt, sprake is.
2. Een computergerichte taal waarvan de instructies een overeenkomst vertonen met de machinetaal van de computer waarvoor deze assembleertaal bestemd is en bepaalde faciliteiten, zoals macro-instructies, bevat.

assembly code *assembleercode*

Computerinstructies en gegevensdefinities die in een zodanige vorm zijn uitgedrukt dat ze door een assembleerprogramma kunnen worden herkend en verwerkt.

assembly language *assembleertaal*

Zie assembler language.

assembly program *assembleerprogramma*

1. Een programma dat in een mnemonische vorm wordt geschreven en gebruikersprogramma's in machinecode vertaalt.
2. Een computerprogramma dat programma's die in een assembleertaal zijn geschreven, vertaalt in de equivalente machinetaal.

assertion *1. assertie; 2. bewering*

Een logische uitdrukking waarin de bestaande toestand van een programma wordt gespecificeerd of een serie condities waaraan programmavariabelen op een bepaald punt tijdens de uitvoering van het programma moeten voldoen.

asset *asset*

Elk product in de levenscyclus van software dat mogelijkserwijs opnieuw kan worden gebruikt. De definitie van asset omvat: domeinmodel, domeinarchitectuur, eisen, ontwerp, codering, databases, databaseschema's, documentatie, handleidingen, testprocessen enzovoort. Bij hergebruik kan een asset een specifieke hoeveelheid informatie zijn, een specifieke functie beschrijven of uitvoeren, of een functie of kenmerk van een softwaresysteem zijn. Een asset is iets dat de potentie bezit om in verschillende andere grootheden (softwaresystemen, domeinmodellen, ontwerpen) te kunnen worden geïntegreerd. Elk van deze hiervoor genoemde grootheden kan zelf weer als een 'asset' worden opgevat. De term 'component' is een generieke term uit de software-ontwikkeling die vaak als een synoniem voor 'asset' wordt gehanteerd.

asset catalog *assetcatalogus*

Een assetcatalogus bevat de index van informatie betreffende assets die in een bibliotheek voor hergebruik zijn opgenomen. De asset en

de assetcatalogus vormen de informatie betreffende een herbruikbaar software-asset.

asset certification *assetcertificatie*

1. Het proces waarmee wordt ingeschat in hoeverre een asset haar gespecificeerde functie correct uitvoert, in hoeverre het conformeert aan de standaarden voor hergebruik en, mogelijkerwijs en wenselijk, in hoeverre het bewezen is correct te zijn.
2. Het proces waarin wordt bevestigd dat de informatie betreffende een asset, zoals deze in de bibliotheek voor hergebruik is opgenomen, correct is. Dit omvat eveneens de als bekend staande fouten. Om in deze bibliotheek voor hergebruik te worden opgenomen, dient een asset te worden gecatalogiseerd. Sommige van deze bibliotheken kunnen ook als voorwaarde stellen dat een asset eerst dient te worden gecertificeerd.

asset evaluation *asetevaluatie*

Het proces waarin wordt bepaald of een specifiek asset overeenkomt met de eisen en beperkingen van een zeker softwaresysteem zoals dat in de architectuur of het domeinmodel van dat systeem is gedocumenteerd. De asetevaluatie kan plaats vinden indien een asset op het punt staat opnieuw te worden gebruikt. Dit in tegenstelling tot de assetcertificatie die plaats vindt indien een asset in de bibliotheek voor hergebruik wordt opgenomen. Een evaluatie kan eveneens plaats vinden indien door de bibliotheek voor hergebruik wordt geconstateerd dat een asset in een domeinmodel of een generieke architectuur past.

asset library *assetbibliotheek*

Begrip dat synoniem is met bibliotheek voor hergebruik (reuse library). Zie aldaar.

asset source for software engineering technology: ASSET *gebruik van assets in de software-ontwikkeling*

Een programma van het Amerikaanse Ministerie van Defensie dat ten doel heeft het hergebruik van software te bevorderen.

assign (to) *toewijzen*

1. Het toekennen van een waarde aan een variabele.
2. Het reserveren van een gedeelte van een computersysteem, gegevensbestand, of database voor een bepaald doel. Deze reservering dient meestal voor de duur van een programma.

assignment by name *toewijzing via naam*

Bij programmeertalen heeft de term betrekking op het toewijzen van een waarde van een record aan een variabele van een record. Dit heeft alleen betrekking op componenten met hetzelfde identificatiesymbool.

assignment statement *toewijzingsopdracht*

1. In hogere programmeertalen is dit een opdracht die wordt gebruikt om een waarde aan een variabele toe te kennen.
2. Een opdracht die wordt gebruikt om een bestand met de symbolische naam van een eenheid te verbinden.
3. Een rekenkundige of logische variabele of een element van een reeks die wordt gevolgd door een 'is gelijk aan' (=) teken dat weer wordt gevolgd door een rekenkundige of logische uitdrukking.

Association for Computing Machinery: ACM *ACM*

Een van oorsprong Amerikaanse vereniging van computergebruikers en informatietechnologen. Thans heeft de vereniging een groot aantal leden die over de gehele wereld verspreid zijn. ACM heeft een groot aantal speciale werkgroepen (special interest groups) die zich met diverse aspecten en facetten van de informatietechnologie en het computergebruik bezig houden.

Association Française pour la Normalisation: AFNOR *AFNOR*

Frans nationaal standaardisatie-instituut. AFNOR is lid van de International Organization for Standardization (ISO).

Athena project *Athena project*

Een project voor de ontwikkeling van een gedistribueerd systeem ten behoeve van de ondersteuning in het gebruik van computers in de educatieve en researchsector in MIT. Veel van de in dit project ontwikkelde software is nu op vele plaatsen in gebruik. Deze software wordt vooral op ruime schaal in het X Window systeem toegepast.

Atherton technology *Atherton-technologie*

De term heeft betrekking op de technologie in het software-ontwikkelingssysteem dat de basis vormt voor de ATIS-standaard.

Atherton technology tool integration system: ATIS *Atherton-technologiesysteem voor gereedschapsintegratie*

Een object-georiënteerde interface voor een serie diensten waarmee informatie in een gemeenschappelijke vergaarbak (repository) kan worden bewaard, bereikt, en beheerd. ATIS is door Atherton Technology en DEC ontwikkeld en is gebaseerd op een uitgebreide versie van de Software BackPlane, een raamwerk voor een software-ontwikkelingsomgeving die als een kandidaat voor standaardisatie is voorgesteld.

atomic *atomair*

Betrekking hebbende op een bewerking die onder alle omstandigheden nimmer wordt onderbroken of in een onvolledige toestand resulteert.

atomicity *atomaire eigenschappen bezittend*

De eigenschap die kan worden geassocieerd met de situatie dat een bewerking of de toestand wijzigt die van toepassing is op alle participerende objecten die voor wijziging in aanmerking komen of waarin geen wijzigingen optreden of voorkomen.

atomic type *atomair type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens uit één enkelvoudig en ondeelbaar gegevensitem bestaan.

attach (to) *1. vastmaken; 2. verbinden*

Bij het programmeren heeft deze term betrekking op het tot stand brengen van een taak die asynchroon kan worden uitgevoerd tijdens de executie van de hoofdcode.

attribute *1. attribuut; 2. gegevenskenmerk*

1. Een attribuut identificeert de kenmerken en eigenschappen van afzonderlijke objecten. Voorbeelden van attributen zijn de naam, het gewicht, de kleur, de grootte, de elasticiteit enzovoort van een object.
2. Een kenmerk van gegevens zoals de lengte en de indeling van een record of de naam van een gegevensverzameling.

attributed file system: AtFS *toegekend bestandssysteem*

De term verwijst naar een bestandssysteem dat de basis voor een bepaalde gereedschapskist (Shape_VC) vormt. Coöperatieve werkzaamheden worden binnen projecten ondersteund door een statusmodel dat de zichtbaarheid van versie-objecten regelt, vastlegt, en eveneens langdurige transacties voor de synchronisatie van tegelijkertijd optredende aanpassingen bevat. Het concept van objectattributen verschaft de ontwikkelaar een mogelijkheid beheersinformatie op te slaan en deze informatie door te geven aan afzonderlijke gereedschappen. Dit mechanisme is nuttig voor de constructie van geïntegreerde omgevingen uitgaande van een verzameling gereedschappen die onderling geen relatie met elkaar hebben.

audit *administratieve controle*

Een onafhankelijk onderzoek van een product of een aantal producten voor het verkrijgen van een indruk van de compliantie van het product of de producten met specificaties, standaarden, contractuele overeenkomsten of andere criteria die respectievelijk op het product of de producten van toepassing is of zijn.

authoring language *auteurstaal*

Een hoog-niveau programmeertaal die wordt gebruikt voor het ontwikkelen van cursusmateriaal voor computer-ondersteunde instructies.

authoring system *auteurssysteem*

Een programmeringssysteem waar een auteurstaal deel van uitmaakt.

automated verification system *geautomatiseerd verificatiesysteem*

1. Een softwaregereedschap dat een computerprogramma en een bepaalde representatie van de eisen en/of behoeften van dat programma als invoer accepteert en, mogelijkerwijs met behulp van de mens, een bewijs of een weerlegging van de juistheid van het programma produceert.
2. Elk softwaregereedschap dat een deel of het geheel van het verificatieproces automatiseert.

autonomous system *autonoom systeem*

Een zelf-regelend systeem dat in staat is om onafhankelijk bewerkingen met een minimum aan interventies van een operateur uit te voeren.

availability *beschikbaarheid*

De mate waarin een systeem of component operationeel is en, indien nodig, voor verder gebruik toegankelijk is. Beschikbaarheid wordt meestal in de vorm van een waarschijnlijkheid uitgedrukt.

AWK *AWK*

Een taal waarmee patronen kunnen worden afgezocht en verwerkt. Het acroniem bevat de eerste letters van de namen van de auteurs van het pakket: Aho, Weinberger en Kernighan.

B**background** *achtergrond*

1. Bij het roosteren van computerkarweien (jobs) is de achtergrond de computeromgeving waarin processen met een lage prioriteit of processen waarin geen interacties met gebruikers voorkomen, worden uitgevoerd.
2. Onder het Unix-bedrijfssysteem kunnen veel programma's tegelijkertijd worden afgewerkt. Indien een programma 'achter de schermen' draait en er geen interacties tussen dit programma en een gebruiker plaats vinden, wordt van dit programma gezegd dat het in de achtergrond draait.

background processing *achtergrondverwerking*

De uitvoering van een proces met een lage prioriteit indien processen met een hogere prioriteit geen gebruik van de computerelementen maken, dan wel de uitvoering van processen waarin geen interacties met gebruikers nodig zijn.

back-propagation *terugwaartse propagatie*

Een belangrijk algoritme dat voor het leerproces in voorwaarts-gevoede netwerken wordt toegepast. Het algoritme maakt gebruik van een wiskundig foefje dat, indien het netwerk op een computer wordt gesimuleerd, twee doorsneden van het netwerk oplevert, een voorwaarts en terugwaarts of achterwaarts gericht netwerk. Daarnaast levert de simulatie het verschil tussen de gewenste en de werkelijke uitvoer op, alsmede de afgeleiden van dit verschil ten opzichte van de gewichten die aan de netwerkverbindingen zijn toegekend.

back-to-back testing *rug-aan-rug testen*

Testen waarin twee of meer varianten van een programma met dezelfde invoergegevens worden uitgevoerd, de uitvoergegevens worden vergeleken, en fouten in het geval van discrepanties worden geanalyseerd.

backup *backup*

1. Een systeem, component, of bestand dat, dan wel procedure of persoon die beschikbaar is om een belangrijk item in het geval van een fout of extern veroorzaakte catastrofe te vervangen of te helpen herstellen.
2. Een kopie van gegevens en/of programmatuur die in reserve wordt gehouden voor het geval de oorspronkelijke versie van gegevens en/of programmatuur verloren raakt, verminkt wordt, of niet meer kan worden gebruikt. Het installeren van backup-gegevens of -programma's komt veelvuldig voor en wordt restauratie (opnieuw instellen of installeren) genoemd.

backup (to) *aanwijzen/creëren van een backup*

Het creëren of aanwijzen van een systeem, component, bestand, procedure, of persoon voor een situatie zoals beschreven is in punt 1 van de hiervoor geldende definitie.

backup programmer *backup-programmeur*

De assistent-chef van het team van een chef-programmeur; de verantwoordelijkheden van deze functionaris omvatten het leveren van bijdragen aan belangrijke delen van de software die door het team wordt ontwikkeld, het assisteren van de chef-programmeur bij het beoordelen van het werk van andere leden van het team, het vervangen van de chef-programmeur indien dat nodig is. Verder dient de backup-programmeur in het bezit te zijn van de algehele technische kennis van de in ontwikkeling zijnde software.

Backus Naur Form: BNF *Backus Naur vorm*

Een formele taal of metataal die wordt gebruikt voor de specificatie van de syntax van (hoofdzakelijk) kunstmatige talen.

backward recovery *terugwaarts herstel*

1. De reconstructie van een bestand naar een situatie of toestand door het omkeren of terugdraaien van alle wijzigingen die op het bestand zijn uitgevoerd sinds dat bestand zich in die situatie bevond.
2. Een soort herstel van een systeem, database, of ander systeem-element naar een vroegere situatie of conditie, waarin het de vereiste functies uit kan voeren.

base address *basisadres*

1. Een numerieke waarde die als een verwijzing wordt gebruikt bij het berekenen van adressen tijdens de uitvoering van een programma.
2. Een bepaald adres waaruit een absoluut adres wordt afgeleid door het met een relatief adres te combineren.
3. Het instructie-adres of de inhoud van een algemeen register waaruit een logisch adres wordt afgeleid tijdens de uitvoering van een instructie.

base item *basisgegeven*

De automatische, bestuurde of statische variabele of de parameter op grond waarvan een variabele wordt omschreven.

baseline *voetlijn*

1. Een specificatie die of een product dat op formele wijze is beoordeeld en goedgekeurd en daarna als een basis voor verdere ontwikkelingen fungeert en slechts door middel van formele wijzigingsprocedures kan worden veranderd.
2. Een document dat of een verzameling documenten die een formele status hebben gekregen en op een specifiek tijdstip tijdens de levenscyclus van een configuratie-item zijn vastgelegd. Opmerking: Voetlijndocumenten vormen tezamen met goedgekeurde wijzigingen op die documenten de huidige configuratie-identificatie.

baseline management *voetlijnbeheer*

De toepassing in configuratiebeheer van technische en administratieve maatregelen voor het markeren van de documenten en wijzigingen in die documenten die formeel de voetlijnen op specifieke tijdstippen tijdens de levenscyclus van een configuratie-item identificeren en vastleggen.

Basic *Basic*

Een programmeertaal die meestal wordt geïnterpreteerd en die geschikt is voor minder complexe programma-ontwikkelingen. Er bestaan een groot aantal versies van deze taal die lang niet allen onderling uitwisselbaar zijn.

basic block *basisblok*

Een serie van één of meer opeenvolgende uitvoerbare opdrachten die geen sprongen bevat.

Basic development system *Basic ontwikkelingssysteem*

Een hulpprogramma om het programmeren in Basic te versnellen en te vereenvoudigen.

basic instruction *basisinstructie*

Bij het wijzigen van een programma wordt deze instructie veranderd om de gewenste instructie te verkrijgen.

basis test set *basis testserie*

Een verzameling testgevallen voor het bepalen van de kwaliteit van een programma die uit de logica van de code is afgeleid en waarmee een 100% dekking van de verschillende takken die een programma bevat, kan worden bereikt.

batch *partij*

1. Een verzameling te verwerken gegevens.
2. Een reeks programma's, deelprogramma's, bestanden, boodschappen, records of bewerkingen die worden gecombineerd om te worden bewerkt, verwerkt of verzonden.
3. De term verwijst naar een systeem of werkwijze waarin invoergegevens op één bepaald moment worden verzameld en verwerkt, in plaats van dat ze op het moment van aankomst worden verwerkt en waarbij een taak die eenmaal is gestart, zonder verdere additionele invoergegevens of ingrepen van de zijde van gebruikers wordt voltooid.

batch compilation *groepsgewijze compilatie*

Een methode om een aantal programma's te compileren zonder dat specifieke aandacht aan de volgorde van de programma's of de uitvoering van de compilaties behoeft te worden geschonken.

bathtub curve *badkuipkromme*

Een grafiek van het aantal distinctieve kenmerken in een systeem of component als functie van de tijd. De naam is afgeleid van de normaal voorkomende vorm van de grafiek: een periode van een afname van de fouten, gevolgd door een relatief stabiele periode, gevolgd door een periode van een toename van de fouten.

bean *1. bean; 2. booncomponent*

Een herbruikbaar softwarecomponent dat in een applicatie op een zodanige wijze kan worden gecombineerd dat het tezamen met andere booncomponenten één geheel vormt. Dat geheel kan een deel van de applicatie of de gehele applicatie omvatten.

bearer *drager*

Het soort object dat een interface voorstelt.

bedrock *bedrock*

Een bibliotheek van C++-klassen voor gebruikersinterfaces van de MacIntosh.

behaviour *gedrag*

De combinatie van invoerwaarden, vooraf gespecificeerde condities en het vereiste resultaat van een functie of een systeem. De complete specificatie van een functie omvat gewoonlijk één of meer gedragsuitingen of -manifestaties.

benchmark *benchmark*

1. Een standaard op basis waarvan metingen of vergelijkingen kunnen worden gemaakt.
2. Een procedure, probleem of test die kan worden gebruikt om systemen of componenten onderling met elkaar te vergelijken of te vergelijken met de standaard genoemd onder punt 1 van deze definitie.
3. Een herstelbestand.
4. Een speciaal samengestelde en geselecteerde verzameling standaardprogramma's die op verschillende soorten computerapparatuur kan draaien en die gebruikt wordt om de resultaten van de verwerking op die verschillende apparaten te kunnen vergelijken.

Berard object and class specifier: BOCS *Berard's object- en klassepecificatiesysteem*

Object-georiënteerd ontwikkelingsgereedschap van Berard Software Engineering.

Berkeley source distribution: BSD *Berkeley bronverdeling*

De Unix-versies die aan de Universiteit van Berkeley zijn ontwikkeld en van daaruit zijn gedistribueerd. Veel commerciële Unix-implementaties zoals SunOS en Dynix zijn hiervan afgeleid.

beta test *bètatest*

De operationele evaluatie van software op geselecteerde locaties. Bij deze testen zijn de software-ontwikkelaars niet betrokken of aanwezig.

big-bang test *oerknaltest*

Een integratietest waarbij geen specifieke testen van de integreernde componenten, waaruit het systeem bestaat, afzonderlijk en voor de integratie van de componenten heeft plaatsgevonden.

bin *bin*

Een index (directory) waarin programma's zijn opgenomen. Een index bevat meestal een aantal indexen en deelindexen (subdirectories) die bin worden genoemd. De oproep vanuit een systeem vindt plaats met behulp van specifieke verwijzingen, bijvoorbeeld /bin en /usr/bin.

binary compatibility standard: BCS *binaire compatibiliteitsstandaard*

De binaire applicatie-interface (ABI) van 88open. 88open is een consortium van bedrijven dat zich ten doel heeft gesteld een open computeromgeving te creëren voor de familie van de 88000 RISC processoren van Motorola.

binary digit: bit *binair getal*

1. Een eenheid van informatie die door een 1 of een 0 kan worden gerepresenteerd.
2. Een element dat een computergeheugen(plaats) die een eenheid van informatie, als bedoeld onder punt 1 van deze definitie, op kan slaan of vast kan houden.
3. Een getalteken dat wordt gebruikt om één van de twee getallen in het tweetallige talstelsel te representeren: nul (0) of één (1).

binary operator *binaire verwerkingsopdracht*

Een verwerkingsopdracht (operator) die twee argumenten heeft.

bind (to) *binden*

1. Het toekennen van een waarde aan een identificatiesymbool, een variabele of een parameter.
2. Het koppelen van een absoluut adres, een virtueel adres of het identificatiesymbool van een machine met een symbolisch adres of een label in een programma.

binding *binding*

1. De selectie van de methode waarmee een gevraagde dienst kan worden verricht en de keuze van de gegevens die door de geselecteerde methode moeten worden opgevraagd. Zie verder statische en dynamische binding.
2. Het toewijzen of koppelen (verbinden) van systeemfaciliteiten aan een reeks taken; dit vindt plaats voordat deze reeks taken wordt uitgevoerd.

bindings *bindingen*

Koppelingen tussen verschillende architecturen, platformen, systemen, interfaces of structuren waarmee een naadloze functionaliteit kan worden verkregen.

bit *bit*

Een zeer kleine hoeveelheid informatie die de waarde 1 of 0 heeft. Bits worden over het algemeen tot grotere informatie-eenheden gebundeld. Zo vormt een groep van 8 gebundelde bits een byte.

Bitnet *Bitnet*

Een academisch en researchnetwerk dat ongeveer 2500 computers onderling met elkaar verbindt. Deze computers zijn over het algemeen IBM mainframes. Het netwerk verschaft de gebruikers elektronische post en bestandsoverdrachtsservices die op de opslag- en doorzendmethode (store and forward) en de NJE-protocollen van IBM zijn gebaseerd. Het verkeer tussen Bitnet en Internet kan via verschillende gateway-voorzieningen worden geregeld. Bitnet wordt thans door de Corporation for Research and Educational Networking (CREN) beheerd.

bit steering *bitgeleiding*

Een techniek uit de microprogrammering waarin de betekenis van een veld in een micro-instructie afhankelijk is van de waarde van een ander veld in de micro-instructie.

bit string *bitrij*

Een ononderbroken stroom bits.

bitwise operator *bitsgewijze bewerkingsopdracht*

Een bewerkingsopdracht (operator) die twee waarden manipuleert door ieder bit van de ene waarde te vergelijken met het corresponderende bit van de andere waarde.

black box *blackbox*

1. Een systeem of component waarvan de invoergegevens, de uitvoergegevens, en de algemene functies bekend zijn, maar waarvan de inhoud of de implementatie onbekend of niet relevant is.
2. Betrekking hebbende op een benadering die een systeem of een component behandelt op de wijze die in punt 1 van deze definitie is beschreven.

black box test *blackboxtest*

Zie het ontwerp van een functioneel testproces (functional test case design).

block *blok*

1. Een blok is in Java elke code die tussen linker- en rechterhaakjes is geplaatst. Voorbeeld: [x = 1;].
2. Een reeks aaneengesloten elementen zoals geheugenlocaties, opdrachten in computerprogramma's, woorden, tekens, cijfers of bits die als een eenheid wordt behandeld.

3. Een verzameling aaneengesloten records die als een eenheid wordt vastgelegd. Blokken worden door bloktussenruimten van elkaar gescheiden en elk blok kan één of meer records bevatten.
4. Een groep bits of cijfers van n posities die als een eenheid wordt behandeld en/of verzonden.
5. Een tekstdeel bij het verwerken van tekst.
6. De gegevenseenheid die tussen een toepassingsprogramma en een terminal wordt verzonden.
7. Een beginblok of een procedureblok.
8. Een vooraf bepaald gebied voor bestanden op schijfengeheugen.
9. Een gedeelte van een programma of van gegevens dat in één sector van een schijfengeheugen is opgeslagen.

block diagram *blokdiagram*

Een diagram of schema van een systeem, een computer, een programma of een eenheid waarin de voornaamste elementen of delen worden weergegeven door geometrische figuren op de juiste wijze met elkaar te verbinden. Met een dergelijk schema worden zowel de basisfuncties als hun onderlinge samenhang vertoont.

blocking factor *blokkingsfactor*

Het aantal records, woorden, tekens, of bits in een blok.

block-structured language *blok-gestructureerde taal*

Een ontwerp- of programmeertaal waarin series opdrachten, blokken genoemd, worden gedefinieerd. Deze series beginnen meestal met begin- en eind-scheidingstekens, terwijl variabelen of labels die in een blok zijn gedefinieerd, niet buiten dat blok worden herkend.

Booch *Booch*

Een bibliotheek met meer dan 500 herbruikbare objecten. De broncode hiervan kan worden toegepast in het object-georiënteerde ontwerp en in standaard abstracte types die op verschillende wijzen kunnen worden geïmplementeerd. Verschillende hulpprogramma's verzorgen zoeken, sorteren, tekenvergelijkingen en dergelijke.

boolean *booleaanse grootheid*

De term verwijst naar een uitdrukking of een variabele die slechts de waarden 'waar' of 'onwaar' aan kan nemen. De programmeertaal Java kent, bijvoorbeeld, booleaanse variabelen en de symbolen TRUE en FALSE.

boot (to) *booten*

Het initialiseren van een computersysteem door het vrijmaken van het geheugen en het opnieuw laden van het besturingssysteem.

bootstrap *zelfstartend programma*

Een kort computerprogramma dat voortdurend in het geheugen aanwezig is of dat daar gemakkelijk in kan worden geladen en waarvan de uitvoering een groter programma, zoals een besturingssysteem of een laadprogramma, in het geheugen doet plaatsen.

bootstrap (to) *zelfstarten*

Het gebruiken van een programma zoals in de voorgaande term is beschreven.

bootstrap loader *zelfstartend laadprogramma*

Een kort computerprogramma dat wordt gebruikt om een zelfstartend programma (bootstrap) te laden.

bottom-up *van onder-naar-boven*

De term heeft betrekking op een activiteit die met het laagst-gelegen component van een hiërarchisch geheel start en zich continu door de daarboven gelegen niveaus naar het hoogst-gelegen component in die hiërarchie verplaatst.

bottom-up testing *testen van 'onder-naar-boven'*

Een testmethodiek waarbij de componenten op het laagste niveau eerst worden getest. Deze worden dan gebruikt om de componenten op een daarboven gelegen niveau te testen enzovoort. Het proces wordt net zo lang voortgezet totdat de top van de hiërarchie is bereikt.

boundary value *grenswaarde*

1. De waarde van een gegeven die overeenkomt met de minimale of maximale waarde van invoergegevens, intern aanwezige gegevens, of uitvoergegevens die als zodanig voor een systeem of een component zijn gespecificeerd.
2. Een invoer- of uitvoerwaarde die in het grensgebied van twee equivalente klassen ligt, of op een incrementele afstand aan beide zijden van de grens zijn gelegen.

boundary value analysis *grenswaarde-analyse*

Een techniek voor het ontwikkelen van een test voor een component waarbij deze test zodanig is geconstrueerd dat ze de grenswaarden eveneens omvatten.

boundary value coverage *grenswaardedekking*

Het percentage grenswaarden van de equivalente klassen van een component die met behulp van een test zijn onderzocht.

boundary value testing *testen van grenswaarden*

Zie grenswaarde-analyse (boundary value analysis).

bounding box *insluitblok*

Voor een rasterobject is dit de kleinste rechthoek die alle niet volledig transparante pixels omsluit.

Bourne shell *Bourne shell*

De meest gebruikte Unix shell.

box diagram *blokdiagram*

Een diagram van een besturingsstroom dat uit een rechthoek bestaat die in kleinere rechthoeken is onderverdeeld en opeenvolgende stappen, IF-THEN-ELSE-bewerkingen, herhalingsfuncties, en CASE-clausules voor kunnen stellen.

branch *sprong (opdracht)*

1. Een conditionele overdracht van de besturing van de ene instructie in een component naar een andere instructie, dan wel de onconditionele besturingsoverdracht van de ene instructie naar de andere in een bepaald component met uitzondering van de onmiddellijk daaropvolgende instructie of, indien een component meer dan één ingangspunt heeft, een overdracht van de besturing naar een bepaald ingangspunt van het component.
2. Een reeks instructies die tussen twee opeenvolgende beslissingsinstructies wordt uitgevoerd.
3. Een directe verbinding tussen twee aangrenzende knooppunten in een netwerk.
4. Een geconditioneerde sprong.

branch (to) *1. springen; 2. vertakken*

1. Een sprong (opdracht) uitvoeren.
2. In meerdere takken overgaan.

branch condition *sprongconditie*

Zie beslissingsconditie (decision condition).

branch condition combination coverage *dekking van sprongopdracht-combinaties*

Het percentage combinaties van de keuzes van alle sprongopdrachten in elke beslissing die in een test of testserie worden uitgevoerd.

branch condition combination testing *testen van sprongopdracht-combinaties*

Een techniek voor het ontwikkelen van testen of testseries waarin de testen zo zijn ontworpen dat ze combinaties van sprongcondities uitvoeren.

branch condition coverage *dekking van sprongcondities*

Het percentage van de keuzes van een sprongconditie in elke beslissing die door een test of testserie wordt uitgevoerd.

branch condition testing *testen van sprongcondities*

Een techniek voor het ontwikkelen van testen of testseries waarin de testen zodanig zijn geconstrueerd dat ze de keuzes van sprongcondities uitvoeren.

branch coverage *dekking van sprongopdrachten*

Het percentage sprongopdrachten dat door een test of testserie wordt afgedekt.

branch instruction *sprongopdracht*

Een opdracht die ervoor zorgt dat er een keuze tussen uit te voeren opdrachten wordt gemaakt. Na het maken van de keuze wordt een sprong naar die gekozen opdrachten in het programma gemaakt waarna de uitvoering van het programma wordt vervolgd op de plaats waarnaar toe is gesprongen.

branch outcome *resultaat van een sprong(opdracht)*

Zie beslissingsresultaat (decision outcome).

branch point *sprongpunt*

1. Een punt in een programma, vanwaar naar een ander deel in het programma wordt gesprongen.
2. Een plaats in een deel van een programma, waar een beslissingsinstructie wordt geselecteerd.

branch testing *testen van sprongopdrachten*

Een techniek voor de ontwikkeling van de testen van een component waarin de testen zodanig zijn geconstrueerd dat ze de keuzes van sprongopdrachten uitvoeren.

break key *onderbrekingstoets*

1. Een toets waarmee een bepaalde bewerking wordt onderbroken of beëindigd.
2. Een toets die in Unix wordt gebruikt bij het oproepen van een modem en waarmee wordt aangegeven dat het systeem een foutieve keuze voor wat betreft het soort modem heeft gemaakt.

breakpoint *breekpunt*

Een punt in een computerprogramma waarin (waarop) de uitvoering van het programma kan worden onderbroken voor de handmatige of automatische controle van de prestaties, de resultaten, en de correcte voortgang van de verdere uitvoering van het programma. Specifieke breekpunt-types zijn codebreekpunt, gegevensbreekpunt, dynamisch breekpunt, epiloog-breekpunt, programmeerbaar breekpunt, proloog-breekpunt, en statisch breekpunt. Opmerking: Van een breekpunt wordt gezegd dat het actief wordt indien zowel een punt in het programma als een gebeurtenis die tot opschorting van de uitvoering van het programma in dat punt leidt, zijn gedefinieerd;

van een breekpunt wordt verder gezegd dat het is geïnitieerd indien de uitvoering van het programma daadwerkelijk is opgeschort.

British Standards Institution: BSI *BSI*

Het nationale normalisatie-instituut van het Verenigd Koninkrijk. BSI is lid van de ISO.

browser *snuffelaar*

Een systeem, programma of gereedschap dat wordt gebruikt bij de navigatie door documenten die hypertext bevatten.

BSD Unix *BSD-Unix*

Zie Berkeley bronverdeling (Berkeley source distribution).

bubble chart *bellenkaart*

Een schema van een gegevensstroom, gegevensstructuur, of enig ander type grafiek of diagram waarin entiteiten met behulp van cirkels (bellen) worden voorgesteld en relaties door lijnen, die de cirkels verbinden, worden gerepresenteerd.

buffer *buffer*

1. Een kleine opslagruimte waarin informatie tijdelijk wordt bewaard totdat het voor verdere verwerkingen nodig is. Een dergelijke buffer wordt vaak gebruikt om verschillen in de snelheid van de gegevensstromen, de tijdsduren van het optreden van gebeurtenissen, of de hoeveelheden gegevens die door eenheden of processen kunnen worden verwerkt, te compenseren. Buffers komen in grote verscheidenheid voor: printers hebben buffers om de volgende zinnen of pagina's die geprint moeten worden in op te slaan; tekstverwerkers noemen de kopieën van de bestanden die moeten worden bijgewerkt meestal buffers; de computer zelf kan hardware-buffers hebben om daar gegevens in op te slaan die later moeten worden getransporteerd of bewerkt enzovoort.
2. Een routine die de doelstelling genoemd in punt 1. van deze definitie verzorgt.

buffer (to) *bufferen*

Het toewijzen, plannen of gebruiken van eenheden of opslagruimten zoals onder punt 1 van de vorige definitie is beschreven.

bug *fout*

Zie de andere definities van fout (error, fault).

build *bouwwerk*

Een operationele versie van een systeem of een component die een gespecificeerde deelverzameling van de operationele mogelijkheden van het uiteindelijke product bevat.

built-in *ingebouwd*

Bij programmeertalen heeft deze term betrekking op een taalobject dat binnen de programmeertaal een specifieke duidelijk gedefinieerde functie heeft. Dit zijn bijvoorbeeld de ingebouwde sinusfunctie ('sin') in PL/1 en het vooraf gedefinieerde gegevenstype INTEGER in Fortran, Algol en Pascal.

built-in function *ingebouwde functie*

Een functie die door een programmeertaal kan worden uitgevoerd.

busy *1. in bedrijf; 2. bezig*

De term heeft betrekking op een systeem of een component dat operationeel is, wordt onderhouden, of in gebruik is.

busy time *bezettingsduur*

De tijd gedurende welke een systeem of component operationeel is, wordt onderhouden, of in gebruik is.

byte *byte*

1. Acht in een rij geplaatste bits. Ieder bit in een byte kan de waarde 0 of 1 bezitten. Een byte heeft 256 verschillende mutaties van enen en nullen en derhalve kunnen in een byte 256 verschillende soorten informatie worden opgeslagen. De meeste computers gebruiken de ASCII-code waarin is vastgelegd wat iedere combinatie van enen en nullen in een byte betekent. Zo betekent 01000001 A, 01000010 B, en 00001010 'einde regel, begin een nieuwe regel'.
2. Een element van een computergeheugen dat een groep van acht in een rij geplaatste bits kan opslaan of bewaren.

bytecode *bytecode*

Machine-onafhankelijke code die door de Java-compiler wordt gegenereerd en door het Java-vertolkingsprogramma wordt uitgevoerd. Een compileerprogramma vertaalt Java-broncode in bytecode en slaat dat in een klassebestand op. Java-chips kunnen deze bytecode direct verwerken, maar alle andere computers hebben een vertolkingsprogramma of een compiler nodig om de bytecode in de objectcode van die computers te vertalen.

C**C** *C*

Een programmeertaal die tegelijkertijd met Unix door D. Ritchie van Bell Laboratories is ontwikkeld. Bijna alle Unix-programma's zijn in deze taal geschreven. C-programma's bevatten naast procedurele en rekenkundige uitdrukkingen ook op machinetaal lijkende instructies. C is populair geworden vanwege de simpliciteit, de

efficiëntie, en de flexibiliteit. C-programma's kunnen op gemakkelijke wijze aan nieuwe programmeeromgevingen worden aangepast.

C++ *C++*

Een uitbreiding van de C-taal die ontwikkeld is door B. Stroustrup van Bell Laboratories. De taal ondersteunt de ontwikkeling van object-georiënteerde programma's.

cache *cache*

Een klein en snel geheugen waarin recentelijk opgevraagde gegevens worden bewaard. Het doel van een dergelijk geheugen bestaat uit het creëren van een snelle toegang tot deze gegevens.

call *oproep*

1. Een overdracht van de besturing van een bepaald softwaremodule naar een ander. Dit gebeurt meestal met de verwachting dat de besturing aan het oproepende module terug zal worden gegeven.
2. Een computerinstructie die de besturing van een softwaremodule op de wijze beschreven in punt 1 van deze definitie overdraagt aan een ander module en waarbij over het algemeen de parameters naar en van het module eveneens worden overgedragen.
3. Een tot uitvoering gebracht programma, module of subroutine. Dit is meestal het resultaat van bepaalde ingangscondities en een specificatie van een sprong naar een ingang (entry).
4. Een poging die wordt ondernomen om contact te krijgen met, bijvoorbeeld, een gebruiker of een systeem.

call (to) *oproepen*

1. Het ten uitvoer brengen van een oproep.
2. Het overdragen van de besturing en eventueel de parameters van een bepaald module aan een gespecificeerd ander module of object.

call by name *adresoproep*

Een methode voor de overdracht van parameters waarbij het oproepende module het opgeroepen module van een symbolische uitdrukking voorziet dat de over te dragen parameter representeert, en een dienstverlenende routine die de uitdrukking evalueert en de resulterende waarde doorgeeft aan het opgeroepen module. Opmerking: Omdat de uitdrukking steeds wordt geëvalueerd wanneer de corresponderende formele parameter in het opgeroepen module wordt gebruikt, kan de waarde van de parameter tijdens de uitvoering van het opgeroepen module veranderen.

call by reference *referentieoproep*

Een methode voor de overdracht van parameters waarbij het oproepende module het opgeroepen module van het adres van de over te dragen

parameter voorziet. Opmerking: Met deze methode heeft het opgeroepen module de mogelijkheid de waarde van de parameter die in het oproepende module is opgeslagen, te wijzigen.

call by value *waarde-oproep*

Een methode voor de overdracht van parameters waarbij het oproepende module het opgeroepen module van de werkelijke (actuele) waarde van de over te dragen parameter voorziet. Opmerking: Met deze methode kan het opgeroepen module de waarde van de parameter, zoals die in het oproepende module is opgeslagen, niet veranderen.

called program *opgeroepen programma*

Een programma waarvan de uitvoering door een ander programma (het oproepende programma) wordt verzocht of waarvan de uitvoering door een opdracht wordt geïnitieerd.

call graph *oproepschema*

Een schema of diagram dat de modules in een systeem of een computerprogramma identificeert en laat zien welke modules elkaar oproepen. Opmerking: Het resultaat is niet noodzakelijkerwijs gelijk aan dat wat in een structuurkaart wordt weergegeven.

calling program *oproepend programma*

Een programma dat om de uitvoering van een ander programma (het opgeroepen programma) vraagt.

calling sequence *oproepprogramma*

1. Een reeks instructies (en soms ook gegevens) die nodig is om een oproep tot stand te brengen.
2. Een terminal-navraaglijst.

call list *oproeplijst*

De geordende lijst van de argumenten die in een oproep naar een softwaremodule worden gebruikt.

candidate key *kandidaatsleutel*

Een minimale verzameling attributen die een object of een link op unieke wijze identificeert.

capture/playback tool *opname en weergavegereedschap*

Een testgereedschap dat de ingevoerde testgegevens voor de te testen software vastlegt en opslaat. Deze ingevoerde en opgeslagen test kan op een later tijdstip worden gebruikt om de test te reproduceren.

cardinality *kardinaliteit*

Bij een gegevensstructuur, een relationele database of een bestandsstructuur verwijst de term naar het aantal tupels in een verwantschap of relatie.

CASE-clause *1. CASE-clausule; 2. meervoudige selectie*

Een meerwegssprong (meerwegsvertakking) met één enkele ingang en één enkele uitgang die een besturingsuitdrukking definieert, de verwerking van iedere waarde van die uitdrukking specificceert, en de besturing in alle gevallen direct aan de opdracht volgend op de meervoudige selectie (CASE-clausule) teruggeeft.

CASE data interchange format: CDIF *CASE-gegevensuitwisselingsformaat*

Een in ontwikkeling zijnde standaard voor de uitwisseling van gegevens tussen verschillende CASE-gereedschappen.

CASE framework *CASE-raamwerk*

Een verzameling producten en conventies met behulp waarvan CASE-gereedschappen in een coherente omgeving kunnen worden geïntegreerd.

CASE integration services: CIS *CASE integratiediensten*

Een commissie die gevormd is om de integratiestandaarden van CASE-gereedschappen die betrekking hebben op ATIS (A Tools Integration Standard), te bediscussiëren.

CASE*Method *CASE*Method*

Een analyse- en ontwikkelingsmethode van Oracle. De methode is bedoeld voor de ontwikkeling van toepassingen die betrekking hebben op informatiebeheer.

CASE tools *CASE-gereedschappen*

Programmatuur die de toepassing van CASE-methoden in een software-project ondersteunt.

casting *vervorming*

De expliciete conversie van het ene gegevenstype naar een ander gegevenstype.

catalog *catalogus*

1. De index met informatie betreffende assets. Deze is van toepassing op de inhoud van een bibliotheek voor hergebruik.
2. Een inhoudsopgave van posities van bestanden en bibliotheken.
3. De verzameling van alle gegevensindexen die door het besturingsprogramma wordt gebruikt om de positie van een fysiek bestand te bepalen.

4. Een gegevensverzameling die een omschrijving bevat van gegevensverzamelingen, andere catalogi en het direct toegankelijke geheugen dat aan een gegevensverzameling en de catalogi is toegewezen.

catalog (to) *catalogiseren*

1. De informatie betreffende assets in een index of catalogus plaatsen.
2. Het invoeren van gegevens vanuit een bestand of een bibliotheek in een catalogus.
3. Het opnemen van de bestandsidentificatie van een gegevensverzameling in een catalogus.
4. Het invoeren van een fase, module of boek in één van de systeembibliotheken.

cataloguing *catalogisering*

Het plaatsen van informatie betreffende een asset in een bibliotheek voor hergebruik van software. De asset en haar catalogusinformatie worden daarmee een herbruikbaar software-asset.

catastrophic failure *catastrofale fout*

Een fout in kritische software.

cause-effect graph *oorzaak/gevolg-diagram*

Een grafische weergave van ingevoerde gegevens of stimuli (oorzaken) en de daarbij behorende uitvoerwaarden (gevolgen) die kunnen worden gebruikt voor het ontwerpen van testen en testseries.

cause-effect graphing *oorzaak/gevolg schematering*

Een techniek die wordt toegepast bij de ontwikkeling van testen en testseries. De techniek gebruikt oorzaak/gevolg-diagrammen voor de ontwikkeling van de testen.

C beautifier: cb *C-verfraaiier*

Een gereedschap waarmee de syntax van de broncode van C kan worden verfraaid.

central processing unit: CPU *centrale verwerkingseenheid*

Een eenheid in een computer die zorgt voor het ophalen, decoderen en uitvoeren van de computerinstructies. Deze eenheid bestaat meestal uit een besturingseenheid en een rekenkundige en logische eenheid.

certification *certificatie*

1. Een op schrift gestelde garantie dat een systeem of component conform de specificaties is ontwikkeld en geproduceerd en dat het voor operationeel gebruik is geaccepteerd. Bijvoorbeeld

- een geschreven autorisatie dat een computersysteem veilig is en in een gedefinieerde omgeving mag werken.
2. Een formele demonstratie dat een systeem of component conform de specificaties is ontwikkeld en geproduceerd en dat het voor operationeel gebruik is geaccepteerd.
 3. Het proces waarin wordt vastgesteld dat een systeem of systeemcomponent aan de gespecificeerde eisen voldoet en geschikt is om de vereiste operationele functies te verrichten.

chained list *kettinglijst*

Een lijst waarin de gegevens in een willekeurige volgorde voor kunnen komen maar waarin elk gegeven een verwijzing bevat om de positie van het volgende gegeven te kunnen bepalen.

change dump *wijzigingsdump*

Een selectieve dump van die geheugenplaatsen waarvan de inhoud sinds een bepaald moment of een zekere gebeurtenis is gewijzigd.

change management *wijzigingsbeheer*

Een consistente verzameling technieken en technische hulpmiddelen die behulpzaam kunnen zijn bij de evolutie, de constructie en het beheer van het ontwerp en de implementatie van een object of een systeem.

channel capacity *kanaalcapaciteit*

De maximale hoeveelheid informatie die via een bepaald kanaal per tijdseenheid kan worden overgedragen; dit wordt meestal in bits per seconde of baud uitgedrukt.

character *teken*

Een letter, cijfer, leesteken of ander symbool waarmee gegevens kunnen worden weergegeven of in een code kunnen worden gebruikt.

character array *1. tekenreeks; 2. tekenrij*

Een benoemde lijst of matrix met bestanden die uit tekens zijn opgebouwd.

character attribute *tekenattribuut*

Bij een gegevensstroom heeft de term betrekking op een code die één enkele eigenschap van een teken of een tekenreeks omschrijft, bijvoorbeeld kleur, tekenset, afbeelding (vet, cursief) enzovoort. Een teken kan meer dan één tekenattribuut bezitten.

character assembly *tekensamenstelling*

De samenstelling van bits tot tekens als deze bits bij een terminal aankomen. Bij een communicatiebesturingseenheid wordt dit samenstellen van tekens door het besturingsprogramma of een scanner

verricht, afhankelijk van het type scanner dat hiervoor beschikbaar is.

character set *tekenset*

Een reeks unieke tekens zoals de 26 letters van het alfabet, de 0 en de 1 van het booleaanse stelsel, de signalen van de Morse-code of de 128 tekens van het ASCII-set.

character type *tekentype*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens de waarden van gespecificeerde tekens aan kunnen nemen en waarop bewerkingen, zoals verwisselingen, kunnen worden uitgevoerd.

character variable *tekenvariabele*

De naam van een bestanddeel van een teken waarvan de waarde wordt toegekend of gewijzigd tijdens de uitvoering van het programma waarvan deze variabele deel uitmaakt.

checkout *test*

Een test die in de operationele of ondersteunende omgeving wordt uitgevoerd en waarmee men zich verzekert van het feit dat een softwareproduct na installatie volgens de specificaties werkt.

checkpoint *controlepunt*

1. Een plaats in een programma waar een controle wordt uitgevoerd of waar het vastleggen van gegevens voor een eventuele herstart plaats vindt.
2. Een punt waar gegevens betreffende de status van een bewerking en het systeem kunnen worden vastgelegd en wel zodanig dat deze bewerkingsstap later opnieuw kan worden gestart.

checksum *controlesom*

Een getal dat wordt berekend door alle tekens van een bestand op een specifieke wiskundige manier bij elkaar op te tellen. De controlesom kan worden gebruikt voor te verzenden berichten waarbij deze controlesom met de berichten kan worden verzonden. Na ontvangst kan de controlesom nogmaals worden berekend en worden vergeleken met de verzonden controlesom. Indien deze twee controlesommen gelijk zijn, wordt over het algemeen aangenomen dat het bericht correct is verstuurd en ontvangen.

chief programmer *chef-programmeur*

De leider van het team van de chef-programmeur; dit is een programmeur met een bepaalde senioriteit waarvan de verantwoordelijkheden bestaan uit het verzorgen van belangrijke delen van de software-ontwikkeling die aan het team is toegewezen, het coördineren van de activiteiten van het team, en het beoordelen van het

werk van de overige leden van het team. De chef-programmeur bezit een allesomvattende kennis van de software die wordt ontwikkeld.

chief programmer team *team van de chef-programmeur*

Een software-ontwikkelingsgroep die een chef-programmeur, een backup-programmeur, een secretaresse/bibliothecaris, en voor zover nodig of vereist additionele programmeurs en specialisten bevat. Het team gebruikt procedures die zodanig zijn geformuleerd of opgesteld dat daarmee de interne communicatie van de groep wordt vergroot en een optimaal gebruik van de vaardigheden van de leden van het team kan worden bereikt.

child version *zoonversie*

Een versie van een versie.

Chorus *Chorus*

Een gedistribueerd bedrijfssysteem dat aan het Franse Institut de Recherche en Informatique et Automatique (INRIA) is ontwikkeld.

Class *Class*

Een taal die in het Andrew-project is ontwikkeld. Het was één van de eerste pogingen om object-georiënteerde functies aan de taal C toe te voegen.

class *klasse*

1. Een implementatie die kan worden gebruikt om meervoudige objecten met hetzelfde gedrag te creëren. Een object is een exemplaar (kopie) van een klasse. Types classificeren objecten in overeenstemming met een gemeenschappelijke interface; klassen classificeren objecten volgens een gemeenschappelijke implementatie.
2. Het prototype van een object in een object-georiënteerde taal; het is analoog aan een afgeleid type in een procedurele taal.
3. In Java is klasse een type dat de implementatie van een specifiek soort object definieert. Een klassedefinitie definieert exemplaar- en klassevariabelen en -methoden, evenals de specificatie van de interfaces, de klasse-implementaties en de directe superklasse van de klasse. Indien de superklasse niet expliciet is gedefinieerd, zal het object impliciet die superklasse vormen.

class condition *klasseconditie*

Een bewering dat de inhoud van een gegevensbestanddeel geheel alfabetisch dan wel geheel numeriek is. Deze bewering kan waar of onwaar zijn.

class diagram *klasseschema*

Een klassenschema geeft in diagramvorm de klassen van een systeem en hun onderlinge relaties weer. Daaronder vallen eveneens de overerving, aggregaties en associaties. Dit werd voorheen een objectmodel genoemd.

class inheritance *klasse-overerving*

De constructie van een klasse door middel van een incrementele modificatie van andere klassen.

class library *klassebibliotheek*

Een bibliotheek van herbruikbare klassen die in een object-georiënteerd programmeringssysteem kunnen worden gebruikt.

class library for high energy physics: CLHEP *klassebibliotheek voor natuurkundige toepassingen*

Een C++-klassebibliotheek voor een specifiek toepassingsgebied, namelijk dat van de hoge-energie toepassingen.

class method *klassemethode*

Een methode die wordt aangeroepen zonder dat daarbij naar een bepaald object wordt verwezen. Methoden van klassen zijn van invloed op de klasse in haar geheel, maar niet op een bepaald exemplaar (afgeleide of kopie) van de klasse. Dit wordt ook wel een statische methode genoemd.

class object *object van een klasse*

Een object fungeert als een klasse. Een object van een klasse fungeert als een fabriek. Zie fabriek (factory).

class object-oriented library: COOL *COOL bibliotheek*

Een bibliotheek van Texas Instruments met klassen voor C++.

class path *klassepad*

Een klassepad is een omgevingsvariabele die de Virtuele Java Machine en andere Java-applicaties een aanwijzing geeft waar de klassebibliotheeken te vinden zijn. Die klassebibliotheeken omvatten eveneens de door de gebruiker gedefinieerde klassebibliotheeken.

class-relation method *klasse-relatiemethode*

Een ontwerpstechniek die gebaseerd is op de concepten van het object-georiënteerde programmeren en het entiteit-relatiemodel van het Franse bedrijf Softeam.

class variable *klassevariabele*

Een gegevensitem dat met een bepaalde klasse in haar geheel is geassocieerd, maar niet met bepaalde exemplaren (afgeleiden of

kopieën) van de klasse. Klassevariabelen worden in klassedefinities gedefinieerd. Dit wordt ook een statisch veld genoemd.

clause *clausule*

1. Een tekstdeel in een programma dat een specifieke functie heeft zoals de IF-THEN-ELSE bewerking.
2. Een reeks samenhangende woorden die tot doel heeft een kenmerk van een ingang te specificeren. Er zijn, bijvoorbeeld, in de programmeertaal Cobol gegevens-, verwerkingsomgevings- en bestandsclausules.

cleanroom *schone kamer*

Een software-ontwikkelingsbenadering waarmee het aantal fouten in de ontwikkelde software kan worden geminimaliseerd.

clear (to) *wissen*

Een variabele, register, of andere geheugen- of opslagplaats op nul, leeg, of enig andere nulwaarde stellen.

client 1. *client*; 2. *cliënt*

- a. In het client/server model dat bij communicatietoepassingen wordt gebruikt, is de client een proces dat op afstand toegang kan krijgen tot de systeemelementen en hulpmiddelen van een server, zoals de verwerkingsmogelijkheden en de capaciteit van de geheugens.
- b. In het X Window systeem en de afgeleiden daarvan, zoals Motif en Open Look, is de client een programma dat al het werk verricht. Dit in tegenstelling tot een programma dat de resultaten van die werkzaamheden op een beeldscherm weergeeft.
- c. Een systeem of een proces dat een dienst (service) van een ander systeem of proces nodig heeft en daarom verzoekt.

client object *clientobject*

Een object dat een verzoek voor een bepaalde vorm van dienstverlening indient. Een bepaald object kan een client voor bepaalde verzoeken zijn en tegelijkertijd als een server voor andere verzoeken fungeren.

client/server architecture *client/server-architectuur*

Een systeem (zoals X Window of een communicatiesysteem) dat een programma (de client) in staat stelt op een specifieke machine te draaien terwijl het programma dat de resultaten weergeeft (de server) op een andere machine wordt verwerkt.

closed loop 1. *gesloten kring(loop)*; 2. *gesloten lus*

Een kringloop of lus die geen uitgang heeft en waarvan de uitvoering slechts door een ingreep van buitenaf, dat wil zeggen van een plaats of positie buiten het computerprogramma of procedure

waar de gesloten kringloop of lus deel van uitmaakt, kan worden onderbroken.

closed subroutine *gesloten subroutine*

1. Een subroutine die in een gegeven locatie is geplaatst in plaats van dat deze op iedere plaats waar de subroutine in een programma wordt opgeroepen in dat programma wordt gekopieerd.
2. Een module die de mogelijkheid bezit om terug te kunnen keren naar het hoofdmodule. Dit is afhankelijk van de condities die bij het oproepen van dit module gelden.

closed system *gesloten systeem*

Een systeem waarin de systeemelementen beschermd zijn. Het systeem kan niet willekeurig of eigenmachtig van buitenaf worden benaderd.

code *code*

1. Een aantal regels dat op ondubbelzinnige wijze aangeeft in welke vorm gegevens moeten worden weergegeven en/of geïnterpreteerd.
2. Eén of meer programma's of modules.
3. Een serie overeengekomen letter- en/of cijfergroepen zoals afkortingen die elk een woord, begrip, opdracht, zinsdeel enzovoort betekenen.
4. Computerinstructies en gegevensdefinities die in een programmeertaal of in een bepaalde uitvoervorm van een assembleerprogramma, compiler of enig andere vertaler worden uitgedrukt.
5. Een teken of bitpatroon waar een bepaalde betekenis aan toe is gekend, zoals een statuscode.

code (to) *coderen*

1. Het samenstellen of schrijven van een programma.
2. Het weergeven van gegevens of een programma in een symbolische vorm en wel zodanig dat deze door een computer kunnen worden verwerkt.
3. Het uitdrukken of opstellen van een computerprogramma in een programmeertaal.

code base *codebasis*

Een codebasis werkt tezamen met het code-attribuut in een applet; het is een identificatie-aanhangsel (tag) dat een volledige specificatie bevat van de plaats waar het klassebestand van het hoofd-applet kan worden gevonden: de code specificeert de naam van het bestand en de codebasis de URL (Uniform Resource Locator) van de directory die het bestand bevat.

code-based testing *code-georiënteerd testen*

Het ontwikkelen van testen op basis van doelstellingen die uit de implementatie kunnen worden afgeleid, bijvoorbeeld, testen die

bepaalde besturingsstromen uitvoeren of specifieke gegevensitems gebruiken.

code breakpoint *codebreekpunt*

Een breekpunt dat bij de uitvoering van een bepaalde computerinstructie wordt geïnitieerd.

CodeCenter *CodeCenter*

Een privé software-ontwikkelingsomgeving voor C-programma's die een geïntegreerde gereedschapskist voor de ontwikkeling, het testen, het opsporen van fouten en het onderhoud van de software bevat.

code coverage *codedekking*

Een analysemethode waarmee kan worden bepaald welke delen van de software door de testen of testseries zijn uitgevoerd (afgedekt) en welke delen nog niet zijn uitgevoerd en daarom extra aandacht behoeven.

code generator *codegenerator*

1. Synoniem voor broncodegenerator.
2. In de compilertechnologie verwijst de term naar het laatste deel van een compiler waarin de doelcode of de assembleercode wordt gegenereerd.
3. Een routine die vaak een deel van een compilatieprogramma is en die een computerprogramma vanuit een bepaalde tussenliggend interpretatieniveau in een vorm vertaalt die dicht bij de taal van de machine ligt waarin het programma zal worden uitgevoerd.
4. Een softwaregereedschap dat als invoergegevens de eisen of het ontwerp van een computerprogramma accepteert en een broncode produceert die de eisen of het ontwerp implementeert.

code management *codebeheer*

Een broncodebeheerssysteem helpt programma-ontwikkelaars bij het volgen van de historie betreffende het ontstaan van de software, de vrijgave ervan, de uitgifte van parallelle versies enzovoort.

code of ethics standard *standaard-gedragscode*

Een standaard die de kenmerken beschrijft van een aantal morele principes die samenhangen met geaccepteerde standaarden betreffende het gedrag door en tussen professionele beroepsbeoefenaren.

code review *codebeoordeling*

Een bijeenkomst waarin de code van de software voor commentaar of goedkeuring wordt voorgelegd aan projectmedewerkers, managers, gebruikers, klanten, of andere betrokken partijen.

coding *codering*

1. Het proces van het schrijven of opstellen van een computerprogramma in een programmeertaal.
2. Het omzetten van logica en gegevens uitgaande van ontwerpspecificaties (ontwerpbeschrijvingen) in een programmeertaal.

cognitive science *cognitiewetenschap*

Een tak van de wetenschap waarin elementen uit de informatica, psychologie, onderwijskunde, linguïstiek, filosofie en wiskunde samenkomen.

COGO *COGO*

Een programmeertaal die is ontwikkeld voor het oplossen van problemen waarin de meetkunde en de verwerking van coördinaatgegevens van doorslaggevende betekenis zijn. De taal wordt hoofdzakelijk in de civiele technieken toegepast.

cohesion *cohesie*

1. De wijze waarop en de mate waarin de taken die door een softwaremodule of asset worden uitgevoerd aan die van respectievelijk andere softwaremodules of assets zijn gerelateerd. Bestaande cohesie-typering omvatten die welke coïncidenteel, communicatief, functioneel, logisch, procedureel, sequentieel, en temporeel van aard zijn.
2. De mate waarin de structuur van een asset de functie ervan ondersteunt.
3. De mate waarin van functionele relaties van de verwerkingselementen in een module sprake is. Een hoog cohesieniveau in een module of asset kan het hergebruik ten goede komen.

coincidental cohesion *coïncidentele cohesie*

Een soort cohesie waarin de taken die door een softwaremodule of asset worden uitgevoerd geen functionele relatie met die van andere softwaremodules of assets heeft.

collaboration diagrams *samenwerkingsdiagrammen*

Diagrammen die de berichtenstroom tussen objecten in een objectgeoriënteerde applicatie weergeven en de inhoud van de fundamentele associaties tussen de objecten tonen.

combined object-oriented language: COOL *COOL taal*

Taal ontwikkeld in het ITHACA ESPRIT-project dat op C-gebaseerde talen combineert met databasetechnieken.

command *1. commando; 2. opdracht*

- a. Een besturingssignaal.
- b. Een wiskundig of logisch bewerkingsteken.
- c. Een verzoek voor het verrichten van een bewerking of het doen uitvoeren van een programma.

- d. Een tekenrij afkomstig van een bron buiten het systeem die een verzoek voor het uitvoeren van een bepaalde systeemactie inhoudt.
- e. Een instructie in een machinetaal.
- f. Een woord dat, nadat het is ingetypt, Unix een bewerking of actie uit laat voeren. De Unix shell reageert op de commando's die worden ingetypt en tracht deze uit te voeren. Van bepaalde commando's weet de shell precies wat er moet worden gedaan. Andere commando's verwijzen naar aparte programma's die in bestanden in het geheugen of op schijven zijn opgeslagen. Nadat een commando is ingetypt dient de gebruiker de 'enter' of 'return' toets in te drukken om het commando te activeren.
- g. Besturingsgegevens die van het primaire station naar de secundaire stations in een netwerk worden verzonden.
- h. Een uitdrukking die in een computersysteem kan worden ingevoerd om een actie te initiëren of de uitvoering van een computerprogramma te beïnvloeden.

command-driven *opdracht-gestuurd*

De term heeft betrekking op een systeem of werkwijze waarin de gebruiker het systeem met behulp van opdrachten stuurt.

command language *besturingstaal*

- 1. Een taal die wordt gebruikt om er opdrachten voor een computersysteem in uit te drukken.
- 2. Een brontaal die voornamelijk uit procedure-bewerkingstekens bestaat. Deze bewerkingstekens kunnen elk een bepaalde uit te voeren functie initiëren.

command processing program *opdrachtverwerkingsprogramma*

Een programma dat een opdracht verwerkt. Dit programma voert de opdracht uit zodat de gewenste functie wordt gerealiseerd.

command statement *besturingsopdracht*

Een opdracht die wordt gebruikt om via de invoerstream opdrachten aan het systeem te verstrekken.

comment *commentaar*

- 1. Een beschrijving, verwijzing of verklaring die aan de tekst van een programma wordt toegevoegd.
- 2. Een uitvoertaal die geen uitwerking heeft op de doeltaal.
- 3. Een opdracht die wordt gebruikt om een programma te documenteren. De opdracht bevat gegevens die van nut kunnen zijn bij het documenteren of controleren van een lijst met uitvoergegevens.
- 4. Verklarende tekst die in een programma voor kan komen en die door de compiler of het vertolkingsprogramma wordt genegeerd. Zo wordt, bijvoorbeeld, in bepaalde programma's de verklarende

tekst van de overige informatie in het programma gescheiden door twee schuine strepen (//) of de combinatie /* en */.

5. Informatie die (midden) in een computerprogramma, job-control opdrachten, of een serie gegevens zijn geplaatst en die de lezers uitleg verschaft maar de machine-interpretaties niet beïnvloedt.

comment field *commentaarveld*

Het gedeelte van een instructie dat wordt gebruikt om een toelichting bij die instructie te geven. Dit commentaar wordt door het compilatie-, interpretatie- of assembleerprogramma genegeerd.

commonality *1. gemeenschappelijkheid; 2. algemeenheid*

1. In domeinanalyse verwijst de term naar een asset of een duidelijk onderscheiden deel, functie, of kenmerkende eigenschap die karakteristiek is voor de klasse van systemen binnen een domein en die in een domeinmodel of domeinarchitectuur is/wordt gerepresenteerd.
2. Een maat die aangeeft hoe vaak het probleem dat met een zeker systeem moet worden opgelost, voorkomt; dus hoe algemeen een bepaald probleem in een zeker systeem is. Dit is één van de factoren in de generaliteit of de herbruikbaarheid van programma's.

common application environment: CAE *gemeenschappelijke applicatieomgeving*

Een applicatieomgeving van X/Open die op POSIX en C is gebaseerd.

common business oriented language: COBOL *COBOL*

Een programmeertaal die speciaal is ontwikkeld voor administratieve en bedrijfsgerichte toepassingen.

common communication services: CCS *gemeenschappelijke communicatiediensten*

De standaard programma-interfaces in IBM's SAA (System Application Architecture).

common data format: CDF *gemeenschappelijk gegevensformaat*

Een bibliotheek en gereedschapskist voor multi-dimensionale gegevensverzamelingen.

common-environment coupling *gemeenschappelijke omgevingskoppeling*

Een soort koppeling waarbij twee softwaremodules een gemeenschappelijk gegevensgebied delen.

common facilities *gemeenschappelijke faciliteiten*

Faciliteiten die in veel applicatiedomeinen nuttig en bruikbaar zijn en die middels klasse-interfaces beschikbaar zijn.

common gateway interface: CGI *gemeenschappelijke communicatiepoort-interface*

Een standaard waarin is vastgelegd hoe externe programma's in het wereldwijde web of een gelijksoortige informatieserver kunnen werken.

Common Lisp object system: CLOS *CLOS*

Een object-georiënteerde taal die van Common Lisp is afgeleid.

common object model: COM *gemeenschappelijk objectmodel*

Een open architectuur van DEC en Microsoft. Het zorgt voor de mogelijkheid ObjectBroker en OLE gemeenschappelijke activiteiten en verwerkingen uit te laten voeren.

common object request broker architecture: CORBA *architectuur van een gemeenschappelijke bemiddelingsfunctie*

Een architectuur die objecten in staat stelt met andere objecten te communiceren, ongeacht de taal waarin ze zijn geschreven en ongeacht het besturingssysteem waaronder ze worden gebruikt. CORBA werd ontwikkeld door een industrieel consortium, de OMG (Object Management Group).

common open software environment: COSE *gemeenschappelijk open software-omgeving*

Een initiatief van Hewlett-Packard, Sun, IBM, Novell, Univel en SCO (Santa Cruz Operation) dat zich ten doel stelt de consistentie en onderlinge samenwerking tussen de leveranciers van Unix-systemen te verbeteren.

common storage *gemeenschappelijk geheugengebied*

Een deel van het hoofdgeheugen dat voor twee of meer modules in een softwaresysteem toegankelijk is.

communicational cohesion *communicatieve cohesie*

Een soort cohesie waarbij de taken die door een softwaremodule worden uitgevoerd dezelfde invoergegevens gebruikt of aan de productie van dezelfde uitvoer bijdraagt.

compare (to) *vergelijken*

Het onderzoeken van twee gegevens om hun relatieve grootte of positie in een reeks te kunnen bepalen of om na te kunnen gaan of zij overeenkomstige kenmerken of eigenschappen bezitten.

compaction *compactheid*

De term verwijst in de microprogrammering naar het proces waarin een microprogramma naar een functioneel equivalent microprogramma wordt geconverteerd dat sneller of korter is dan het originele microprogramma.

comparison *vergelijking*

Het onderzoek van de relatie tussen twee of meer gegevens met betrekking tot identiteit, overeenkomst, gelijkheid, relatieve grootte of volgorde in een reeks.

comparison definition *vergelijkingsdefinitie*

Twee opdrachtelelementen die van elkaar zijn gescheiden door een vergelijkingsbewerkingsteken dat in een IF-uitdrukking voorkomt.

comparison operator *vergelijkingsbewerkingsteken*

Speciale bewerkingstekens (gelijk aan, groter dan, kleiner dan enzovoort) die bij vergelijkingen worden gehanteerd.

comparator *vergelijker*

Een softwaregereedschap dat twee computerprogramma's, bestanden, of gegevensverzamelingen met elkaar vergelijkt voor de identificatie van gemeenschappelijkheden of verschillen. Typische objecten die worden vergeleken omvatten gelijksoortige versies van de broncode, objectcode, databasebestanden of testresultaten.

compatibility *compatibiliteit*

1. De mogelijkheid die twee of meer systemen bezitten om hun functies uit te voeren terwijl ze dezelfde hardware- of software-omgeving delen.
2. De mogelijkheid die twee of meer systemen of componenten hebben om onderling informatie uit te kunnen wisselen.

compatibility testing *testen van de compatibiliteit*

Het uitvoeren van testen om na te gaan of een systeem compatibel is met andere systemen waarmee het moet kunnen communiceren.

compatible *compatibel*

Een term die betrekking heeft op een programma of een systeem dat op computers van verschillende makelij kan worden verwerkt of op een computer die dezelfde programma's kan verwerken als een andere computer.

compilation *compilatie*

De vertaling van een bronprogramma in een uitvoerbaar programma (doelprogramma, machineprogramma of werkprogramma).

compilation unit *compilatie-eenheid*

De kleinste eenheid broncode die kan worden gecompileerd. In de huidige implementatie van Java is die eenheid een bestand.

compile (to) *compileren*

1. Het vertalen van een computerprogramma dat in een hogere-orde taal is geschreven, in de equivalente machinetaal.
2. Het vertalen van een programma dat in een probleemgerichte taal is geschreven, in een machinetaal.
3. Het vervaardigen van de machinetaal van een programma dat in een andere programmeertaal is geschreven en waarbij gebruik wordt gemaakt van de logische structuur van dat programma.

compile-and-go *compileren en uitvoeren*

Een werkwijze waarin geen pauzes of onderbrekingen tussen het compileren, het monteren (het linken), het laden en de uitvoering van een computerprogramma voorkomen.

compiler *compilatieprogramma*

Een programma dat programma's die in een hogere-orde taal zijn geschreven of uit broncode bestaan, vertaalt naar een equivalente code die door een computer kan worden begrepen en uitgevoerd. Zo vertaalt, bijvoorbeeld, een Java-compiler de Java-broncode in Java-bytecode.

compiler code *compilercode*

Computerinstructies en gegevensdefinities die in een vorm zijn uitgedrukt die door een compiler kan worden herkend en verwerkt.

compiler generator *compilergenerator*

Een vertaal- of vertolkingsprogramma dat kan worden gebruikt om een deel van een compiler of een complete compiler te construeren.

completion code *voltooiingscode*

Een code die door een programma dat groepsgewijs wordt verwerkt, wordt overgedragen aan een bewerkingseenheid die een jobstroom verwerkt. De code wordt gebruikt om de uitvoering van de opeenvolgende stappen in de stroom invoergegevens te beïnvloeden.

complex instruction set computer: CISC *computer met een complex instructieset*

Een soort CPU waarvan de instructies redelijk gecompliceerd zijn ten opzichte van een computer met een gereduceerd instructieset (RISC)

complexity *complexiteit*

1. De moeilijkheidsgraad van een systeem of een component (vanwege het ontwerp of de implementatie) voor wat betreft het kunnen begrijpen en verifiëren van dat systeem of component.
2. De term verwijst naar een structuur-gebaseerde maatleer waarmee de attributen van punt 1 van deze definitie kunnen worden gemeten.

component *component*

1. Een minimaal klein software-item waarvan een afzonderlijke specificatie beschikbaar is.
2. Eén van de vele delen die een systeem vormen. Een component kan zowel een hardware- als softwarecomponent zijn en kan worden onderverdeeld in weer andere componenten. Deze term wordt in dit verband ook als een synoniem van de termen *module*, *eenheid*, of *asset* gebruikt.

component diagram *componentschema*

Een schema dat de herbruikbare softwarecomponenten met hun interfaces en onderlinge relaties toont.

component standard *componentstandaard*

Een standaard die de kenmerken en eigenschappen van gegevens of componenten van programma's beschrijft.

component testing *testen van een component*

Het testen van individuele hardware- of softwarecomponenten of groepen onderling aan elkaar gerelateerde componenten.

composite type *samengesteld type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens uit meervoudige gegevensitems bestaan. Zo is, bijvoorbeeld, PAIR het gegevenstype van een gegeven dat uit een geordend paar (x,y) bestaat.

compound document architecture: CDA *gecombineerde documentarchitectuur*

Een verzameling standaarden van DEC voor de generatie, de opslag, het terugzoeken en ophalen, de uitwisseling, en de manipulatie van gecombineerde en/of samengestelde documenten.

compound object *samengesteld object*

Een object dat uit een verzameling onderling gerelateerde objecten bestaat.

compress (to) *comprimeren*

1. Bij een tekenrij verwijst deze term naar het reduceren van de ruimte die op een gegevensmedium door zich herhalende tekens wordt ingenomen.
2. Het besparen van geheugenruimte door de eliminatie van tussenruimten, lege velden, redundante of niet-noodzakelijke gegevens om de lengte van de records of bestanden te verkleinen.
3. Het verplaatsen van bestanden en bibliotheken op een schijvengeheugen om op die wijze een aaneengesloten gebied van niet-gebruikte geheugenplaatsen te verkrijgen.

compression *compressie*

Een manier waarmee bestanden kunnen worden verkleind (ingekrompen of gecomprimeerd). Het doel hiervan is om ruimte te besparen. Bestandcompressieprogramma's bevatten compressie- en verpakkingsfuncties (compress and pack). Om het oorspronkelijke bestand weer te verkrijgen, dient men gebruik te maken van decompressie- en uitpakfuncties (decompress and unpack). Veel voorkomende voorbeelden van bestanden die worden gecomprimeerd zijn bestanden die grafische informatie bevatten, alsmede uitvoerbare programmatuur. Er bestaat een overvloed aan utiliteitsprogramma's die deze functies verzorgen.

computer 1. *computer*; 2. *rekenmachine*

Een elektronische eenheid of elektronisch apparaat dat in principe uit een verwerkingseenheid, een geheugen en in- en uitvoerapparatuur bestaat. De bewerkingen die in de verwerkingseenheid worden uitgevoerd, worden bepaald door een programma dat in het geheugen is opgeslagen.

computer aided detector design: CADD *computer ondersteund detector ontwerp*

Een project waarin standaarden en methoden worden ontwikkeld waarmee de ontwerpers van HEP-detectors (instrumenten die deeltjes met hoge energie detecteren) die in verschillende instituten werkzaam zijn, hun ideeën, informatie en ontwerpgegevens onderling uit kunnen wisselen.

computer aided software engineering: CASE *computer ondersteunde software-ontwikkeling*

Een techniek waarbij computers worden gebruikt om de systematische analyse, ontwikkeling, implementatie en het onderhoud van software te ondersteunen. Het gebruik van een CASE-benadering voor de constructie en het onderhoud van systemen betekent dat softwaregereedschappen bij de ontwikkeling van software worden toegepast en dat de ontwikkelaars die deze gereedschappen moeten hanteren, in het gebruik daarvan worden opgeleid.

computer aided test engineering: CATE *computer ondersteunde testontwikkeling*

De term heeft betrekking op CASE-methoden die worden gebruikt voor het testen van elektronica.

computer graphics metafile: CGM *metafile voor de computergrafiek*
Een standaard bestandsformaat voor de opslag en de communicatie van grafische informatie. Dit formaat wordt op grote schaal in PC's gebruikt en wordt geaccepteerd in bureau-publicatiesystemen.

computer instruction *computerinstructie*

1. Een machine-instructie die door de computer kan worden herkend en uitgevoerd.
2. Een instructie of opdracht in een programmeertaal waarin de bewerking die door een computer moet worden uitgevoerd en de adressen of waarden van de daarbij behorende operanden worden gespecificeerd.
3. Elke uitvoerbare instructie of opdracht in een computerprogramma.

computer instruction code *computerinstructiecode*

Een code die wordt gebruikt om instructies te representeren.

computer instruction set *serie computerinstructies*

Een complete serie instructies voor een bepaalde computer. Ze gaan vergezeld van een beschrijving van de betekenis van de bijbehorende opdrachtelelementen.

Computer Integration Laboratories: CIL *CIL*

Een consortium van Apple Computers, IBM, Novell, Oracle, Talingent, WordPerfect en Xerox waarin getracht wordt een gemeenschappelijk raamwerk te creëren voor de onderlinge koppeling en het uitwisselen van gegevens en programma's tussen, en de gedistribueerde verwerking van bureau-computers.

computer language *computertaal*

Een taal die de mens in staat stelt met computers te communiceren.

computer oriented language *computergerichte taal*

1. Een programmeertaal die een weerspiegeling is van de structuur van een bepaalde computer of bepaald type computer.
2. Een programmeertaal waarvan de woorden en de syntaxis voor een bepaald soort computer zijn ontworpen.

computer performance evaluation *evaluatie van de prestaties van computers*

Een ontwikkelingsdiscipline waarin de prestaties van computersystemen worden gemeten en waarin methoden waarmee die prestaties kunnen worden verbeterd, worden onderzocht.

computer program *computerprogramma*

1. Een reeks instructies die door een computer kan worden verwerkt. Deze verwerking kan het gebruik van een assembleertaal, een compileertaal, een vertolker (te interpreteren taal) of een vertaalprogramma omvatten om het programma gereed te maken en uit te voeren.
2. Een combinatie van computerinstructies en gegevensdefinities met behulp waarvan computerapparatuur rekenkundige functies of besturingsfuncties uit kan voeren.

computer program abstract *computerprogramma-resumé*

Een korte beschrijving van een computerprogramma die voldoende informatie voor potentiële gebruikers bevat en waarmee die gebruikers de geschiktheid van dat programma voor hun behoeften in kunnen schatten.

computer program component: CPC *computerprogrammacomponent*

Zie computersoftwarecomponent.

computer program configuration item: CPCI *computerprogrammaconfiguratie-item*

Zie computersoftwareconfiguratie-item.

computer resource allocation *allocatie van de computerelementen*

De toekenning van de computerelementen aan in uitvoering zijnde en wachtende karweien; bijvoorbeeld, de toekenning van het hoofdgeheugen, invoer/uitvoereenheden, en hulpgeheugen voor karweien die gelijktijdig in een computersysteem worden verwerkt.

computer resources *computerelementen*

De computerapparatuur, programma's, documentatie, diensten, faciliteiten, voorzieningen, onderdelen, en personeel die/dat voor een bepaald doel zijn/is bestemd en/of vrijgemaakt.

computer science *computerwetenschappen*

De studie van computers, hun principes en het gebruik ervan. Computerwetenschappen omvatten onderwerpen zoals: compilers en besturingssystemen, computernetwerken en interfaces, de architectuur van computersystemen, hardware-ontwerp en -testen, informatiestructuren, programmeertalen, programmeren, software-ontwikkeling, systeemanalyse en -ontwerp, theorieën betreffende informatiesystemen en berekeningen, en applicaties uit de sociale, politieke, en educatieve wetenschappen, alsmede die uit de bedrijfswetenschappen en de engineering.

computer software component: CSC *computersoftwarecomponent*

Een functioneel of logisch duidelijk te herkennen onderdeel van een computersoftwareconfiguratie-item. Dit is over het algemeen een samenvoeging of combinatie van twee of meer software-eenheden.

computer software configuration item: CSCI *computersoftwareconfiguratie-item*

Een aggregatie (combinatie of samenvoeging) van software die is aangewezen om deel uit te gaan maken van het configuratiebeheer en als een enkelvoudige entiteit in het configuratiebeheersproces wordt behandeld.

computer supported cooperative work: CSCW *computer ondersteunde coöperatieve werkzaamheden*

De term staat ook bekend onder de naam 'groupware'. CSCW bestaat uit methoden, softwaregereedschappen en technieken die groepen ontwikkelaars, die gezamenlijk aan een bepaald project werken en dat meestal op verschillende locaties doen, ondersteunt.

computer system *computersysteem*

Een systeem dat uit één of meer computers en daarmee geassocieerde software bestaat.

computing center *rekencentrum*

Een faciliteit die is aangewezen voor het leveren van computerdiensten aan een grote verscheidenheid aan gebruikers. Dit geschiedt met behulp van computers en andere ondersteunende apparatuur en de diensten en dienstverlening die door de staf van het rekencentrum worden verzorgd.

concept phase *conceptfase*

1. De fase in de cyclus van de software-ontwikkeling gedurende welke de eisen en behoeften van de gebruikers worden beschreven en aan de hand van documentatie worden geëvalueerd. Die documentatie kan, bijvoorbeeld, bestaan uit een behoefteverklaring, planningrapporten, een projectinitiatiememorandum, haalbaarheidsstudies, systeemdefinitie, procedures, regulerende wetten of beleidsvoornemens of -beslissingen die voor het project van belang zijn.
2. De initiële fase van een software-ontwikkelingsproject waarin de eisen en behoeften van de gebruiker aan de hand van een uitgebreide documentatie worden geëvalueerd. Zie voor die documentatie punt 1 van deze definitie.

concrete class *concrete klasse*

Een klasse waarvan exemplaren, afgeleiden of kopieën kunnen worden gemaakt, dat wil zeggen een klasse die kan worden geïnstantieerd. Dit in tegenstelling tot een abstracte klasse.

concurrency *gelijktijdigheid*

De mogelijkheid van een systeem of proces gelijktijdig bewerkingen op meerdere objecten uit te kunnen voeren.

concurrent *gelijktijdig*

De term heeft betrekking op het optreden of voorkomen van twee of meer activiteiten die binnen hetzelfde tijdsinterval moeten worden uitgevoerd. Verwerking van die activiteiten wordt gerealiseerd door de activiteiten simultaan of afwisselend uit te laten voeren.

Concurrent Clean *Concurrent Clean*

Een functionele taal die geschikt is voor de Macintosh computers en ontwikkeld is aan de Universiteit van Nijmegen.

concurrent programming structures *gelijktijdig aanwezige programmastructuren*

De term verwijst naar de situatie waarin de semantiek van een object in de aanwezigheid van meervoudige besturingscomponenten, -functies, of -modules (onderlinge combinaties van besturende functies) bewaard blijft en waarbij de wederzijdse uitsluiting door het object zelf wordt verzorgd en bekrachtigd.

condition *conditie*

1. Eén van een reeks specifieke waarden die een gegeven aan kan nemen.
2. Een booleaanse uitdrukking die geen booleaanse bewerkingfuncties (operatoren) bevat.

conditional construction *conditionele constructie*

Een taalconstructie die een aantal verschillende uitvoeringsvolgorden specificiert, bijvoorbeeld een CASE-clausule, een IF-uitdrukking, en conditionele uitdrukkingen zoals in de programmeertalen Algol en Pascal voorkomen.

conditional expression *conditionele uitdrukking*

Een logische opdracht die bij twee bestanddelen de verwantschap tussen die bestanddelen beschrijft, zoals: groter dan, gelijk aan, kleiner dan enzovoort.

conditional jump *conditionele sprong*

Een sprong die slechts plaats vindt indien de instructie die deze sprong specificiert, wordt uitgevoerd en waarbij tevens aan bepaalde voorwaarden wordt voldaan.

conditional statement *conditionele uitdrukking*

1. Een opdracht die het uitvoeren van één uit een aantal mogelijke bewerkingen mogelijk maakt. Daarbij kan de besturing al dan

niet worden overgedragen; bijvoorbeeld een berekende GOTO in een Fortran-programma of een IF-opdracht in een Pascal-programma.

2. Een opdracht die wordt gebruikt om een toewijzing of een sprong uit te drukken die gebaseerd is op bepaalde criteria zoals het geval is in een IF-THEN bewerking.
3. Een instructie die uit gegevensnamen, lettersymbolen, figuurlijke constanten, logische bewerkingstekens, of een combinatie van dergelijke bewerkingstekens bestaat en die zodanig is samengesteld dat kan worden gecontroleerd of de evaluatie van de instructie 'waar' of 'onwaar' is. De hier op volgende actie van het programma is hiervan afhankelijk.

condition name *conditiennaam*

1. De naam die aan een bepaalde waarde of serie waarden, die een gegeven kan aannemen, wordt toegekend.
2. Een sleutelwoord in de programmeertaal PL/1, dat een aanconditie die binnen een taak kan ontstaan, aanduidt.

condition outcome *resultaat van een conditie*

De evaluatie van een conditie die tot de waarden 'waar' of 'onwaar' (TRUE of FALSE) leidt.

configuration *configuratie*

1. De samenstelling van apparatuur en programmatuur van een computersysteem, component en/of netwerk zoals dat door de aard, het aantal, en de voornaamste kenmerken van de functionele eenheden wordt bepaald.
2. De apparatuur en programmatuur die een systeem, deelsysteem, of netwerk definiëren.
3. De functionele en fysieke eigenschappen en kenmerken van hardware en software zoals dat in de technische documentatie is vastgelegd of in een product wordt gerealiseerd.

configuration control *configuratiebesturing*

Een onderdeel van het configuratiebeheer dat de volgende activiteiten omvat: evaluatie, coördinatie, goedkeuring of afkeuring, en implementatie van wijzigingen op configuratie-items nadat de formele vastlegging van hun configuratie-identificaties heeft plaatsgevonden.

configuration control board: CCB *configuratie-stuurgroep*

Een groep mensen verantwoordelijk voor de evaluatie en het goedkeuren of afkeuren van voorgestelde wijzigingen op configuratie-items. De stuurgroep ziet er verder op toe dat de implementatie van de goedgekeurde wijzigingen correct wordt uitgevoerd.

configuration identification *configuratie-identificatie*

1. Een onderdeel van het configuratiebeheer dat uit de selectie van de configuratie-items voor een systeem bestaat en dat hun functionele en fysieke kenmerken en eigenschappen in de technische documentatie vastlegt.
2. De momentane goedgekeurde technische documentatie van een configuratie-item zoals dat in specificaties, tekeningen, daaraan verwante rapporten en documenten, waarnaar in deze documenten wordt verwezen, is vastgelegd.

configuration index *configuratie-index*

Een document dat wordt gebruikt in configuratiebeheer en dat een opsomming bevat van de configuratie-items waaruit een product is opgebouwd of samengesteld.

configuration item: CI *configuratie-item*

1. Een aggregatie van hardware of software die aan een bepaalde functie voldoet en kan worden aangemerkt als deel uit te moeten maken van een configuratiebeheerssysteem.
2. Een digitaal gegevensbestand dat deel uitmaakt (of deel uit moet maken) van een configuratiebeheerssysteem.
3. Een aggregatie van hardware en/of software die deel uitmaakt van een configuratiebeheerssysteem en als een enkelvoudige entiteit in het configuratiebeheersproces moet worden behandeld.

configuration item development record *ontwikkelingsrapport van een configuratie-item*

Een document dat in het configuratiebeheer wordt gebruikt en dat de status van de ontwikkeling van een configuratie-item bevat. Die status is gebaseerd op de resultaten van configuratie-audits en ontwerpbeoordelingen.

configuration management: CM *configuratiebeheer*

1. Het proces waarin de configuratie-items in een systeem en in wijzigingsverzoeken worden geïdentificeerd, gedefinieerd, vastgelegd, en gedocumenteerd.
2. Het proces waarin de technische en administratieve begeleiding en het toezicht betreffende de levenscyclus van configuratie-items wordt uitgevoerd. Dit omvat: de unieke identificatie van configuratie-items met inbegrip van hun versies en stand van zaken; de identificatie en documentatie van de functionele en fysieke kenmerken van CI's; wijzigingen in het beheer van en de controle op CI's en de daaraan verwante documentaties; het vastleggen van en het rapporteren over informatie die nodig is om de CI's te beheren, met inbegrip van het vrijgeven van voorgestelde wijzigingen en de stand van zaken betreffende implementaties van vrijgegeven wijzigingen; en het controleren (verzorgen van audits) van CI's ten behoeve van de verificatie

van de mate waarin aan specificaties wordt voldaan. Bij het hergebruik van software omvat dit eveneens het proces van de vastlegging en de classificatie van alle assets; het vastleggen van en de controle op het vrijgeven en het wijzigen van deze assets gedurende de gehele levenscyclus van het systeem waarin ze worden gebruikt; het vastleggen van en het rapporteren over de status van de assets en de wijzigingsverzoeken; en de verificatie van de volledigheid en de juistheid van de assets.

configuration status accounting *verantwoording van de status van de configuratie*

Een onderdeel van het configuratiebeheer dat uit de vastlegging en rapportage van informatie bestaat die nodig is om een configuratie op effectieve wijze te beheren. Deze informatie omvat een lijst van alle goedgekeurde configuratie-identificaties, de status van de voorgestelde wijzigingen op de configuratie, en de status van de implementatie van de goedgekeurde wijzigingen.

configure (to) *configureren*

Het beschrijven van een systeem voor wat betreft de eenheden, voorzieningen, programma's, ondersteunende apparatuur, bedieningsmogelijkheden enzovoort.

conformance *conformantie*

Een relatie die tussen bepaalde typen is gedefinieerd en wel zodanig dat type x zich conformeert aan type y indien elke waarde die van toepassing is op type x eveneens van toepassing is op type y.

conformance criterion *conformantiecriterium*

Een bepaalde methode waarmee kan worden beoordeeld of de actie van een component op een zekere gespecificeerde invoerwaarde al dan niet conform de specificatie is.

conformance testing *conformantietesten*

Het proces waarin wordt nagegaan of een implementatie zich conformeert aan de specificatie waarop deze is gebaseerd.

consecutive *opeenvolgend*

De term verwijst naar het voorkomen of optreden van twee opeenvolgende gebeurtenissen of items zonder dat die gebeurtenissen of items elkaar overlappen of elkaar beïnvloeden, dat wil zeggen de ene volgt onmiddellijk na de andere.

Conseil Européen pour la Normalisation: CEN *Europese Normalisatieraad*

Het Europese standaardisatie- en normalisatie-instituut. CEN werkt nauw samen met de ISO maar is in de eerste plaats verantwoordelijk

voor de ontwikkeling en standaardisatie van normen en normbladen die door de leden van CEN zijn voorgesteld en ontwikkeld. De leden van CEN zijn de nationale normalisatie-instituten van de landen die deel uitmaken van de Europese Unie.

consistency *consistentie*

De mate van uniformiteit en standaardisatie, alsmede afwezigheid van tegenspraken of conflicten die tussen de documenten of delen van een systeem of een component bestaan.

consistency check *consistentiecontrole*

Een controle waarbij wordt nagegaan of bepaalde gegevens compatibel zijn, of de samenhang van programma's en/of apparatuur deugdelijk is, of aan gespecificeerde verwantschappen en/of eigenschappen van eenheden of samenstellingen wordt voldaan enzovoort.

constant *constante*

1. In een computer is dit een getal dat aan de computer is toegevoegd en daarin voor herhaald gebruik beschikbaar is.
2. Een vaste of onveranderlijke waarde.

constant-failure period *periode met constant optredende fouten*

De tijdsperiode in de levenscyclus van een systeem of component waarin hardwarefouten elkaar met ongeveer gelijke tijdsintervallen opvolgen.

constraint *beperking*

1. In het hergebruik van software en herbruikbare bibliotheken verwijst de term naar een regel of een restrictie die het hergebruik van softwarecomponenten of assets regelt. Dat is speciaal van toepassing op de omgeving waarin deze worden gebruikt.
2. In de software-ontwikkeling heeft de term betrekking op een functionele of operationele eis die aan een softwaresysteem wordt gesteld. Een dergelijke eis limiteert mogelijke oplossingen.

constraints *beperkingen*

Functionele afhankelijkheden tussen objecten anders dan die welke van toepassing zijn op afhankelijkheden met betrekking tot in- en uitvoer.

constructive cost model: COCOMO *constructief kostenmodel*

Een methode waarmee de kosten van een softwarepakket kunnen worden geëvalueerd.

constructor *constructor*

1. Een functie in een klasse in C++ waarmee een object kan worden geïnstantieerd.
2. Een pseudo-methode waarmee een object kan worden gecreëerd. Constructors zijn in Java exemplaren van methoden met dezelfde naam als hun klasse. Java-instructors worden geactiveerd door het nieuwe sleutelwoord te gebruiken.

container class *containerklasse*

Een klasse waarvan de exemplaren verzamelingen van andere objecten zijn. Voorbeelden zijn onder andere stapels (stacks), wachtrijen (queues), lijsten (lists) en rijen (arrays).

content coupling *inhoudskoppeling*

Een soort koppeling waarin een bepaald deel of de gehele inhoud van een softwaremodule in de inhoud van een ander module wordt geplaatst of ondergebracht.

context *context*

1. De omstandigheden, de situatie, of de omgeving waarin een specifiek softwaresysteem of domein bestaat.
2. De academische discipline waarin een bepaalde term betekenis krijgt of heeft, bijvoorbeeld de disciplines van de software-ontwikkeling, de domeinanalyse, of de beslissingstheorie.

context analysis *contextanalyse*

Het proces waarin het bereik en de grenzen van een domein worden bepaald zoals dat is beschreven in de methode van de kenmerkgeoriënteerde domeinanalyse (feature-oriented domain analysis: FODA). Zie ook domeindefinitie. In het proces van de domeinontwikkeling neemt de contextanalyse dezelfde plaats in als de domeindefinitie, behalve dat in FODA de contextanalyse de analyse van de interactie tussen systemen in een domein en een domeinomgeving benadrukt.

contiguous allocation *aangrenzende toewijzing*

De techniek van een geheugentoe wijzing waarbij programma's of gegevens die moeten worden opgeslagen aan een geheugenpartitie van gelijke of grotere omvang worden toegewezen zodat programma's en gegevens die logisch aan elkaar grenzen, geheugengebieden krijgen toegewezen die fysiek aangrenzend zijn.

continuous iteration *continue iteratie*

Een lus die geen uitgang heeft.

control (to) *besturen*

Het bepalen van het tijdstip waarop en de volgorde waarin de verschillende delen van een computer de invoer-, verwerkings-, opslag-, en uitvoerfuncties moeten verrichten.

control code *besturingscode*

Een code van één byte en de betekenis van de daaraan toegekende besturingsfuncties.

control coupling *besturingskoppeling*

Een soort koppeling waarin een softwaremodule informatie met een ander softwaremodule uitwisselt met het expliciete doel de uitvoering van dat laatste module te beïnvloeden.

control data *besturingsgegevens*

Gegevens die de keuze van een werkwijze bepalen, de sequentiële stroom van een programma sturen, of de werking van software direct beïnvloeden, zoals dat het geval is met, bijvoorbeeld, een besturingsvariabele in een lus.

control field *besturingsveld*

1. Een veld dat met andere velden wordt vergeleken om de volgorde van de records te bepalen.
2. Een veld in een record dat het verband tussen het desbetreffende record en andere records definieert.
3. Een groep aangrenzende bits in een besturingswoord die wordt gebruikt om de volgorde van records bij sorteer- en samenvoegbewerkingen te bepalen.

control flow *besturingsstroom*

1. Een abstracte representatie van alle mogelijke opeenvolgende gebeurtenissen tijdens de uitvoering van een programma.
2. Bij programmeertalen is dit een abstractie van alle mogelijke verbindingswegen die de uitvoeringsvolgorde door een programma kan volgen. Deze besturingsstroom kan door een speciale grafiek worden voorgesteld.

control flow diagram *besturingsstroomschema*

Een schema dat de verzameling van alle mogelijke volgorden weergeeft waarin bewerkingen tijdens de uitvoering van een systeem of programma zouden kunnen worden uitgevoerd.

control flow path *pad van de besturingsstroom*

De grafische of schematische voorstelling van de mogelijke paden van alternatieve besturingsstromen door een component.

control language *besturingstaal*

Alle opdrachten waarmee bepaalde functies kunnen worden uitgevoerd.

control program *besturingsprogramma*

Een programma dat is ontworpen om de uitvoering van de programma's van een computersysteem te coördineren en te controleren.

control section *besturingssectie*

Het gedeelte van een programma dat door de programma-ontwikkelaar als een verplaatsbare eenheid wordt gedefinieerd en waarvan alle elementen in aangrenzende posities van het (hoofd)geheugen moeten worden geplaatst.

control statement *besturingsopdracht*

1. Een opdracht die de uitvoering van een programma in een computer beïnvloedt of bestuurt.
2. In programmeertalen verwijst de term naar een opdracht die wordt gebruikt om het voortdurend opeenvolgend uitvoeren van opdrachten te wijzigen. Een besturingsopdracht kan zowel een conditionele opdracht als een absolute opdracht (zoals een STOP-opdracht) zijn.
3. Een opdracht in een programma die een keuze maakt uit alternatieve series programma-instructies of die van invloed is op de volgorde waarin bewerkingen worden uitgevoerd, zoals dat het geval is in een IF-THEN-ELSE bewerking of een CASE-clausule.

control store *besturingsgeheugen*

In een computer die met behulp van een microprogramma is geprogrammeerd, verwijst de term naar het geheugen waarin microprogramma's zijn of worden opgeslagen.

control variable *besturingsvariabele*

Een variabele die wordt gebruikt om de iteratieve uitvoering van een lus of een groep te besturen.

control word *besturingswoord*

Alle besturingsvelden die worden gebruikt om een bepaalde groep records te sorteren of samen te voegen. Het belangrijkste veld verschijnt het eerst en de andere velden volgen in een afdalende volgorde van belangrijkheid.

conventions *conventies*

Eisen die worden gebruikt om een gedisciplineerde en uniforme benadering voor het verkrijgen van de consistentie in een software-product te beschrijven, dat wil zeggen uniforme methoden of werkwijzen voor de sortering, de vastlegging en de rangschikking van de gegevens.

conversational *conversatieel*

De term heeft betrekking op een interactief systeem of een interactieve bewerking waarin de interactie tussen de gebruiker en het systeem op die van een menselijke dialoog lijkt.

conversion testing *conversietesten*

Het testen van programma's of procedures die worden gebruikt om gegevens van bestaande systemen naar andere systemen te converteren.

conversion *conversie*

Modificatie van bestaande software om het te kunnen laten werken met gelijksoortige functionele mogelijkheden in een andere omgeving; bijvoorbeeld, het converteren van Fortran naar Ada, of het converteren van een programma dat op een bepaalde computer draait naar een andere computer.

convert (to) *converteren*

Het wijzigen van de weergave van gegevens van de ene vorm in een andere zonder verandering in de betekenis ervan aan te brengen, bijvoorbeeld codeconversie of analoog/digitaal conversie.

copy *kopie*

Het resultaat van het kopieerproces zoals dat in de volgende definitie wordt omschreven.

copy (to) *kopiëren*

Gegevens vanuit een bepaalde bron uitlezen waarbij die brongegevens ongewijzigd blijven en dezelfde gegevens op een andere plaats en in een fysieke vorm die van de oorspronkelijke kan verschillen, overnemen, deponeren of schrijven. Bijvoorbeeld, het kopiëren van gegevens vanaf een magnetische schijf naar een magneetband.

core *kern*

Een oude naam voor een computergeheugen. De naam stamt uit de tijd dat computergeheugens uit ferrietringetjes bestonden.

core class *kernklasse*

Een voor iedereen toegankelijke en beschikbare klasse (of interface) die standaard deel van het Java-platform uitmaakt. Het ligt in de bedoeling dat de kernklassen van Java ten minste op alle bedrijfssystemen waarop Java draait, beschikbaar zal zijn. Een 100% zuiver Java-programma is slechts afhankelijk van de kernklassen. Dit betekent dat het overal kan draaien.

coroutine *coroutine*

Een routine waarvan de werking op het punt aanvangt waarop deze het laatst was opgeschort. Het is hierbij niet nodig dat de besturing wordt teruggegeven aan het programma of subprogramma dat de coroutine heeft opgeroepen.

Corporation for Open Systems: COS *COS*

Een internationaal consortium van computergebruikers en -leveranciers. Het is in het leven geroepen met het doel beschikbare OSI-implematies te testen.

corrective maintenance *correctief onderhoud*

Onderhoud dat wordt uitgevoerd om fouten in hardware of software te corrigeren.

correctness *correctheid*

1. De mate waarin een systeem of een component vrij is van fouten voor wat betreft de specificatie, het ontwerp en de implementatie van dat systeem of component.
2. De mate waarin software, documentatie of andere items zich aan de specificaties conformeren.
3. De mate waarin software, documentatie of andere items zich conformeren aan de behoeften en verwachtingen van gebruikers. Deze kunnen al dan niet gespecificeerd zijn.

counter *teller*

Een variabele die wordt gebruikt om het aantal occurrenties van een bepaalde gebeurtenis tijdens de uitvoering van een computerprogramma vast te leggen; bijvoorbeeld, een variabele die het aantal malen dat een lus wordt uitgevoerd, vastlegt.

coupling *koppeling*

1. De wijze waarop en de mate waarin softwaremodules en assets een onderlinge afhankelijkheid bezitten. Koppelingen kunnen in de volgende soorten voorkomen: gemeenschappelijke omgevingskoppeling, inhoudskoppeling, besturingskoppeling, gegevenskoppeling, hybridische koppeling en pathologische koppeling.
2. De mate waarin de gegevens of de besturing tussen de verschillende softwaremodules of assets van een softwaresysteem onderling zijn gekoppeld of afhankelijk van elkaar zijn.
3. Een maat van de sterkte van de onderlinge verbinding tussen twee assets. Een hoge mate van de koppeling tussen de assets in een softwaresysteem kan de herbruikbaarheid beperken. Koppelingen zijn er de oorzaak van dat de modificatie van een systeem gecompliceerd en moeilijk kan worden. Omdat gekoppelde assets niet gemakkelijk van elkaar kunnen worden gescheiden en afzonderlijk gebruikt, wordt de mate waarin ze herbruikbaar zijn kleiner.

coverage *dekking*

De mate, uitgedrukt in een percentage, waarin een gespecificeerd dekkingsitem door een test of testserie is benut.

coverage item *dekkingsitem*

Een entiteit of eigenschap die als de basis voor een test of het testen wordt gebruikt.

crash *crash*

Een plotselinge en meestal moeilijk herstelbare storing van een computersysteem of component.

critical design review: CDR *kritische ontwerpbeoordeling*

1. Een beoordeling die wordt uitgevoerd met het doel (1) om na te gaan of het gedetailleerde ontwerp van één of meer configuratie-items voldoet aan de gespecificeerde eisen; (2) om vast te stellen wat de compatibiliteiten tussen de configuratie-items en andere items van de apparatuur, de faciliteiten, de software en het personeel zijn; (3) om de risicodragende gebieden voor elk afzonderlijk configuratie-item te bepalen; en (4), voor zover mogelijk en toepasbaar, de resultaten van productie-analyses in te schatten, voorlopige hardware-productspecificaties te beoordelen, voorlopige testplannen te evalueren, en de toepasselijkheid van voorlopige documenten betreffende de operationele en ondersteuningsfuncties te evalueren.
2. Een beoordeling zoals beschreven in punt 1 van deze definitie voor elk hardware- of softwarecomponent.

critical item *kritisch item*

Een item in een configuratie-item dat vanwege speciale ontwikkelingsoverwegingen of logistieke consideraties een goedgekeurde specificatie behoeft voor het bepalen van de technische besturing van dat item op componentniveau.

criticality *kritiek element*

De mate van de invloed die een behoefte, eis, module, fout, storing, of ander item op de ontwikkeling of werking van een systeem heeft.

critical piece first *kritisch (onder)deel eerst*

Een benadering van een systeemontwikkeling waarin de meest kritische aspecten van een systeem het eerst worden geïmplementeerd. Het kritische (onder)deel kan in termen van te verschaffen diensten, het risico, de moeilijkheidsgraad, of andere criteria zijn gedefinieerd.

critical section *kritieke sectie*

Een deel van een code waarin een draadmodule (thread) bepaalde systeemelementen of -middelen gebruikt (zoals bepaalde instantiatievariabelen) die door andere onderlinge koppelingen van draadmodules kunnen worden benut maar die niet op hetzelfde moment door die koppelingen mogen worden gebruikt.

critical software *kritische software*

Software waarvan storingen de veiligheid beïnvloeden of tot een groot financieel verlies of sociale onrust kunnen leiden.

cron *cron*

De klokdaemon in Unix die opdrachten op gespecificeerde data en tijden overeenkomstig instructies die in een bestand zijn opgeslagen, uit doet voeren.

cross assembler *kruisassembleerprogramma*

Een assembleerprogramma dat op een computer wordt verwerkt om uitvoer in een machinetaal te verkrijgen die op een computer met een geheel andere architectuur moet worden verwerkt.

cross compiler *kruiscompiler*

Een compileerprogramma dat een programma assembleert dat op een computer, die afwijkt van de computer waarop de kruiscompiler operationeel is, moet worden verwerkt.

cross program *kruisprogramma*

Een programma voor een computer dat door een andere computer wordt uitgevoerd.

cross-reference generator *kruisverwijzingsgenerator*

Een softwaregereedschap dat als invoer de broncode van een computerprogramma accepteert en als uitvoer een lijst produceert waarin alle variabelen, labels, en andere identificatiesymbolen worden opgesomd en aangeeft welke opdrachten in het programma deze variabelen, labels en identificatiesymbolen definieert of instelt, dan wel gebruik van deze variabelen, labels en identificatiesymbolen maakt.

cross-reference list *kruisverwijzingslijst*

Een lijst die alle variabelen, labels, en andere identificatiesymbolen in een computerprogramma identificeert en aangeeft welke opdrachten in het programma deze variabelen, labels en identificatiesymbolen definieert of instelt, dan wel gebruik van deze variabelen, labels en identificatiesymbolen maakt.

cross software *kruissoftware*

Software die op een bepaald type computer is ontwikkeld en op een ander type computer wordt gebruikt. De laatste computer bezit meestal niet de mogelijkheden die nodig zijn voor de ontwikkeling van de software.

C shell *1. C-schil; 2. C-shell*

De C-shell is een Unix-shell waarvan de indruk wordt gewekt alsof deze in de C-aal is geschreven. Deze shell is ontwikkeld aan de Universiteit van Berkeley. De shell roept de gebruiker op met het percentageteken (%).

current job *huidig karwei*

Het karwei dat het meest recentelijk is gestart of gestopt.

curses *curses*

Een verzameling subroutines in Unix voor het verplaatsen (het navigeren) van de cursor over het scherm van een beeldstation.

cursor *cursor*

De schermindicator die aangeeft met welke op het scherm geprojecteerde gegevens, menu's, programma's of commando's de gebruiker gaat werken of bezig is te werken. Een cursor kan diverse vormen aannemen (een streepje, een rechthoekje, een handje, een pijl, een zandloper enzovoort); dit is eventueel afhankelijk van het programma of de functie die wordt gebruikt of met behulp van de cursor wordt aangeropen, dan wel de plaats waarop de cursor zich op het scherm bevindt.

cycle *cyclus*

1. Een tijdsperiode waarbinnen een serie gebeurtenissen wordt voltooid of afgewerkt.
2. Een reeks bewerkingen die op regelmatige wijze in dezelfde volgorde, mogelijk met geringe wijzigingen, wordt herhaald.

cycle stealing *monocyclische interrupt*

Het proces waarin een bewerking van een centrale bewerkingseenheid gedurende één of meer cycli wordt opgeschort voor de occurrence van andere bewerkingen zoals het op verzoek transporteren van gegevens vanuit het hoofdgeheugen naar een invoer/uitvoerbesturingseenheid.

cyclic search *cyclische zoekprocedure*

Een techniek voor de toewijzing van geheugenruimte waarin elk zoekproces naar een geschikt geheugenblok aanvangt met het blok dat volgt op het blok dat het laatst werd toegewezen.

D

daemon *daemon*

Een proces dat bepaalde huishoudelijke taken uitvoert en in de achtergrond blijft draaien. Het proces zoekt naar situaties die verder moeten of kunnen worden afgewerkt. Zo heeft bijna iedere computer

een printer daemon. Deze daemon heeft als taak om gegevens die in een print-wachtrij staan, waar mogelijk af te doen drukken.

data 1. *data*; 2. *gegevens*

- a. Een representatie van feiten, concepten, of instructies op een wijze die geschikt is voor communicaties, interpretaties, of verwerkingen door mensen of door daarbij van automatische hulpmiddelen gebruik te maken.
- b. Gegevens die op zichzelf geen specifieke betekenis behoeven te hebben. Gegevens die in een bepaalde volgorde of structuur zijn geplaatst worden, indien er een betekenis aan wordt toegekend, 'informatie' genoemd. Zo bestaat de zin: 'hond de baas naast loopt zijn' uit gegevens zonder betekenis. Bij wijziging van het syntactische verband in 'de hond loopt naast zijn baas' krijgen de gegevens betekenis en vormen ze informatie.
- c. De term wordt soms als synoniem voor documentatie gebruikt.

data abstraction *gegevensabstractie*

1. Het proces waarin de essentiële eigenschappen en kenmerken van gegevens worden geëxtraheerd door gegevenstypen en hun geassocieerde functionele karakteristieke eigenschappen te definiëren en de details voor wat betreft de representatie te verwaarlozen.
2. Het resultaat van het proces dat in punt 1 van deze definitie is beschreven.

data aggregate *gegevensaggregaat*

1. Een verzameling gegevens in een database.
2. Een logische verzameling van twee of meer gegevensbestanddelen waarnaar zowel collectief als afzonderlijk kan worden gerefereerd.

data attribute *gegevensattribuut*

Een eigenschap van een gegeven of een eenheid van gegevens, zoals lengte, kleur, de waarde, of weergavemethodiek.

database 1. *database*; 2. *gegevensbank*

Een verzameling onderling aan elkaar gerelateerde gegevens die gezamenlijk in één of meer gecomputeriseerde bestanden zijn opgeslagen. Een database kan een bestand, een asset, het mechanisme voor de opslag van een verzameling bij elkaar behorende bestanden of assets, of een catalogus zijn.

data base call *database-oproep*

Een methode om de identiteit van bepaalde gegevens binnen een database te bepalen.

data base component *databasebestanddeel*

Het bestanddeel of component dat faciliteiten ten behoeve van databases biedt.

data base file *databasebestand*

Een object dat omschrijvingen bevat van de wijze waarop invoergegevens aan een programma moeten worden aangeboden vanuit het interne geheugen. Dit geschiedt door middel van door de gebruiker gecodeerde opdrachten.

data base management system: DBMS *databasebeheerssysteem*

Een systeem dat grote gestructureerde verzamelingen blijvende (niet vluchtige) gegevens bevat en voorzien is van mogelijkheden waarmee gebruikers ad hoc vragen aan de database kunnen stellen. Databases worden op grote schaal in systemen die bedrijfsgegevens verwerken en opslaan toegepast. Commerciële applicaties zijn Ingres, Oracle, Sybase enzovoort.

database schema *databaseschema*

De informatie die de structuur van de gegevens in een database beschrijft. Het ontwerp van een catalogus of een bibliotheek kan in een databaseschema zijn opgenomen. De database verwijst echter naar een databaseschema dat actuele gegevens of actuele exemplaren (instantiaties) van gegevens bevat. Daarnaast representeren een databaseschema en de database afzonderlijke assets en ondersteunen verschillende mogelijkheden voor het hergebruik daarvan.

data base system *databasesysteem*

Een systeem of uitgebreid programma dat het tot stand brengen van meervoudige toepassingen, die gebruik maken van een gemeenschappelijke database, ondersteunt.

data breakpoint *gegevensbreekpunt*

Een breekpunt dat wordt geïnitieerd indien toegang tot een gespecificeerd gegevensitem wordt verkregen.

data characteristic *gegevenskenmerk*

Een inherente en mogelijk toevallige kwaliteit of eigenschap van een gegevensitem of gegevens.

data code *gegevenscode*

Een gestructureerde reeks tekens die wordt gebruikt om bepaalde gegevens weer te geven; zo kunnen de coderingen 01, 02 enzovoort worden gebruikt om de maanden van het jaar weer te geven.

data collection *gegevensverzameling*

1. Verzameling van gegevens die, bijvoorbeeld, in een wachtrij zijn geplaatst om op een bepaald tijdstip te worden verwerkt.
2. Verzameling gegevens die een deel van een te verzenden bericht in een netwerk vormt en tezamen met andere gegevensverzamelingen een volledig te verzenden of verzonden bericht vormt.

data compaction *gegevensverdichting*

Een methode om gegevens te coderen om de daarvoor benodigde geheugenruimte te kunnen verkleinen.

data component *gegevensbestanddeel*

Het gedeelte van een gegevensverzameling, index of catalogus dat specifieke gegevens of gegevensrecords bevat.

data content *gegevensinhoud*

Alle gegevens en kenmerken in een informatieverwerkend systeem met inbegrip van de onderlinge afhankelijkheden tussen de gegevens.

data conversion *gegevensconversie*

De omzetting van gegevens van de ene vorm in een andere of van de ene code in een andere code.

data converter *gegevensomzetter*

Een eenheid die gegevens converteert.

data coupling *gegevenskoppeling*

Een soort koppeling waarin de uitvoer van één bepaald softwaremodule als invoer van een ander module wordt gebruikt.

data declaration *gegevensverklaring*

Een niet-uitvoerbaar opdracht die de kenmerken en eigenschappen van de gegevens beschrijft waarop (eventueel) een bewerking moet worden verricht.

data definition *gegevensdefinitie*

1. Een programma-opdracht die de kenmerken van of de samenhang tussen gegevens specificceert.
2. Een uitvoerbare uitdrukking of instructie waarin aan een variabele een waarde is toegekend.

data definition C-use pair *gegevensdefinitie en C-gebruikspaar*

Een gegevensdefinitie en gegevensgebruik voor rekenprocessen waarbij het gegevensgebruik de waarde hanteert die in de gegevensdefinitie is gedefinieerd.

data definition C-use pair coverage *dekking van de gegevensdefinitie en C-gebruik*

Het percentage van de gegevensdefinitie en C-gebruiksparen in een bepaald component dat door een test of testserie wordt onderzocht.

data definition language: DDL *gegevensdefinitietaal*

Een taal waarmee de structuur en de inhoud van een database in zowel door de mens als machine interpreteerbare vorm kan worden gedefinieerd.

data definition P-use pair *gegevensdefinitie en P-gebruikspaar*

Een gegevensdefinitie en gegevensgebruik voor predikaten waarbij het gegevensgebruik de waarde hanteert die in de gegevensdefinitie is gedefinieerd.

data definition P-use pair coverage *dekking van de gegevensdefinitie en P-gebruik*

Het percentage van de gegevensdefinitie en P-gebruiksparen in een bepaald component dat door een test of testserie wordt onderzocht.

data definition-use coverage *dekking van het gegevensdefinitiegebruik*

Het percentage gegevensdefinitie-gebruiksparen in een component dat door een test of testserie wordt onderzocht.

data definition-use pair *gegevensdefinitie-gebruikspaar*

Een gegevensdefinitie en gegevensgebruik waarbij het gegevensgebruik de waarde hanteert die in de gegevensdefinitie is gedefinieerd.

data definition-use testing *testen van gegevensdefinitiegebruik*

Een techniek voor het ontwikkelen van testen of testseries voor een component waarin de testen zodanig zijn geconcipieerd dat ze gegevensdefinitie-gebruiksparen uitvoeren.

data delimiter *gegevens-scheidingstekens*

1. Een identificatieteken dat gegevens van elkaar scheidt of gegevens indeelt.
2. Een rij van één of meer tekens die wordt gebruikt om elementen van programma's of gegevens van elkaar te scheiden of in te delen. Voorbeelden hiervan zijn linker en rechter haakjes, het spatieteken, rekenkundige bewerkingstekens en speciale gereserveerde woorden (of sleutelwoorden) zoals IF en BEGIN.
3. Een teken dat woorden of waarden in een invoerregel in groepen indeelt of van elkaar scheidt.
4. Alle bewerkingstekens en commentaren, alsmede de volgende tekens: procent, haakje, komma, punt, puntkomma, toewijzingsymbool en spatieteken; zij bepalen de grenzen van identificatiesymbolen, constanten, beeldspecificaties en sleutelwoorden.

data description language: DDL *gegevensbeschrijvingstaal*

Een taal die faciliteiten bezit waarmee gegevens en hun onderlinge samenhang in een database kunnen worden beschreven.

data dictionary *gegevensdictionaire*

1. Een verzameling gegevensbeschrijvingen die door verschillende toepassingen kan worden gedeeld.
2. Een gecentraliseerde opslagplaats van informatie over gegevens zoals de betekenis, de verwantschappen of associaties met andere gegevens, de oorsprong, het gebruik en de indeling. Deze dictionaire draagt bij tot het doelmatig plannen, beheren en evalueren van het verzamelen, opslaan en gebruik van gegevens.

data element *gegevenselement*

De kleinste gegevenseenheid waaraan nog een betekenis kan worden toegekend.

data exception *gegevensexceptie*

Een uitzonderingstoestand die optreedt indien een programma gegevens op onjuiste wijze probeert te gebruiken of probeert toegang tot dergelijke gegevens te krijgen.

data file *gegevensbestand*

1. Een verzameling van aan elkaar verwante gegevens of gegevens-records die op een bepaalde wijze zijn georganiseerd.
2. Een direct toegankelijk bestand dat de programmeergegevens en/of -records van de in- en uitvoer bevat.

data flow *gegevensstroom*

1. De volgorde waarin tijdens de uitvoering van een programma het transport, het gebruik, en de transformatie van gegevens wordt uitgevoerd.
2. De weg die een gegeven of bericht volgt vanaf een terminal of een (toepassings)programma van de oorsprong naar de bestemming met inbegrip van alle eventuele knooppunten in een netwerk die dit gegeven of bericht verwerken en/of doorgeven.

data flowchart *gegevensstroomschema*

Een stroomschema dat de weg van de gegevens aangeeft bij het oplossen van een probleem en dat de belangrijkste verwerkingsfasen en/of -functies omschrijft, alsmede de verschillende gebruikte gegevensmedia. Zie ook gegevensstroomdiagram (data flow diagram).

data flow coverage *dekking van de gegevensstroom*

Een maat van de dekking van een test of testserie die gebaseerd is op het gebruik van variabelen in de code. Voorbeelden hiervan zijn: de dekking van het gegevensdefinitiegebruik, de dekking van de

gegevensdefinitie en C-gebruik, en de dekking van de gegevensdefinitie en P-gebruik.

data flow diagram: DFD *gegevensstroomdiagram*

1. Een schema of diagram dat gegevensbronnen, gegevensputten (data sinks), gegevensopslag en processen uitbeeldt en waarin de gegevens als knooppunten en de logische verbindingspaden als takken tussen die knooppunten worden uitgebeeld.
2. Een grafische notatie die wordt gebruikt om te beschrijven hoe gegevens tussen processen in een systeem worden uitgewisseld. Dit is een belangrijk gereedschap in de meeste gestructureerde analysetechnieken.

data flow testing *testen van de gegevensstroom*

Het uitvoeren van testen waarin testseries zodanig zijn geconcipieerd dat ze gebaseerd zijn op het gebruik van variabelen in de code.

data generation language: DGL *gegevensgeneratietaal*

Een gereedschap voor de generatie van testgegevens voor hardware- of softwaresystemen.

data input sheet *gegevensinvoerblad*

Gebruikersdocumentatie die de vereiste en optionele invoergegevens voor een systeem of een component bevat. Een dergelijk blad heeft de vorm van een rekenblad (work sheet).

data integrity *gegevensintegriteit*

1. De bestaande kwaliteit van de gegevens, zolang het al dan niet opzettelijk vernietigen, wijzigen of verliezen van gegevens wordt voorkomen.
2. Het behoud van de gegevens voor het beoogde doel.

data library *gegevensbibliotheek*

Een verzameling van aan elkaar gerelateerde bestanden.

data list *gegevenslijst*

Een tussen haakjes geplaatste lijst met uitdrukkingen of zich herhalende specificaties die door komma's van elkaar zijn gescheiden. Deze lijst wordt gebruikt in een stroom-georiënteerde in- of uitvoerspecificatie die de geheugenposities weergeeft waaraan de gegevenselementen moeten worden toegewezen tijdens het invoeren, of waarden die moeten worden verkregen voor de uitvoer. Deze definitie wordt o.a. in de programmeertaal PL/1 gebruikt.

data model *gegevensmodel*

1. Een verzameling entiteiten, operatoren, bewerkingsopdrachten, en consistentieregels.

2. Bij een database verwijst de term naar een omschrijving van de wijze waarop gegevens zijn georganiseerd.
3. Een verzameling gegevensstructuren met bewerkingsinstructies ten behoeve van de manipulatie en de validatie van deze structuren en de daarin vervatte gegevens. Deze zijn voor algemeen gebruik. Voorbeelden van gegevensmodellen zijn het entiteit-relatiemodel en Nijsens informatie-analysemethode (NIAM).

data name *gegevensnaam*

1. Een teken dat of een groep tekens die wordt gebruikt om een gegevensbestanddeel aan te duiden.
2. Een naam die door een programmeur aan een gegevensbestanddeel in een programma wordt toegekend.

data object *gegevensobject*

Een element van een gegevensstructuur in een programma. Bijvoorbeeld, een bestand, een reeks, of een opdrachtelelement dat nodig is om het programma uit te voeren en dat wordt benoemd of anderszins wordt gespecificeerd door de toegestane tekenset van de taal waarin het programma is gecodeerd.

data organization *gegevensorganisatie*

De rangschikking en indeling van de gegevens in een gegevensbestanddeel. Dit kan, bijvoorbeeld, sequentieel of gesegmenteerd zijn.

data processing *gegevensverwerking*

De systematische uitvoering van bewerkingen op gegevens.

data record *gegevensrecord*

Een verzameling gegevensbestanddelen gezien vanuit de optiek van het gebruik ervan in een bepaalde toepassing.

data-sensitive fault *gegevensgevoelige fout*

Een fout die een storing veroorzaakt als reactie op een specifiek patroon of een specifieke volgorde van de aangeboden of te verwerken gegevens.

data set *gegevensverzameling*

De belangrijkste eenheid waarin gegevens kunnen worden opgeslagen en van waaruit ze kunnen worden opgezocht. Deze eenheid bestaat uit gegevens die in een bepaalde volgorde zijn geplaatst en die door de besturingsinformatie waartoe het systeem toegang heeft, zijn omschreven.

data stream *gegevensstroom*

1. Alle gegevens die via een kanaal in één enkele lees- of schrijfbewerking worden verzonden.
2. Een ononderbroken stroom gegevenselementen die in de vorm van bits of tekens in een bepaalde opmaak moeten worden verzonden.
3. Gegevens die naar of van een bepaalde systeem-, computer-, programma- of netwerkfunctie worden verzonden.

data structure *gegevensstructuur*

1. De syntactische structuur van symbolische uitdrukkingen en de kenmerken van de geheugentoewijzing ervan.
2. Een fysieke of logische relatie tussen gegevenselementen die als zodanig is geconcipeerd en tot doel heeft bepaalde functies te ondersteunen waarin gegevensmanipulaties kunnen worden uitgevoerd.

data structure-centered design *ontwerp gebaseerd op gegevensstructuren*

Een software-ontwikkelingstechniek waarin de architectuur van een systeem wordt afgeleid door middel van een analyse van de structuur van de gegevensverzamelingen waarmee het systeem moet werken.

data structure diagram *gegevensstructuurdiagram*

Een diagram dat een verzameling gegevenselementen, hun attributen, en de logische relaties tussen die gegevenselementen uitbeeldt.

data transfer *gegevensoverdracht*

Het transporteren of kopiëren van gegevens van de ene locatie naar een andere.

data type *gegevenstype*

1. De structurele kenmerken en eigenschappen van gegevens die rechtstreeks kunnen worden gespecificeerd met behulp van een programmeertaal, zoals de gehele en reële getallen in Fortran of Pascal.
2. De wiskundige eigenschappen en de interne representatie van gegevens en functies.
3. Een klasse gegevens die wordt gekenmerkt door de leden van de klasse en de bewerkingen die op die leden kunnen worden uitgevoerd. Bijvoorbeeld, tekentype, opsommingstype, integer type, logisch type, reëel type enzovoort.

data use *gegevensgebruik*

Een uitvoerbare instructie waarin de waarde van een variabele wordt opgehaald.

data value *gegevenswaarde*

Bij grafische voorstellingen heeft de term betrekking op één enkel gegevensbestanddeel dat, bijvoorbeeld, als de waarde van een x- of y-coördinaat wordt ingevoerd of verwerkt.

deadlock *dodelijke omhelzing*

Een situatie waarin de computerverwerking wordt opgeschort omdat twee of meer eenheden of processen op de systeemelementen en/of computermiddelen wachten die aan anderen zijn toegekend.

deblock (to) *ontblokken*

De delen van een blok van elkaar scheiden.

debug (to) *fouten opsporen*

De fouten en de oorzaken van fouten in software opsporen, lokaliseren, en corrigeren. Technieken die hierbij worden toegepast omvatten breekpunten, handmatige controles, dumps, inspecties, omgekeerde uitvoeringen van bewerkingen, stap-voor-stap bewerkingen, en traceringen (naspeuringen).

debugger *testprogramma*

Een programma dat is ontwikkeld om op eenvoudige wijze fouten in programmatuur op te sporen en eventueel te verwijderen.

debug mode *werkwijze voor het opsporen van fouten*

Een werkwijze die door de software-ontwikkelaar wordt toegepast gedurende het testen van de ontwikkelde programmatuur om na te kunnen gaan welke veranderingen er onder normaal gebruik op kunnen treden.

DECdesign *DECdesign*

Een gereedschap voor de analyse en het ontwerp van software van DEC. Het ondersteunt verschillende methodologieën.

decimal point *decimale punt*

De punt of de komma die in decimale getallen de fracties van de gehele delen van de getallen scheidt.

decision *beslissingspunt*

Het punt in een programma (het programmapunt) waarin het logische vervolg van het programma twee of meer alternatieve routes heeft.

decision condition *1. beslissingstoestand; 2. beslissingsconditie*

Een toestand of een conditie binnen een te nemen beslissing.

decision coverage *dekking van beslissingspunten*

Het percentage programma- of beslissingspunten (beslissingsresultaten) dat door een test of een testserie is onderzocht.

decision instruction *beslissingsinstructie*

Een instructie die er voor zorgt dat er een keuze wordt gemaakt tussen opeenvolgende uit te voeren spronginstructies binnen een programma.

decision outcome *beslissingsresultaat*

Het resultaat of de consequentie van de actie in een beslissingspunt en waarin als gevolg van dat resultaat een alternatieve route van de besturingsstroom wordt bepaald.

decision point *beslissingspunt*

Het punt in een proces of programma waarin een beslissing wordt of moet worden genomen.

decision table *beslissingstabel*

Een tabel die wordt gebruikt om een aantal condities en de acties die daar het gevolg van zijn, uit te beelden.

declaration *1. verklaring; 2. declaratie*

- a. Een niet-uitvoerbare instructie die een identificatiesymbool en de daarmee geassocieerde attributen definieert. In deze instructie wordt de opslagruimte voor de gegevens die bij het identificatiesymbool of de attributen behoren, niet noodzakelijkerwijs vastgelegd noch wordt daarin de uiteindelijke implementatie van de methoden gerealiseerd.
- b. Een niet-uitvoerbare programma-instructie die de interpretatie van andere opdrachten in een assembleerprogramma of compileerprogramma beïnvloedt. Bijvoorbeeld een verklaring die aangeeft wat de naam voorstelt en mogelijk ook een initiële waarde aan die naam toekent.

declarative language *niet-procedurele taal*

Een niet-procedurele taal waarmee de gebruiker een aantal feiten kan specificeren en waarin hij of zij vragen of problemen kan formuleren waarin deze feiten worden gebruikt.

declaratives *verklarende uitdrukkingen*

Een reeks van één of meer modules die aan het begin van het proceduregedeelte van bepaalde programmeertalen (waaronder Cobol) zijn geschreven.

declare (to) *benoemen*

Bij programmeertalen verwijst deze term naar het geven van een naam van de variabele symbolen die door het programma zullen worden gebruikt.

DECnet *DECnet*

Het netwerk waarmee de firma DEC haar computers onderling verbindt.

decompile (to) *decompileren*

Het vertalen van een gecompileerd computerprogramma vanuit de machinetaal-versie van dat programma naar een vorm die lijkt op de oorspronkelijke hogere-orde programmeertaal. Deze vorm is niet noodzakelijkerwijs identiek aan die oorspronkelijke vorm.

decompiler *decompileerprogramma*

Een softwaregereedschap dat computerprogramma's decompileert.

decoupling *ontkoppeling*

Het proces waarin softwaremodules of assets onafhankelijker ten opzichte van elkaar worden gemaakt. Dit wordt gedaan om het effect van wijzigingen en fouten in afzonderlijke modules of assets te verkleinen.

DECstation *DECstation*

Een serie op RISC gebaseerde werkstations die door DEC worden gefabriceerd en verkocht.

DECwindows *DECwindows*

De window-omgeving van DEC. Deze is gebaseerd op het X Window systeem.

default *systeemgekozen waarde*

Een waarde of een mogelijkheid die door het systeem wordt aangenomen wanneer geen andere waarden of mogelijkheden zijn gespecificeerd.

defined item *gedefinieerd item*

Een variabele waarvan wordt of is vastgelegd dat deze een gedeelte van hetzelfde geheugen vertegenwoordigt als dat wat aan een andere variabele is toegewezen.

definition *definitie*

Een verklaring (declaration) waarin opslagruimte voor gegevens wordt gereserveerd of waarin de implementatie van methoden wordt vastgelegd.

definition and design of an open dependable distributed system architecture: Delta-4 *Delta-4*

Een ESPRIT-project waarin de realisatie van afhankelijkheden in open gedistribueerde systemen (waartoe ook real-time systemen worden gerekend) wordt onderzocht.

delimiter *scheidingstekens*

1. Een identificatieteken dat gegevens van elkaar scheidt of gegevens indeelt.
2. Een teken dat woorden of waarden in een invoerregel in groepen indeelt of van elkaar scheidt.
3. Een teken dat woorden of waarden in een invoerregel in groepen indeelt of van elkaar scheidt.
4. Alle bewerkingstekens en commentaren, alsmede de volgende tekens: procent, haakje, komma, punt, puntkomma, toewijzings-symbool en spatieteken; zij bepalen de grenzen van identificatiesymbolen, constanten, beeldspecificaties en sleutelwoorden.

delivery *oplevering*

De vrijgave van een systeem of component aan de klant of de beoogde gebruiker.

delta *delta*

De informatie waarin een versie van de overige leden van een bepaalde familie van versies verschilt.

Delta-4 system environment: DELTASE *Delta-4 systeemomgeving*

Een gedistribueerde verwerkingsomgeving die gericht is op fout-tolerante toepassingen en applicaties uit de procesbesturing. Deze omgeving wordt in het kader van het Delta-4 ESPRIT-project gedefinieerd.

demand paging *paginering op verzoek*

Een geheugentoekenningstechniek waarin pagina's slechts vanuit het hulpgeheugen naar het hoofdgeheugen worden overgebracht indien die pagina's nodig zijn.

demodularization *demodularisatie*

Het proces in de software-ontwikkeling waarin aan elkaar verwante softwaremodules worden gecombineerd. Dit geschiedt gewoonlijk met het doel de prestaties van het systeem te optimaliseren.

demonstration *demonstratie*

Een dynamische analysetechniek die gebaseerd is op het waarnemen en het volgen van het gedrag van een systeem of component tijdens de uitvoering daarvan. Het is daarbij niet nodig een analyse na afloop van de bewerkingen uit te voeren of fouten, overtredingen van ontwikkelingsnormen, en andere problemen te detecteren.

deployment diagrams *deployeringsdiagrammen*

De term verwijst naar diagrammen of schema's die de configuratie van verwerkingseenheden gedurende de uitvoeringstijd tonen, met inbegrip van de hardware en de software die er op draait. Deze schema's zijn eenvoudige modellen die vaak worden gebruikt om aan

te geven hoe de hardware- en software-eenheden zullen worden geconfigureerd en hoe deze voor applicaties zullen worden ingezet.

depreciation *depreciatie*

De term heeft betrekking op een klasse, interface, constructor, methode of veld dat niet langer meer is aanbevolen en in toekomstige versies wellicht niet meer aanwezig zal zijn.

derived from *afgeleid van*

Klasse X is 'afgeleid van' klasse Y indien klasse X klasse Y uitbreidt. Zie ook bij deelklassen en superklassen.

derived type *afgeleid type*

1. Een type waarvan de bewerkingen en waarden replica's van die van bestaande typen zijn.
2. Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens en de bewerkingen daarop van die van andere types zijn afgeleid. Dit gebeurt volgens bepaalde gespecificeerde regels.

description standard *descriptiestandaard*

Een standaard waarin de kenmerkende eigenschappen van productinformatie of procedures zijn beschreven die kunnen helpen bij het begrijpen, testen, installeren, bedrijven of onderhouden van het product.

descriptive intermediate attributed notation for Ada: DIANA *DIANA*

Een ontwikkel- en ontwerptaal voor Ada-programma's.

design *ontwerp*

1. Het proces waarin de architectuur, componenten, interfaces en andere kenmerkende eigenschappen van een systeem of component worden gedefinieerd.
2. Het resultaat van het in punt 1 van deze definitie genoemde proces.
3. Het ontwerp bestaat meestal uit de fase van de ontwikkeling van software die op de analyse volgt en zich bezighoudt met de vraag hoe het probleem moet worden opgelost.

design analysis *ontwerpanalyse*

Een formele technische beoordeling of een sterk gereguleerde en gedisciplineerde bijeenkomst die door de technische staf wordt uitgevoerd, respectievelijk belegd met het doel kwaliteitsproblemen op te sporen.

design-based testing *ontwerp-gebaseerd testen*

Ontwerptesten die gebaseerd zijn op doelstellingen die uit het architectonische of gedetailleerde ontwerp van de software zijn

afgeleid. Bijvoorbeeld testen die specifieke paden met bewerkingen uitvoeren of het meest slechte gedrag van algoritmen navorsen.

design description *ontwerpbeschrijving*

Een document waarin het ontwerp van een systeem of component is/wordt beschreven. De inhoud van een dergelijk document bevat over het algemeen de architectuur, de besturingslogica, gegevensstructuren, invoer/uitvoerformaten, interfacebeschrijvingen en algoritmen.

design element *ontwerpelement*

Een basiscomponent of bouwblok in een ontwerp.

design entity *ontwerpentiteit*

Een element (component) van een ontwerp dat zich structureel en functioneel onderscheidt van andere elementen en dat een eigen unieke naam heeft waar apart naar wordt verwezen.

design language *ontwerptaal*

Een specificatietaal met speciale constructies en soms verificatieprotocollen. De taal wordt gebruikt voor het ontwikkelen, analyseren en documenteren van een in hardware of software uitgevoerd ontwerp.

design level *ontwerpniveau*

De decompositie in een ontwerp van een software-item (bijvoorbeeld, systeem, subsysteem, programma, module of asset).

design phase *ontwerpfase*

Het tijdsbestek in de levenscyclus van software gedurende welke de ontwerpen voor architectuur, softwarecomponenten, interfaces, en gegevens worden gerealiseerd, gedocumenteerd, en op de gestelde eisen worden geverifieerd.

design recovery *herstel van het ontwerp*

Een deelgebied van de terugwerkende ontwikkeling (reverse engineering) waarin domeinkennis, externe informatie, en deductie van vage redeneringen aan de observaties van het beoogde systeem worden toegevoegd om op die wijze een meer betekenisvol hoger abstractieniveau te identificeren buiten dat wat met behulp van een direct onderzoek van het systeem zelf kan worden verkregen.

design requirement *ontwerpeis*

Een eis die het ontwerp van een systeem of een systeemcomponent specificiert of beperkt.

design review *ontwerpbeoordeling*

Een proces of een bijeenkomst waarin een systeem-, hardware- of software-ontwerp aan projectmedewerkers, managers, gebruikers, klanten, of andere geïnteresseerden voor becommentariering of goedkeuring wordt gepresenteerd. Er bestaan een aantal verschillende soorten ontwerpbeoordelingen zoals, kritische ontwerpbeoordeling, voorlopige ontwerpbeoordeling en systeemontwerpbeoordeling.

design standard *ontwerpnorm*

Een standaard (norm) die de kenmerkende eigenschappen van een ontwerp of een ontwerpspecificaties van gegevens of modules beschrijft.

design unit *ontwerpeenheid*

Een logische verzameling van onderling aan elkaar verwante ontwerpelementen. Zo kan in de programma-ontwerptaal van Ada een ontwerpeenheid door een Ada compilatie-eenheid worden gerepresenteerd.

desk checking *handmatige controle*

1. Het testen van software met behulp van de handmatige simulatie van de uitvoering of werking ervan.
2. Een statische analyse waarin coderingslijsten, testresultaten, of andere documenten visueel worden gecontroleerd met het doel fouten, het niet juist toepassen van ontwerpnormen, of andere problemen op te sporen en te verbeteren. Deze controle wordt meestal uitgevoerd door de personen die deze documenten hebben gegenereerd.

destination address *bestemmingsadres*

Het adres of de geheugeneenheid waar gegevens naar toe worden gezonden.

destructive read *destructieve leesbewerking*

Een leesbewerking die de gegevens in de locatie waaruit gegevens moeten worden opgehaald of gelezen, verandert.

destructor *destructor*

Een functie die door een klasse in C++ wordt uitgevoerd en die een object schrapt.

detailed design *gedetailleerd ontwerp*

1. Het proces waarin het voorlopige ontwerp van een systeem of component dusdanig volledig wordt verfijnd en uitgewerkt dat het kan worden geïmplementeerd.
2. Het resultaat van het proces dat in punt 1 van deze definitie is beschreven.

Duits normalisatie- en standaardisatie-instituut. Het DIN is het Duitse lid van de CEN en ISO.

development *ontwikkeling*

Het proces van de analyse, het ontwerp, de codering, en het testen van een product, bijvoorbeeld software.

developmental configuration *ontwikkelingsconfiguratie*

De software en daarmee verwante technische documentatie die de tijdens de ontwikkeling geleidelijk opgebouwde configuratie van een configuratie-item definieert. Opmerking: De ontwikkelingsconfiguratie wordt door de ontwikkelaar beheert en beheerst en wordt daarom niet met de term 'voetlijn' aangeduid.

development testing *ontwikkelingstest*

Een formele of informele test die tijdens de ontwikkeling van een systeem of component wordt uitgevoerd. Dit geschiedt meestal in de ontwikkelingsomgeving van de ontwikkelaar.

deviation *afwijking*

1. Een afwijking van een gespecificeerde eis of behoefte.
2. Een op schrift gestelde machtiging, die wordt verstrekt voordat de productie van een item plaats vindt en voor het mogen afwijken van een bepaalde prestatie- of ontwerp-eis die geldt voor een specifiek aantal eenheden en een gedefinieerde tijdsperiode. Opmerking: Anders dan dat het geval is in een ontwerpwijziging, wordt hier niet geëist dat de documentatie waarin het betrokken item is gedefinieerd eveneens wordt gewijzigd.

device *eenheid*

Een mechanisme of een (deel van een) apparaat dat ontworpen is met het doel een bepaalde gespecificeerde functie te kunnen verrichten.

Dhrystone *Dhrystone*

Een benchmark-programma in C en Ada.

diagnostic *diagnostisch*

Een term die naar het opsporen en isoleren van fouten of storingen in hardware of software verwijst.

diagnostic manual *diagnostische handleiding*

Een document dat de informatie bevat die nodig is om diagnostische procedures voor een systeem of component uit te kunnen voeren, fouten te kunnen identificeren, en die fouten te kunnen herstellen. Met name worden in een dergelijke handleiding de diagnostische kenmerken van het systeem of component beschreven, alsmede de

diagnostische gereedschappen die daarvoor ter beschikking staan of kunnen worden gebruikt.

diagnostic routine *diagnostische routine*

Een module die het onderhoud van een programma vergemakkelijkt door de mogelijkheid die het heeft om fouten op te sporen en te isoleren.

diagnostics *diagnostiek*

Het proces van het onderzoeken van de oorzaak of de aard van een situatie of probleem in een product of een systeem.

diagonal microinstruction *diagonale micro-instructie*

Een micro-instructie waarmee een beperkt aantal simultaan uit te voeren bewerkingen, die nodig zijn om een machinetaalinstructie uit te voeren, zijn gespecificeerd. Diagonale micro-instructies vallen, voor wat betreft omvang en functionaliteit, tussen de horizontale en verticale micro-instructies in. Het is deze laatste eigenschap die met het begrip 'diagonaal' wordt aangeduid.

dictionary *dictionnaire*

Een reeks gegevens waarvan elk element op ondubbelzinnige wijze door één of meer argumenten en de positie van het element ten opzichte van de andere elementen of door enig ander kenmerk wordt geïdentificeerd.

difference *1. differentie; 2. verschil*

De term verwijst in de domeinanalyse naar een asset of een duidelijk te onderscheiden deel, functie of eigenschap dat systemen binnen een bepaald domein van elkaar doet verschillen. Dit verschil wordt in een domeinmodel of domeinarchitectuur benadrukt.

digital document interchange format: DDIF *digitaal documentuitwisselingsformaat*

Een specificatie van DEC voor een gecombineerde documentarchitectuur (CDA) waarin samengestelde of gecombineerde documenten in een wijzigbare vorm worden gerepresenteerd.

dimension *dimensie*

Het maximale aantal waarden dat aan een rij, een reeks of een matrix kan worden toegewezen.

direct address *direct adres*

Een adres dat de geheugenplaats van een operand identificeert.

directed graph *1. gericht netwerk; 2. gerichte graaf*

Een netwerk of graaf waarin de richting van de verbindingen tussen de interne knooppunten van het netwerk of de graaf zijn weergegeven.

direct instruction *directe opdracht*

Een computeropdracht die de directe adressen van de operanden bevat.

directive *directief*

1. Bij een programmeertaal is dit een zinvolle uitdrukking die invloed heeft op de interpretatie van andere uitdrukkingen in de taal.
2. De definitie van een identificatiesymbool als een naam en de specificatie van een reeks kenmerken voor die naam.
3. Een niet-uitvoerbare opdracht waarin informatie over bepaalde gegevens van een programma wordt verstrekt.

directory *1. directory; 2. index*

- a. Een tabel met identificatiesymbolen en verwijzingen naar gegevensitems en informatie betreffende die items.
- b. Een index die door een besturingsprogramma wordt gebruikt om de positie van één of meer gegevensblokken, die in afzonderlijke gebieden van een geheugen zijn opgeslagen, te bepalen.
- c. Een index die door het systeembesturingsprogramma en de algemene programma's worden gebruikt om de positie van één of meer in een geheugen opgeslagen opeenvolgende blokken met informatie over een programma te bepalen.
- d. Een verzameling bestanden waaraan een naam is toegekend. Een index of directory kan worden vergeleken met een folder die één of meer bestanden bevat. Directories kunnen ook andere directories bevatten. Men kan een directory eveneens als een soort werkplek of werkomgeving beschouwen omdat één specifieke directory steeds de momentane werkdirectory is.

direct system output writer *uitvoerbesturing*

Een functie van het programma dat de bewerkingen indeelt. Die bewerkingen besturen het vastleggen van de uitvoergegevens tijdens het uitvoeren van de desbetreffende bewerkingen.

disassemble (to) *disassembleren*

Het vertalen van een geassembleerd computerprogramma vanuit de machinetaal van dat programma naar een vorm die op het oorspronkelijke assembleerprogramma lijkt, maar daar niet geheel identiek mee hoeft te zijn.

disassembler *disassembleerprogramma*

Een softwaregereedschap dat computerprogramma's disassembleert.

discrete type *discreet type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens een serie afzonderlijke waarden aan kunnen nemen. Een discreet type kan een opsommingstype of een integer type zijn.

disk operating system: DOS *DOS*

Een bedrijfssysteem voor computers voor persoonlijk gebruik dat met Unix als voorbeeld is ontwikkeld.

distributed *gedistribueerd*

Een proces of programma dat in meer dan één adresruimte werkt (draait).

distributed application architecture: DAA *gedistribueerde applicatie-architectuur*

Een beheersomgeving voor gedistribueerde objecten die door Hewlett-Packard en Sun Microsystems is/wordt ontwikkeld. In deze architectuur kunnen applicaties onafhankelijk van een bedrijfssysteem, netwerk of window-systeem worden ontwikkeld

distributed data base system: DDBS *gedistribueerd databasesysteem*

Een database die de verschijningsvorm van één enkele logische eenheid heeft doch die fysiek over verschillende locaties is verspreid.

distributed naming service van DEC: DECdns *DECdns*

Een dienst waarmee namen kunnen worden toegekend in een gedistribueerde rekenomgeving (DCE). Deze DEC-service is door de Open Software Foundation geaccepteerd en overgenomen.

distributed object environment: DOE *gedistribueerde objectomgeving*

Een gedistribueerd object-georiënteerde applicatieraamwerk van de firma SunSoft.

distributed object management facility: DOMF *gedistribueerde faciliteit in het objectbeheer*

Een objectbeheerssysteem van de firma SunSoft dat voldoet aan de eisen die de Object Management Group (OMG) daaraan heeft gesteld.

distributed operating multi-access interactive network: DOMAIN *DOMAIN*

Het particuliere netwerkprotocol dat in (door) Apollo-werkstations wordt toegepast.

diversity *diversiteit*

Het realiseren van dezelfde functie met behulp van verschillende methoden of middelen. Bijvoorbeeld, het gebruik van diverse

processoren, opslagmiddelen, programmeertalen, algoritmen, of ontwikkelingsteams.

document *document*

1. Een gegevensmedium en de gegevens die hierop of hierin zijn vastgelegd en die over het algemeen een permanent karakter hebben. De gegevens kunnen over het algemeen zowel door de mens als door een machine worden gelezen en eventueel geïnterpreteerd.
2. Een uniforme verzameling gegevens die op een bepaald onderwerp betrekking hebben.
3. Bij het verwerken van tekst is een document een tekstdeel dat afzonderlijk kan worden aangeduid en opgeslagen.

document (to) *documenteren*

1. Het vervaardigen van een document zoals dat in de punten 1 en 3 van de vorige definitie is gedefinieerd.
2. Commentaar aan een computerprogramma toevoegen of daarin opnemen.

documentation *documentatie*

1. Het beheer van documenten. Dit kan eveneens het herkennen, verzamelen, verwerken, opslaan en verspreiden inhouden.
2. Een verzameling documenten over een bepaald onderwerp.
3. Elke op schrift gestelde of grafische informatie die activiteiten, eisen, behoeften, procedures of resultaten beschrijft, definieert, daar verslag van doet, of certificeert.
5. Het proces waarin een document wordt vervaardigd of gereviseerd.

documentation tree *documentatieboom*

Een schema dat alle documenten voor een bepaald systeem uitbeeldt en hun onderlinge relaties weergeeft.

document image processing: DIP *verwerking van afbeeldingen in documenten*

De opslag, het beheer, en het opzoeken en ophalen van afbeeldingen, plaatjes, figuren enzovoort.

document style semantics and specification language: DSSSL *semantische en specificatietaal voor de stijl van documenten*

Een ISO-standaard die in voorbereiding is en die de semantiek van de hoogwaardige presentatie en compositie van documenten behandelt. Dit geschiedt in deze standaard op een wijze die onafhankelijk is van bepaalde formatteringssystemen of processen. DSSSL is een standaard die SGML aanvult, maar specifiek bedoeld is voor de specificatie van de semantiek.

document type definition: DTD *definitie van documenttypes*

De definitie van een documenttype in SGML. DTD bestaat uit identificatie-aanhangsels (tags) voor de markering van de tekst en de daarmee geassocieerde interpretaties.

DoD-STD-2167A *DoD-STD-2167A*

Een standaard van het Amerikaanse Ministerie van Defensie (Department of Defense) waarin het volledige proces van de ontwikkeling en de documentatie van software die in kritieke omstandigheden moet worden gebruikt, is vastgelegd.

DoD-STD-2168 *DoD-STD-2168*

Een standaard van het Amerikaanse Ministerie van Defensie waarin de procedures voor kwaliteitsborging van software zijn vastgelegd.

DO-group *DO-groep*

Een reeks opdrachten die door een DO-opdracht vooraf wordt gegaan en met een END-opdracht wordt afgesloten.

DO-loop *DO-lus*

Het herhaald uitvoeren van dezelfde opdracht of reeks opdrachten die tussen een DO-opdracht en een daarbij behorende END-opdracht zijn geplaatst.

domain *domein*

1. Een duidelijk herkenbaar functioneel gebied dat door een klasse softwaresystemen met gelijksoortige eisen en mogelijkheden wordt bepaald. Een domein kan reeds bestaan voordat er sprake is van een softwaresysteem dat dat domein kan verzorgen, ondersteunen, van gegevens voorzien enzovoort.
2. De verzameling van waaruit waarden worden geselecteerd.

domain analysis: DA *domeinanalyse*

1. De analyse van systemen binnen een domein. Die analyse is bedoeld om gelijksoortige eigenschappen en verschillen tussen die systemen te ontdekken.
2. Het proces waarin informatie die in softwaresystemen wordt gebruikt, wordt geïdentificeerd, verzameld, en zodanig georganiseerd dat het kan worden gebruikt voor de creatie van nieuwe systemen binnen een domein.
3. Het resultaat van de processen beschreven onder de punten 1 en 2 van deze definitie. Domeinanalyse kan worden gezien als systeemanalyse die wordt toegepast op een klasse softwaresystemen in een domein. De belangrijkste producten van de domeinanalyse zijn het domeinmodel, en in bepaalde methoden die voor de domeinanalyse worden gebruikt ook de domeinarchitectuur.

domain analyst *domeinanalist*

De persoon die domeinanalyses uitvoert.

domain architecture *domeinarchitectuur*

Een generieke organisatorische structuur of een generiek ontwerp voor softwaresystemen in een domein. De domeinarchitectuur bevat de ontwerpen die moeten voldoen aan de eisen die in een domeinmodel zijn gespecificeerd. Een domeinarchitectuur kan (1) zodanig worden aangepast dat er softwaresystemen binnen een domein kunnen worden gecreëerd en (2) een raamwerk bieden voor het configureren van modules en assets binnen individuele softwaresystemen. In de domeinarchitectuur worden ontwerpen vastgelegd, terwijl in het domeinmodel de eisen zijn geformuleerd.

domain definition *domeindefinitie*

1. Het proces waarin de omvang (de reikwijdte) en grenzen van een domein worden vastgelegd.
2. Het resultaat van het proces dat onder punt 1. is genoemd. In het proces worden de belangrijkste functies en mogelijkheden binnen het domein vastgelegd, er wordt gespecificeerd welke functies en mogelijkheden buiten het domein worden gelaten en welke interacties er met externe domeinen (kunnen) bestaan. In het domeinontwikkelingsproces wordt de domeindefinitie opgesteld en vastgelegd. Dit gebeurt voordat een aanvang met het ontwerp van het domeinmodel wordt gemaakt.

domain description *domeinbeschrijving*

Zie domeindefinitie.

domain engineering *domeinontwikkeling*

Een op hergebruik gebaseerde benadering voor een klasse systemen, deelsystemen, of applicaties waarin de reikwijdte (de omvang) wordt gedefinieerd, de structuur (zoals de domeinarchitectuur) wordt gespecificeerd, en de assets (met hun eisen, ontwerpen, deelsystemen, code en documentatie) worden vastgelegd of gedefinieerd. Domeinontwikkeling kan de volgende activiteiten omvatten: domeindefinitie, domeinanalyse, de ontwikkeling van de domeinarchitectuur, en domeinimplementatie.

domain expert *domeinexpert*

Een persoon die zich zeer vertrouwd heeft gemaakt met het domein en de domeinanalist een gedetailleerde informatie van het domein kan verstrekken.

domain implementation *domeinimplementatie*

Het proces waarin wijzigbare assets worden gecreëerd. Deze assets kunnen opnieuw in de ontwikkeling van softwaresystemen binnen een domein worden gebruikt (hergebruik van assets). Een domeinimplemen-

tatie kan eveneens de specificatie van een software-ontwikkelingsproces inhouden. Daarin dient dan te worden beschreven hoe softwaresystemen in een domein met behulp van het hergebruik van assets worden ontwikkeld.

domain manager *domeinbeheerder*

Een persoon of een organisatie die verantwoordelijk is voor het beheer, de definitie, het gebruik, de evaluatie, en de evolutie van assets binnen een domein.

domain models *domeinmodellen*

Domeinmodellen zijn een product van de domeinanalyse. Hierin is de wijze waarop aan de eisen van het domein wordt voldaan, vastgelegd. Het domeinmodel identificeert en beschrijft de structuur van gegevens, de informatiestromen, de functies, de beperkingen, en besturingen binnen het domein dat deel uitmaakt van softwaresystemen in het domein. Het domeinmodel beschrijft de gemeenschappelijke aspecten en de verschillen van de eisen die aan de softwaresystemen in het domein worden gesteld. Het domeinmodel beschrijft de eisen, terwijl in de domeinarchitectuur het ontwerp is vastgelegd.

double precision *dubbele precisie*

De term verwijst in de specificatie van, bijvoorbeeld, de Java-taal naar de beschrijving van een drijvende-kommagetal dat 64 bits aan informatie bevat.

DO-variable *DO-variabele*

Een variabele die in een DO-opdracht wordt gespecificeerd en die wordt geïnitieerd of in waarde wordt vermeerderd om de opdracht of opdrachten binnen een DO-lus uit te voeren. Deze DO-variabele wordt gebruikt om een aantal malen dat de opdracht binnen de DO-lus wordt uitgevoerd, te besturen.

down *buiten bedrijf*

De term verwijst naar een systeem of component dat niet (meer) operationeel is of waarop geen onderhoud (meer) van toepassing is.

down time *storingstijd*

De tijdsperiode gedurende welke een systeem of component buiten bedrijf is.

downward compatible *benedenwaarts compatibel*

De term heeft betrekking op hardware of software die compatibel is met een eerdere of minder complexe versie van zichzelf; bijvoorbeeld, een programma dat bestanden kan verwerken die met een eerdere versie van dat programma zijn gemaakt.

downward compression *benedenwaartse compressie*

Een vorm van demodularisatie in het ontwerp van software waarin een sterk op volgorde geplaatst module in de code van een minder strak geordend module wordt gekopieerd.

DRAGON *DRAGON*

Een Esprit-project dat gericht is op het geven van een doelmatige ondersteuning voor het hergebruik van programma's of routines in real-time gedistribueerde Ada-applicaties.

DRAGOON *DRAGOON*

Een op Ada gebaseerde taal die in het kader van het DRAGON Esprit-project is ontwikkeld. De taal is object-georiënteerd en kan met behulp van een simultane werkwijze in gedistribueerde omgevingen worden gebruikt.

driver *driver*

1. Een softwaremodule die de uitvoering (verwerking) van één of meer andere softwaremodules initieert en soms bestuurt en controleert.
2. Een computerprogramma dat een randapparaat bestuurt en soms gegevens formatteert voor de overdracht van en naar dat apparaat.

dummy *dummy*

Dit begrip heeft betrekking op iets dat ergens op lijkt, maar dat niet de daarbij behorende eigenschappen bezit.

dummy argument *dummy-argument*

1. Een variabele binnen een functie of een subroutine of de omschrijving van een opdrachtfunctie waarmee de werkelijke argumenten van het oproepende programma of de functieverwijzing worden gekoppeld.
2. Een tijdelijk geheugengebied dat automatisch tot stand wordt gebracht om er de waarde van een argument in op te slaan.

dummy instruction *dummy-instructie*

Een gegevenselement in de vorm van een instructie die in een reeks instructies wordt ingevoegd, maar die niet bedoeld is om te worden uitgevoerd.

dummy variable *dummy-variabele*

Een symbool dat in de definitiefase wordt ingevoegd maar dat later door de werkelijke waarde wordt vervangen.

dump *dump*

1. Een lijst met een aantal aspecten betreffende de toestand en conditie van de uitvoering van een programma. De lijst bevat meestal de inhoud van het interne geheugen of de registers. Er

bestaan een aantal soorten dumps namelijk: wijzigingsdumps, dynamische dumps, geheugendumps, postmortem-dumps, selectieve dumps, statische dumps enzovoort.

2. Een afbeelding van de inhoud van een bestand of een eenheid.

dump (to) *dumpen*

1. Het kopiëren van de inhoud van een geheugen of een gedeelte daarvan. Dit gebeurt gewoonlijk van een intern naar een extern geheugen om het interne geheugen voor andere doelen of werkzaamheden te kunnen gebruiken of om de gegevens van het gedumpte geheugen tegen eventueel optredende storingen of foutieve handelingen te beschermen.
2. De vervaardiging van de lijst en/of de afbeelding genoemd onder de punten 1 en 2 van de vorige definitie.

dyadic selective construct *dyadische selectieve constructie*

Een IF-THEN-ELSE bewerking waarin de verwerking voor de beide resultaten van de sprong zijn gespecificeerd.

dynamic *dynamisch*

1. Bij programmeertalen heeft deze term betrekking op eigenschappen die alleen kunnen worden vastgelegd gedurende het uitvoeren van een programma.
2. De term kan ook van toepassing zijn op een bewerking die plaats vindt op het tijdstip dat deze nodig is in plaats van een vooraf bepaald of vast tijdstip. Voorbeelden zijn dynamische analyse en dynamische binding.

dynamic allocation *dynamische toewijzing*

Zie dynamische toewijzing van systeemelementen (dynamic resource allocation).

dynamic analysis *dynamische analyse*

Het proces waarin een systeem of een component op basis van het gedrag van het systeem of component gedurende de uitvoering ervan wordt geëvalueerd.

dynamic binding *dynamische binding*

1. Een binding die pas plaats vindt of wordt uitgevoerd nadat een verzoek daarvoor is geplaatst of ontvangen.
2. Een binding die tijdens de verwerking van een computerprogramma plaats vindt.

dynamic breakpoint *dynamisch breekpunt*

Een breekpunt waarvan de vooraf gedefinieerde initiatie een karakteristieke eigenschap van het programma is dat tijdens de werking ervan wordt gerealiseerd; zoals de verwerking van elke twintig opeenvolgende opdrachten van het programma.

dynamic buffering *dynamisch bufferen*

Een buffertechniek waarin de buffer, die aan een computerprogramma is toegewezen, tijdens de verwerking van het programma varieert; dit is afhankelijk van de behoefte aan buffercapaciteit die een programma op een bepaald moment nodig heeft.

dynamic data exchange manager *beheerssysteem voor dynamische gegevensuitwisselingen*

Een product van de firma Oracle waarmee Windows-applicaties die het dynamische uitwisselingsprotocol (DDE protocol) toepassen als toeleveranciers van gegevens voor Oracle kan laten fungeren. Het creëert de mogelijkheid om applicaties zoals Excel, Word enzovoort of informatie die in Oracle is opgeslagen, op te vragen, te wijzigen, in grafische vorm weer te geven, en af te drukken.

dynamic data exchange protocol *dynamisch gegevensuitwisselingsprotocol*

Een protocol van de firma Microsoft met behulp waarvan Windows-toepassingen via een client/servermodel onderling gegevens uit kunnen wisselen.

dynamic dump *dynamische dump*

Een dump die tijdens de werking van een programma plaats vindt. Dit geschiedt gewoonlijk onder besturing van dat programma.

dynamic error *dynamische fout*

Een fout die afhankelijk is van de in de tijd variërende aard van ingevoerde gegevens.

dynamic relocation *dynamische relocatie*

De relocatie van een programma die plaats vindt tijdens de werking van dat programma.

dynamic resource allocation *dynamische toewijzing van systeemelementen*

1. Een toewijzingsmethode waarbij de systeemelementen bestemd voor het uitvoeren van een programma worden bepaald op het moment dat deze nodig zijn.
2. Het toewijzen van systeemelementen aan een programma op het moment dat dit programma wordt uitgevoerd in plaats van op het moment dat het in het hoofdgeheugen wordt geladen.

dynamic restructuring *dynamische herstructurering*

Het proces van de herstructurering van een database, gegevensstructuur, computerprogramma, of een aantal systeemcomponenten tijdens de uitvoering van het programma.

dynamic storage allocation *dynamische geheugentoewijzing*

Een geheugentoewijzingstechniek waarin de geheugenruimte die aan een computerprogramma wordt toegekend, tijdens de uitvoering van dat programma varieert. Dit wordt gebaseerd op de momentane behoeften van het programma en die van andere programma's die op dat moment in uitvoering zijn.

dynamic subroutine *dynamische subroutine*

Een module in de vorm van een raamwerk met bepaalde kenmerken zoals het aantal herhalingen of de grootte van een gegevensbestanddeel. Deze worden geselecteerd of aangepast aan de hand van de verwerkingseisen van de desbetreffende gegevens.

E**E** *E*

Een programmeertaal voor databases die ten behoeve van het EXODUS-project aan de Universiteit van Wisconsin is ontwikkeld.

early-failure period *periode van aanvankelijke storingen*

De tijdsperiode in de levenscyclus van een systeem of component gedurende welke het aantal hardwarestoringen afneemt naarmate er meer problemen worden ontdekt en verholpen.

EAST *EAST*

Een Eureka-project waarin een platform voor software-engineering is ontwikkeld.

echo (to) *echoën*

Een uitgezonden signaal terugzenden naar de bron van herkomst. Dit geschiedt meestal met een zekere tijdsvertraging die aangeeft dat het signaal een reflectie van het oorspronkelijke signaal is.

ECIP2 *ECIP2*

Een Esprit-project voor de definitie van een specificatietaal die op het niveau van de specificatie van eisen en behoeften kan worden gebruikt.

Eden *Eden*

Een object-georiënteerd en gedistribueerd bedrijfssysteem dat op het mechanisme van de 'remote procedure call' (RPC) is gebaseerd.

edit (to) *1. opmaken; 2. aanpassen*

a. Het gereedmaken van gegevens met het doel deze verder te verwerken. Het opmaken kan het opnieuw rangschikken of controleren van de opmaak, het omzetten van de code en het toepassen

van standaard-bewerkingen (zoals het onderdrukken van nullen) zijn.

- b. Het wijzigen van de vorm of de indeling van gegevens, bijvoorbeeld het invoegen of verwijderen van tekens zoals decimaal-tekens.

editing *1. opmaking; 2. aanpassing*

- a. Bij programmeertalen verwijst de term naar het transformeren van waarden naar voorstellingen die door een bepaalde opmaak worden gespecificeerd.
- b. Aanpassen van de inhoud van een document of vorm van een programmatekst. (Dit kan ook het verwijderen van fouten uit die inhoud of tekst betekenen).

editing statement *opmaakopdracht*

Een opdracht die een specificatie geeft van de syntactische en indelingsbewerkingen die op bepaalde gegevens moeten worden uitgevoerd.

edit program *opmaakprogramma*

Een programma dat speciaal ontwikkeld is om functies te verrichten die voor het opmaken nodig zijn. Deze functies omvatten het opnieuw rangschikken, wijzigen en weglaten van gegevens aan de hand van vooraf opgestelde regels.

effective address *effectief adres*

Het adres dat het resultaat is van een index-bewerking, een indirecte adressering, of enig andere adresmodificatie op een gespecificeerd adres.

effective instruction *effectieve opdracht*

De computeropdracht die het resultaat is van een index-bewerking, een indirecte adressering, of enig andere modificatie op de adressen in een gespecificeerde computeropdracht.

efferent *efferent*

De term heeft betrekking op een gegevens- of besturingsstroom van een op volgorde geplaatst module naar een minder sterk geordend module in een softwaresysteem.

efficiency *doelmatigheid*

De mate waarin een systeem of component de daaraan toegekende functies met een minimum verbruik of gebruik aan hulpmiddelen uitvoert.

egoless programming *onzelfzuchtige programmering*

Een software-ontwikkelingstechniek die gebaseerd is op het concept van de verantwoordelijkheid van een team in plaats van die van een

individuele programmeur. Het doel van deze vorm van programmering bestaat uit het trachten te voorkomen dat individuele programmeurs zich emotioneel zo sterk met hun werk vereenzelvigen dat dit een objectieve evaluatie van hun werk in de weg zou kunnen staan.

Eiffel *Eiffel*

Een object-georiënteerde programmeertaal die door B. Meyer et al. is ontwikkeld en door de firma Interactive Software Engineering (ISE) op de markt is gebracht.

ELSE-clause *ELSE-clausule*

Het gedeelte van een IF-bewering in een IF-THEN-ELSE bewerking dat wordt gebruikt om de actie te specificeren die moet worden verricht indien de vergelijking van de opdrachtelelementen van de IF-opdracht onwaar is.

emacs *emacs*

Een populaire Unix-editor van de Free Software Foundation (FSF). De editor is voorzien van een aantal utiliteitsprogramma's.

emacs hypertext system: EHTS *emacs hypertekststysteem*

Een experimenteel hypertekststysteem voor meervoudig gebruik van de Universiteit van Aalborg. Het systeem bestaat uit een teksteditor en een grafische browser. Het systeem werkt onder het X Window systeem en OpenWindows. Beide gereedschappen benutten HyperBase als database.

embedded command *ingesloten opdracht*

Een opdracht die is gespecificeerd als een waarde in een parameter van een andere opdracht.

embedded computer system *ingebed computersysteem*

Een computersysteem dat deel uitmaakt van een groter systeem en bepaalde functies van dat systeem uitvoert, zoals dat het geval is met een computersysteem dat in een vliegtuig is geïnstalleerd.

embedded Java *1. embedded Java; 2. ingebed Java*

Een omgeving voor de uitvoering van Java-programma's die specifiek bestemd is voor ingebedde (embedded) eenheden die grote hoeveelheden gegevens verwerken.

embedded software *ingebede software*

Software die deel uitmaakt van een groter systeem en bepaalde functies van dat systeem uitvoert.

embedded system *ingebed systeem*

Computersysteem dat een integraal onderdeel van een groter systeem vormt en waarvan de belangrijkste functies meestal administratief, regelend of sturend zijn, bijvoorbeeld een computersysteem in een vliegtuig, een commando- en besturingssysteem, of een transportsysteem.

embedding *inbedding*

Het genereren van een object uit een entiteit die zelf geen object is door dit in een daarvoor geschikte schil te plaatsen.

emulate (to) *emuleren*

Het gedeeltelijk of geheel nabootsen van een bepaald computersysteem door een ander computersysteem. Dit geldt voornamelijk apparatuur waarbij het nabootsende systeem dezelfde gegevens aanvaardt, dezelfde programma's uitvoert en dezelfde resultaten oplevert als het nagebootste systeem.

emulation *emulatie*

1. Zie emuleren (to emulate).
2. Het toepassen van programmeringstechnieken en speciale machine-eigenschappen om een computersysteem programma's uit te laten voeren die voor een ander systeem zijn geschreven.
3. Een model dat dezelfde invoer accepteert en dezelfde uitvoer produceert als dat van een ander gegeven systeem.
4. Het proces waarin het model beschreven in het vorige punt van deze definitie, wordt ontwikkeld of gebruikt.

emulation program *emulatieprogramma*

1. Een programma dat een systeem of een eenheid in staat stelt te functioneren alsof het een ander systeem of een andere eenheid betreft.
2. Een besturingsprogramma dat een lokale besturingseenheid in staat stelt de functies van een andere eenheid te emuleren.

emulator *emulator*

1. Een eenheid, computerprogramma of systeem dat dezelfde invoer accepteert en dezelfde uitvoer produceert als een gegeven systeem.
2. Een combinatie van programmeringstechnieken en speciale machine-eigenschappen om een computersysteem programma's uit te laten voeren die voor een ander systeem zijn geschreven.

emulator generation *emulatorgeneratie*

Het assembleren en monteren van een emulatieprogramma in een besturingssysteem tijdens het genereren van het systeem.

encapsulation *inkapseling*

1. De term verwijst naar twee eigenschappen van objecten: onderling verwante code (methoden) en gegevens die bij elkaar worden gegroepeerd en daarmee het object vormen. De klassedefinitie kan worden gebruikt om gegevens voor externe methoden te verbergen of deze tegen dergelijke methoden te beschermen.
2. De mogelijkheid gebruikers van een goed gedefinieerde interface van een verzameling functies te voorzien en wel op een zodanige wijze dat het de interne werkwijzen van de functies verbergt. In de object-georiënteerde programmering is dit de techniek waarmee gegevensstructuren en methoden (procedures) die van die gegevens gebruik maken, bij elkaar houdt.
3. De bundeling van kennis binnen een bepaald module. Omdat objecten zowel de gegevens als de implementatie ervan bevatten (inkapselen) kan de gebruiker van een object dat object als een 'black box' die bepaalde diensten verleent, behandelen en beschouwen. Aan dit object kunnen variabelen en methoden worden toegevoegd, daaruit worden verwijderd, of daarin worden gewijzigd. Echter, zolang de diensten of functies die door het object worden geleverd hetzelfde blijven, kan de code die het object gebruikt ongewijzigd blijven en hoeft niet te worden herschreven.

encode (to) *coderen*

Het omzetten van gegevens in een andere vorm met gebruikmaking van een bepaalde code of een gecodeerd tekenset op een zodanig wijze dat het terugbrengen naar de oorspronkelijke vorm mogelijk blijft. Deze aanduiding wordt soms ook gebruikt indien het terugbrengen naar de oorspronkelijke vorm niet mogelijk is.

encoder *codeereenheid*

Een eenheid die gegevens codeert.

END-statement *END-opdracht*

Een opdracht die wordt gebruikt om het einde van een procedure, één of meer DO-lussen of een bepaalde programma-clausule aan te geven.

engineering *1. engineering; 2. ontwikkeling; 3. constructie*

Het toepassen van een systematische, gedisciplineerde en kwantificeerbare benaderings- en/of ontwerpmethodiek op structuren, machines, producten, systemen of processen.

engineering change *ontwerpwijziging*

Een verandering in de configuratie van een configuratie-item of ander benoemd item na de formele vastlegging van de configuratie-identificatie van dat item.

engineering change proposal: ECP *ontwerpwijzigingsvoorstel*

In het configuratiebeheer verwijst de term naar een voorgestelde ontwerpwijziging en de documentatie waarin de voorgestelde wijziging is beschreven.

entity *entiteit*

Elk item in de programmering van computers dat in een programma van een naam kan worden voorzien of waarnaar kan worden verwezen. Bijvoorbeeld, een gegevensitem, een programma-instructie, of een deelprogramma.

entity attribute *entiteitattribuut*

Een kenmerk of eigenschap van een ontwerpentiteit waaraan een naam is toegekend. Een entiteitattribuut bevat een feitelijke verklaring betreffende die entiteit.

entity-relationship diagram *entiteit-relatiediagram*

Een schema dat wordt gebruikt om een aantal entiteiten uit de reële wereld en de logische relaties tussen die entiteiten in grafiek- of diagramvorm weer te geven.

entity-relationship model *entiteit-relatiemodel*

Een methode voor het modelleren van gegevens waarin entiteiten en de relaties tussen die entiteiten worden gedefinieerd.

entry *ingang*

1. Een samenhangende reeks omschrijvingen die met een punt worden afgesloten.
2. Een informatie-element in een tabel, een lijst, een wachtrij of enig andere georganiseerde gegevensstructuur dan wel een informatie-element in besturingsinformatie.
3. Een enkele invoerbewerking die via een terminal wordt gerealiseerd.
4. Zie ingangspunt (entry point).

entry conditions *ingangsvoorwaarden*

1. De voorwaarden die moeten worden gespecificeerd bij het binnenkomen in een programma, een module of een subroutine, bijvoorbeeld het adres van de locaties waar het programma, het module of de subroutine de opdrachtelelementen op moet halen en van die locaties waarmee de ingangspunten en de uitgangen moeten worden gekoppeld.
2. De gegevens en de besturingscondities waaraan moet worden voldaan voordat een bepaald module kan worden uitgevoerd.

entry constant *ingangsconstante*

Een uitdrukking die bij evaluatie een ingangswaarde oplevert.

entry name *ingangsnaam*

1. Een naam binnen een besturingssectie die een ingangspunt omschrijft en waarnaar door elk besturingsgedeelte kan worden verwezen.
2. Een door de programmeur gespecificeerde naam die een ingangspunt tot stand brengt.
3. Een unieke naam voor elk bestanddeel of onderwerp dat in een catalogus wordt geïdentificeerd.

entry point *ingangspunt*

1. De eerste uitvoerbare instructie in een bestanddeel of component.
2. Elk punt in een module waarnaar vanuit een ander module kan worden gesprongen.
3. Bij een module is dit elke positie waaraan de besturing kan worden doorgegeven.
4. Een punt in een procedure waar deze kan worden opgeroepen.

entry symbol *ingangssymbool*

Een symbool dat een ingangnaam of de naam van een besturingsgedeelte representeert.

entry value *ingangswaarde*

Het ingangspunt dat door een ingangskonstante wordt gerepresenteerd.

entry variable *ingangsvariabele*

Een variabele die ingangswaarden kan representeren.

enumeration type *opsommingstype*

Een discreet gegevenstype waarvoor geldt dat de bij dat type behorende gegevens waarden aan kunnen nemen die expliciet door de programmeur zijn of worden gedefinieerd. Bijvoorbeeld, een gegevenstype dat KLEUR wordt genoemd en waarvoor de bij dat type behorende gegeven de mogelijke waarden 'rood', 'blauw' en 'geel' heeft.

environment *omgeving*

1. De omstandigheden of condities waarin en/of waaronder een softwaresysteem functioneert. Dit houdt onder andere de volgende elementen in: interfaces met een bedrijfssysteem, interfaces met andere systemen, de relatie met of afhankelijkheid van databasebeheerssystemen, hardware- of netwerkbependingen, of elke andere factor die het functioneren van het softwaresysteem beïnvloedt.
2. Alle toe te passen gereedschappen en methoden die van invloed zijn op of gebruikt worden in de ontwikkeling van een softwarecomponent, -module, of -systeem.

epilog *epiloog*

De processen die automatisch plaatsvinden bij het beëindigen van een taak of een blok in de programmeertaal PL/1.

epilog breakpoint *epiloogbreekpunt*

Een breekpunt dat bij het verlaten van een bepaald programma of een bepaalde routine wordt geïnitieerd.

Epoch *Epoch*

Een versie van Gnu-Emacs voor het X Window systeem. De versie is afkomstig van het National Center for Supercomputing Applications (NCSA).

equivalence class *equivalentieklasse*

Een deel van het invoer- of uitvoerdomein van een component waarvan wordt aangenomen dat het gedrag van het component daarin overeenkomt met dat welke in de specificaties van het component zijn vastgelegd.

equivalence partition *equivalentiepartitie*

Zie equivalentieklasse.

equivalence partition coverage *dekking van de equivalentiepartitie*

Het percentage equivalentieklassen dat voor een component wordt gegenereerd en die door een test of testserie zijn onderzocht.

equivalence partition testing *equivalentiepartitietest*

De techniek van de ontwikkeling van een test of testserie voor een component waarin deze testen of testseries zodanig zijn gekozen of geconstrueerd dat ze in staat zijn representatieve functies van de equivalentieklassen te kunnen evalueren.

equivalent faults *equivalente fouten*

Twee of meer fouten die dezelfde storing veroorzaken.

error *fout*

1. Het verschil tussen een berekende, waargenomen of gemeten waarde of toestand en de werkelijke, gespecificeerde of theoretisch juiste waarde of toestand.
2. Een menselijke actie die een onjuist resultaat oplevert.
3. Een onjuiste stap of gegevensdefinitie of een incorrect proces, zoals een foutieve opdracht in een computerprogramma.
4. Een onjuist resultaat.

Opmerking: hoewel alle vier definities in het algemene gebruik worden gehanteerd, bestaan er in de Engelse taal termen, zoals 'mistake', 'fault', en 'failure' die respectievelijk ook in de

definities, beschreven in de punten 2, 3, en 4, worden gebruikt.

error guessing *foutengissing*

De techniek van de constructie van een test of testserie waarin de ervaring van de testexpert wordt gebruikt om te postuleren welke fouten op kunnen treden, waarna de testen of testseries zodanig worden opgesteld dat ze deze gepostuleerde fouten op kunnen sporen.

error model *foutmodel*

Een model dat in de evaluatie van software wordt gebruikt om het aantal overblijvende fouten, de vereiste testtijd, en gelijksoortige kenmerken van een systeem te schatten of te voorspellen.

error prediction *foutvoorspelling*

Een kwantitatieve verklaring betreffende het verwachte aantal fouten of de soort van de fouten in een systeem of component.

error seeding *fouten toevoegen*

Het proces waarin opzettelijk bekende fouten aan die welke eventueel reeds in een computerprogramma aanwezig zijn, worden toegevoegd. Dit wordt gedaan met het doel om de mate waarin fouten worden gedetecteerd en verwijderd te kunnen volgen en een indruk te krijgen van het aantal fouten dat niet wordt opgespoord.

error tolerance *fouttolerantie*

De mogelijkheid van een systeem of component de normale werking te vervolgen ondanks de aanwezigheid van foutieve invoergegevens.

Esprit *Esprit*

Esprit (European Strategic Programme for Research and development of Information Technology) geeft een financiële ondersteuning aan diverse research- en ontwikkelingsprojecten die gericht zijn op IT-methoden en -technieken. Het programma is omstreeks 1985 gestart en heeft onder andere tot diverse methoden, standaarden, en programma's ten behoeve van de software-engineering geleid.

Estelle *Estelle*

Een formele beschrijvingstechniek die voor de specificatie van OSI-protocollen is ontwikkeld.

Eureka *Eureka*

Eureka is door de regeringen van diverse Europese landen in het leven geroepen om industriële research- en ontwikkelingsprojecten financieel te ondersteunen. Het wijkt in zoverre van het Esprit-programma af dat het direct op industriële activiteiten is gericht en niet specifiek de informatietechnologie tot onderwerp heeft. Het beheer van dit programma berust dan ook niet bij een EEG-instelling.

evaluation *evaluatie*

1. Het reduceren van een bewerking tot één enkele waarde; dit kan een reeks of een structuurwaarde zijn.
2. Het oplossen van een vergelijking of het bepalen van de waarde van een term, een factor, een uitdrukking enzovoort.
3. Algemeen: Het bepalen van de waarde van iets.

event *gebeurtenis*

1. Een gebeurtenis die van belang is voor een bepaalde taak of een gegeven proces, bijvoorbeeld het gereedkomen van een asynchrone bewerking zoals een in- of uitvoerbewerking.
2. Een activiteit binnen een programma waarvan de status en het gereedkomen uit een bijbehorende variabele kunnen worden bepaald.

Excelerator *Excelerator*

Een serie CASE-gereedschappen van de firma Index Technology Corporation.

exception *1. exceptie; 2. uitzondering*

Een gebeurtenis gedurende de uitvoering van een programma die er voor zorgt dat het programma de verdere uitvoering op moet schorten; dit vindt meestal in het geval van fouten plaats. Er bestaan een aantal verschillende soorten uitzonderingen, zoals adresseringsexceptie, verwerkingsexceptie, overloopexceptie, beschermingsexceptie, en onderloopexceptie. Java verwerkt uitzonderingen met de sleutelwoorden: 'try', 'catch', en 'throw'.

exception handler *exceptiecode*

Een blok code dat op specifieke uitzonderingsgevallen reageert. Indien het een uitzondering voor een fout is waarvan het programma kan worden hersteld, zal het programma de werking ervan vervolgen nadat de exceptiecode geactiveerd is geweest.

exchange format *uitwisselingsformaat*

De vorm (het formaat) van een beschrijving die wordt gebruikt om objecten, programma's, records, bestanden enzovoort tussen verschillende modules, programma's, applicaties, en systemen uit te wisselen.

executable content *uitvoerbare inhoud*

Een applicatie die vanuit een HTML-bestand wordt aangeroepen en geactiveerd.

executable file *uitvoerbaar bestand*

Een bestand dat door Unix als een programma kan worden beschouwd en geactiveerd. Een dergelijk bestand bevat binaire machine-instructies waarvan de computer weet hoe deze moeten worden uitgevoerd, of

het kan een Unix-script (een lijst met Unix shell-instructies) zijn waarvan Unix weet hoe deze moeten worden uitgevoerd.

executable program *uitvoerbaar programma*

Een programma dat gereed is om te worden verwerkt.

executable statement *uitvoerbare instructie*

1. Een instructie die een omschrijving geeft van één of meer acties die door een programma moeten worden genomen op het moment van uitvoering, zoals opdrachten om bepaalde berekeningen te maken, condities te beproeven of de besturingsstroom te veranderen.
2. Een instructie die, nadat deze is gecompileerd, in objectcode wordt vertaald en tijdens de verwerking van het programma zal worden uitgevoerd.

execute (to) *uitvoeren*

Een opdracht, instructie, proces of computerprogramma verwerken.

execute statement *uitvoeringsopdracht*

Een programmabesturingsopdracht die het begin van een verwerkingsstap aangeeft en waarin het uit te voeren programma of de te gebruiken procedure wordt opgegeven.

execution efficiency *uitvoeringsefficiëntie*

De mate waarin een systeem of component de gedefinieerde, gespecificeerde, toegewezen of toegekende functies in een zo klein mogelijk tijdsbestek uitvoert.

execution sequence *uitvoeringsvolgorde*

Bij programmeertalen verwijst de term naar de volgorde waarin de opdrachten en delen van opdrachten van een programma moeten worden uitgevoerd.

execution time *uitvoeringstijd*

1. De tijd nodig om een instructie uit het geheugen te halen, te decoderen en uit te voeren.
2. De tijd gedurende welke een programma de instructies uitvoert.

execution trace *uitvoeringstrace*

Een verslaggeving van de opeenvolgende instructies die tijdens de uitvoering van een programma worden afgewerkt. Een dergelijk verslag heeft meestal de vorm van een lijst met labels van de gecodeerde instructies (de code) die tijdens de werking van het programma worden gepasseerd.

executive program *supervisieprogramma*

Een programma dat de uitvoering van andere programma's bestuurt en dat de werkzaamheden binnen een computersysteem regelt. Een dergelijk programma is meestal een deel van een besturingssysteem.

executive routine *besturingsroutine*

Een module die de uitvoering van andere modules bestuurt en de werkzaamheden binnen een computersysteem regelt. Zie verder besturingsprogramma (executive program).

exhaustive testing *uitputtend testen*

Een techniek waarmee testen en testseries worden vervaardigd die alle combinaties van invoerwaarden en pre-condities voor variabelen van componenten omvatten.

exit *uitgang*

Een instructie in een programma, een deel van een programma of een subroutine waarbij, na uitvoering van de instructie, de besturing niet meer door die module of subroutine wordt uitgeoefend.

exit point *uitgangspunt*

De laatste uitvoerbare instructie in of van een component.

exit program *uitgangsprogramma*

Een programma dat de besturing tijdens het verwerken van een systeemfunctie krijgt.

exit routine *uitgangsroutine*

Een routine die de besturing ontvangt indien een gespecificeerde gebeurtenis, zoals een fout, optreedt.

expected outcome *verwacht resultaat*

Zie voorspeld resultaat.

expert system *expertsysteem*

Een intelligent computerprogramma dat een kennisbank, gespecialiseerde software, en een hoeveelheid algoritmen of regels bevat waarmee feiten vanuit de kennis en aangevoerde gegevens worden afgeleid en/of beredeneerd.

explicit address *expliciet adres*

1. Een adres in een programmeertaal dat de identiteit van een geheugenpositie of een eenheid bepaalt zonder gebruik te maken van een tussenverwijzing.
2. Een adres dat door de ontwerper van het systeem permanent aan een geheugenpositie wordt toegewezen.

explicit declaration *expliciete verklaring*

Het verschijnen van een identificatiesymbool in een declaratie-opdracht. In de programmeertaal PL/1 verschijnt deze expliciete verklaring als een labelvoorvoegsel of komt voor in een parameterlijst.

export (to) *exporteren*

Een beschrijving van een object naar een externe entiteit overbrengen (transporteren).

Express *Express*

Een datamodelleringstaal die door de ISO-commissie waarin STEP wordt ontwikkeld, als modelleringstaal is geaccepteerd en voor de definitie van de STEP-standaarden wordt gebruikt.

expression *uitdrukking*

1. Een configuratie van tekens.
2. Een combinatie van één of meer bewerkingen.
3. Één of meer bewerkingen die door een combinatie van tussen haakjes geplaatste termen worden voorgesteld.
4. Een notatie binnen een programma die een waarde voorstelt.
5. Een constante of een afzonderlijk voorkomende verwijzing of combinatie van constanten en verwijzingen met bewerkingstekens.

extendability *uitbreidbaarheid*

1. Het gemak waarmee een systeem of component kan worden gemodificeerd voor het vergroten van de geheugenruimte of de functionele capaciteit van dat systeem of component.
2. Zie ook de definitie van uitbreidbaarheid onder 'extensibility'.

extendable database *uitbreidbare database*

Een database waarvan het beheerssysteem de mogelijkheid biedt gegevens van andere (en eventueel op andere locaties geplaatste) bronnen te benaderen op een wijze die de indruk wekt alsof deze bronnen een deel van de database zelf zijn.

extended systems modelling language: ESML *uitgebreide systeem-modelleringstaal*

Een methodologie voor de software-engineering van real-time systemen. De methodologie is gebaseerd op RTSA (real-time structured analysis).

extends *breidt uit*

Klasse X breidt klasse Y uit met het doel daar functionaliteiten aan toe te voegen. Dit geschiedt door òf velden of methoden aan klasse Y toe te voegen òf methoden van klasse Y te overschrijven. Een interface breidt een andere interface uit door daar methoden

aan toe te voegen. Van klasse X wordt dan gezegd dat het een deelklasse (subklasse) van klasse Y is.

extensibility *uitbreidbaarheid*

1. De term verwijst naar de mogelijkheid om de ontwikkeling en het gebruik van uitbreidingen toe te passen.
2. Een vorm van herbruikbaarheid waarin delen van herbruikbare software of systemen kunnen worden gecombineerd met het doel nieuwe interfaces en functies te creëren.

extension of a type *uitbreiding van een type*

De serie waarden die met het type kunnen worden geassocieerd of die aan de eigenschappen van het type voldoen.

external command *externe opdracht*

Een opdracht waarvan de Unix-shell niet weet hoe deze moet worden uitgevoerd. In plaats daarvan wordt een programma met dezelfde naam als de opdracht in een bestand geplaatst. Unix zal dus, indien de opdracht niet door Unix kan worden uitgevoerd en een externe opdracht aanwezig is, het programma met de naam van de opdracht starten.

external function *externe functie*

Een functie waarvan de omschrijving extern is ten opzichte van het programma of het module dat er naar verwijst.

external name *externe naam*

1. Een naam waarnaar door elk besturingsgedeelte of afzonderlijk geassembleerd of gecompileerd programma kan worden verwezen; dat wil zeggen de naam van een besturingsgedeelte of de ingangnaam van een ander module.
2. Een naam die niet beperkt is tot één blok en de blokken die zich daarin bevinden.

external procedure *externe procedure*

Een procedure die niet in een andere procedure is opgenomen.

external program parameter *externe programmaparameter*

Een parameter waaraan tijdens het oproepen van het programma een waarde moet worden toegekend.

external symbol *extern symbool*

1. De naam van een besturingsgedeelte, de ingang, of een externe verwijzing die is omschreven of waarnaar in een bepaald deel van het programma wordt verwezen.
2. Een symbool dat een externe verwijzing voorstelt.
3. Een symbool dat in de lijst van externe symbolen is opgenomen.

external unit identifier *identificatiesymbool van een externe eenheid*

Een constante of een variabele in een in- en uitvoeropdracht die een specificatie bevat van de gegevensverzameling die moet worden verwerkt of bewerkt.

extract (to) *onttrekken*

1. Uit een groep gegevens die gegevens selecteren die aan bepaalde criteria voldoen.
2. Het afscheiden van bepaalde delen van een woord van het gehele woord.
3. Het verwijderen van bepaalde gegevens uit een bestand.
4. Het opzoeken en ophalen van assets uit een bibliotheek.

extractions *onttrekkingen*

Assets die in een bibliotheek zijn opgezocht en voor hergebruik zijn benut.

F

faceted classification *gefacetteerde classificatie*

Een methode die afkomstig is uit de wetenschap der bibliotheken en die kan worden benut voor het bepalen van meervoudige toegangspaden naar herbruikbare software-assets in een bibliotheek voor hergebruik. Elk facet in de classificatie vertegenwoordigt een specifiek aspect van een software-asset zoals de functie, de omgeving waarin het asset kan worden gebruikt, of andere betekenisvolle attributen. Elke gebruiker van een bibliotheek voor hergebruik kan bij het zoeken naar relevante assets van verschillende facetten of sleutelgegevens gebruik maken.

factoring *ontbinding in factoren*

1. Het proces waarin een systeem in een hiërarchieke verzameling modules wordt opgedeeld.
2. Het proces waarin een functie uit een module wordt verwijderd en in een apart module voor die functie wordt geplaatst.

factory *fabriek*

De term verwijst naar een dienstverlening of dienstverlenende instantie voor de constructie van nieuwe objecten.

fail safe *storingsveilig*

De term verwijst naar een systeem of een component dat in het geval van een storing automatisch een veilige werkwijze aanneemt. Dit is, bijvoorbeeld, het geval met verkeerslichten die in alle richtingen gaan knipperen indien de normale werkwijze wordt verstoord.

fail soft *zachte storing*

De term heeft betrekking op een systeem of component dat gedeeltematig betrouwbaar blijft werken in het geval van bepaalde storingen.

failure *storing*

1. Een afwijking van de software van de gespecificeerde functionaliteit of verwachte oplevering.
2. De situatie die optreedt indien een systeem of component de functies niet meer binnen de gespecificeerde prestatie-eisen uit kan voeren. Opmerking: Een fouttolerante discipline maakt een onderscheid tussen een foutieve menselijke actie, de manifestatie van de storing, het resultaat van de storing en de grootte (of omvang) van dat foutieve resultaat.

failure mode *storingstoestand*

De fysieke of functionele manifestatie van een storing. Bijvoorbeeld, een systeem in een storingstoestand kan een vertraagde werking, foutieve resultaten, of het compleet afbreken van de werking tot gevolg hebben.

failure rate *storingsfrequentie*

De verhouding tussen het aantal storingen van een bepaalde categorie en een gegeven maateenheid; bijvoorbeeld, storingen per tijdseenheid, storingen per aantal transacties, storingen per aantal computeropdrachten.

fatal error *fatale fout*

Een fout die het resultaat is van de algehele onmogelijkheid van een systeem of component te kunnen (blijven) functioneren.

fault *fout*

1. Een manifestatie van een tekortkoming in de software. Een fout kan, indien deze optreedt, een storing tot gevolg hebben.
2. Een foutieve stap, gegevensdefinitie of foutief proces in een computerprogramma. Opmerking: In de Engelse taal worden voor een softwarefout de termen 'error' en 'bug' gebruikt.
3. Een defect in een hardware-eenheid of -component.

fault dictionary *foutendictionaire*

Een lijst met de fouten in een systeem of component en de testen die zijn ontwikkeld om deze fouten op te sporen.

fault masking *foutmaskering*

Een toestand waarin een bepaalde fout de detectie van een andere fout onmogelijk maakt.

fault secure *foutveilig*

De term heeft betrekking op een systeem of component waarin fouten niet het gevolg zijn van een gespecificeerde reeks fouten.

fault tolerance *fouttolerantie*

1. De ingebouwde mogelijkheid van een systeem om voortdurend juiste resultaten af te leveren ondanks de aanwezigheid van fouten in het systeem of het falen van de normale werking van bepaalde softwarecomponenten.
2. De potentie van een systeem of component de normale werking te vervolgen ondanks de aanwezigheid van hardware- of softwarefouten.
3. Het aantal fouten dat een systeem of component kan hebben voordat de normale werking daar last van ondervindt.
4. De term verwijst naar de bestudering van fouten en storingen, alsmede methoden die systemen in staat stellen de normale werking bij de aanwezigheid van fouten te vervolgen.

fault tolerant *fouttolerant*

De term heeft betrekking op een systeem of component dat de normale werking ondanks de aanwezigheid van fouten kan vervolgen.

feasibility *haalbaarheid*

De mate waarin de eisen, de behoeften, het ontwerp, of de plannen voor een systeem of component onder bestaande beperkingen kunnen worden geïmplementeerd.

feasible path *uitvoerbaar pad*

Een pad (in een programma) waarvoor een serie invoerwaarden en uitvoeringscondities bestaan en een verwerking van die invoerwaarden mogelijk maakt.

feature *1. kenmerk; 2. functie; 3. kenmerkende eigenschap*

1. Een attribuut of een karakteristieke eigenschap van een systeem die een zekere betekenis heeft voor of beïnvloed wordt door de gebruiker, de ontwikkelaar, of enig andere entiteit die met het systeem communiceert.
2. Een attribuut of functie van een klasse in een Eiffel-programma.
3. Zie karakteristieke software-eigenschap (software feature).

feed forward *voorwaartse koppeling*

Een perceptron-netwerk met meervoudige lagen waarin de uitvoer van alle neuronen naar de volgende, maar niet naar de voorgaande lagen gaan. Er zijn dus geen teruggekoppelde lussen.

fetch (to) *ophalen*

Het lokaliseren van computerinstructies of gegevens en het laden van die instructies en gegevens in het geheugen of de registers van een computer vanuit een ander geheugen of opslagmedium.

field *veld*

1. Een lid van een klasse die gegevens bevat. Tenzij dit anders is gespecificeerd, geldt dat een veld niet statisch is.
2. De kleinste gegevenseenheid in een database waarnaar kan worden verwezen.
3. Een logische zone binnen een constructie.
4. Een gespecificeerd gebied in een record dat voor een bepaalde categorie bij elkaar behorende gegevens wordt gebruikt.
5. Het gedeelte van de gegevensstroom waarvan de breedte (of lengte), uitgedrukt in het aantal tekens, door één enkel gegevensopmaakbestanddeel wordt omschreven.

fifth generation language: 5GL *vijfde generatietaal*

Een computertaal waarin de concepten van op kennis-gebaseerde systemen, expertsystemen, inferentiemachines en de natuurlijke taalverwerking zijn opgenomen.

figurative constant *figuratieve constante*

De naam van een gegeven die voor een specifieke constante in een programmeertaal wordt gereserveerd. Zo kan de naam DRIE gereserveerd worden (of blijven) voor de waarde 3.

file *1. file; 2. bestand*

- a. Een hoeveelheid informatie die als één geheel is opgeslagen en van een naam is voorzien. Een bestand kan tekst, programma's, of gegevens in een bepaald formaat bevatten.
- b. Een verzameling bij elkaar behorende gegevens, records of blokken die als een eenheid wordt behandeld. Zo kan een regel van een factuur een gegevenselement, een volledige factuur een record, en een volledige verzameling facturen een bestand vormen.

file attribute *bestandskenmerk*

Een kenmerk dat de eigenschappen van een bestand beschrijft.

file constant *bestandsconstante*

Een naam die voor een bestand wordt vastgesteld en waarvoor een complete serie bestandskenmerken gedurende de tijd dat het bestand open is, bestaat.

file management system *bestandsbeheerssysteem*

Een programma dat is ontwikkeld om bestanden samen te stellen of te produceren en te beheren; het zorgt eveneens voor het toekennen van geheugenruimte.

file name *bestandsnaam*

Een naam die aan een reeks in- of uitvoergegevens wordt toegekend. In bepaalde programmeertalen gelden regels voor dit soort namen zoals het moeten bevatten van tenminste één alfabetisch teken.

file structure *bestandsstructuur*

De organisatie van gegevensitems of -velden in een bestand.

file system *bestandssysteem*

Een serie bestanden die op een schijf of een partitie van een schijf is opgeslagen. Een bestandssysteem heeft een wortel-directory (index) waarin de bestanden en de subdirectories zijn opgeslagen. Een dergelijke subdirectory of subindex kan eveneens bestanden en andere directories bevatten.

file transfer *bestandsoverdracht*

Het kopiëren van bestanden van een bepaald computersysteem naar een ander.

file transfer, access, and management: FTAM *bestandsoverdracht, -toegang, en -beheer*

Een protocol van een applicatielaag en gegevensverwerkingen die op afstand kunnen worden uitgevoerd.

file transfer program: FTP *bestandsoverdrachtprogramma*

1. Een programma dat een gebruiker in staat stelt vanaf de eigen computer een andere computer te benaderen, daar op in te loggen en na het realiseren van een communicatieverbinding bestanden tussen de eigen computer en de andere uit te wisselen.
2. FTP is het basis Internet bestandsoverdrachtprogramma. Het is gebaseerd op TCP/IP. Met behulp van FTP kunnen bestanden tussen gastheercomputers via het Internet worden opgehaald en opgeslagen.
3. FTP is eveneens de naam van een utiliteitsprogramma dat op verschillende computers aanwezig is en het bestandsoverdrachtprogramma van Internet gebruikt om contact te maken met andere computers en daar bestanden mee uit te wisselen.

file variable *bestandsvariabele*

Een variabele waaraan bestandsconstanten kunnen worden toegekend.

filter *filter*

Een klein Unix-programma dat wordt gebruikt voor het doorsturen, verzenden, of herinrichten van de invoer en de uitvoer. De meest gebruikte filters zijn de 'more'- en 'sort'-opdrachten.

finite state machine: FSM *eindige-toestand machine*

Een rekenmodel dat uit een eindig aantal toestanden en overgangen tussen die toestanden bestaat. Die overgangen kunnen vergezeld gaan van gedefinieerde acties.

firmware *firmware*

1. De combinatie van een hardware-eenheid en computerinstructies en gegevens die als lees-alleen-software in die eenheid is geplaatst. Opmerking: Deze term wordt soms òf alleen voor de hardware-eenheid òf alleen voor de computerinstructies en gegevens gebruikt.
2. Programma's die in lees-alleen-geheugens of -eenheden zijn of worden opgeslagen. Dit is als het ware programmatuur in apparatuur of software in hardware.
Opmerking: De verwarring rondom het gebruik van deze term heeft er toe geleid dat door velen het gebruik van de term wordt vermeden.

first customer ship: FCS *eerste aflevering naar de klant*

De dag waarop een product wordt vrijgegeven of naar de klant wordt verzonden.

flag *vlag*

Een variabele die in een voorgeschreven toestand wordt geplaatst. Die toestand is meestal 'waar' of 'onwaar' en is afhankelijk van de resultaten van een proces of de occurrence van een gespecificeerde toestand.

flexibility *flexibiliteit*

Het gemak waarmee een systeem of component kan worden gemodificeerd met het doel een dergelijk gemodificeerd systeem of component toe te kunnen passen of te kunnen gebruiken in applicaties of omgevingen anders dan die waarvoor het speciaal was ontworpen.

Floppy *Floppy*

Een systeem voor het controleren van de coderingsconventies in Fortran. De laatste versie bevat een functie waarmee HTML-code kan worden gegenereerd.

flow analysis *stroomanalyse*

1. Een techniek in compileerprogramma's om de onderlinge afhankelijkheid van de verschillende elementen van een programma te bepalen.
2. Het opsporen en vastleggen van de opeenvolging van instructies in een programma.

flowchart *stroomschema*

Een grafische weergave van de omschrijving, analyse of oplossing van een probleem. Hierin worden symbolen gebruikt om bewerkingen, gegevens, stroomrichting, apparatuur en specifieke functies aan te duiden. Een stroomschema is een logische schakel tussen de specificatie van algoritmen en de programmering daarvan. Het kan het opsporen van fouten vergemakkelijken omdat het programma in het stroomschema in logische modules is opgesplitst.

flowcharter *stroomschemaprogramma*

Een softwaregereedschap dat als invoer een ontwerp- of coderepresentatie van een programma heeft en als uitvoer een stroomschema van dat programma produceert.

flow diagram *stroomdiagram*

Zie stroomschema (flowchart).

folder *folder*

Een bestand dat elektronische mail bevat.

foreground *voorggrond*

1. De partitie van het adresseerbare geheugen waarin bij de multiprogrammering programma's met een grotere prioriteit worden uitgevoerd.
2. De partitie waarin programma's afwisselend in het hoofdgeheugen worden gehaald en daaruit worden verwijderd zodat de verwerkingstijd onder de verschillende gebruikers kan worden verdeeld. Alle commandoverwerkende programma's worden in de voorgrond uitgevoerd.
3. De computeromgeving waarin processen met een hoge prioriteit of processen waarin interacties met gebruikers nodig zijn, worden uitgevoerd.

foreground processing *voorggrondverwerking*

De verwerking van een proces met hoge prioriteit terwijl processen met lagere prioriteiten de beschikbaarheid van de computerelementen en -middelen af moeten wachten, dan wel de verwerking van processen waarvoor interacties met gebruikers nodig zijn.

foreground program *voorggrondprogramma*

Een programma dat op een gegeven moment wordt uitgevoerd en een communicatie met de terminal van de gebruiker onderhoudt.

Foresight *Foresight*

Een softwareproduct dat grafische modelleringsgereedschappen bevat die kunnen worden gebruikt voor het ontwerp en de simulatie van hogere-niveau systemen.

fork *vork*

Indien Unix een nieuw proces start, wordt een bestaand proces gekloond. Een dergelijk proces wordt in het Unix-jargon 'forking' genoemd.

formal language *formele taal*

Een taal waarvan de regels voor het gebruik van de taal expliciet zijn gedefinieerd. Voorbeelden hiervan zijn programmeertalen en op de wiskunde gebaseerde talen.

formal parameter *formele parameter*

1. Een variabele die in een softwaremodule wordt gebruikt om gegevens of programma-elementen te representeren die naar deze module vanuit een oproepend module moeten worden doorgegeven.
2. Een taalobject in programmeertalen waarvan het identificatiesymbool in een ingang van een procedure verschijnt. De formele parameter is gekoppeld aan de actuele parameter die door de procedure-oproep is gespecificeerd met het doel deze bij het oproepen van de procedure te gebruiken.

formal parameter list *formele parameterlijst*

De parameters die in de definitie van een specifieke methode zijn gedefinieerd.

formal proof *formeel bewijs*

Een methode met behulp waarvan kan worden bewezen dat een programma aan de specificaties van het programma voldoet en waarbij van wiskundige regels gebruik wordt gemaakt.

formal qualification review: FQR *formele kwalificatiebeoordeling*

De test, de inspectie, of het analytische proces waarmee of waarin een groep configuratie-items die een systeem definiëren, op de conformantie van die groep met specifieke contractuele prestatie-eisen wordt geverifieerd.

formal specification *formele specificatie*

1. Een specificatie die in overeenstemming met geaccepteerde standaarden is geschreven en goedgekeurd.
2. Een specificatie die in een formele notatie is opgesteld. Een dergelijke specificatie wordt vaak gebruikt om de juistheid van datgene wat de specificatie omschrijft te bewijzen.

formal test *formele test*

Een test die in overeenstemming met testplannen en procedures, die door een klant, door een gebruiker, of in het daarvoor aangewezen managementniveau zijn beoordeeld en goedgekeurd, wordt verricht.

format 1. *formaat*; 2. *opmaak*

De rangschikking en structuur van de gegevens op een gegevensdrager of in een in te voeren, op te slaan, of uit te voeren bestand.

form, fit, and function *vorm, geschiktheid en functie*

Die configuratie in het configuratiebeheer die de fysieke en functionele eigenschappen van een item als entiteit bezit, maar geen van de eigenschappen of kenmerken van de elementen heeft waaruit dat item is samengesteld of opgebouwd.

Forth *Forth*

Een programmeertaal voor wetenschappelijke toepassingen die uit op te roepen modules bestaat. In deze taal wordt gebruik gemaakt van de prefixnotatie.

Fortran *Fortran*

Een programmeertaal die gedurende een lange tijd voor wetenschappelijke applicaties werd gebruikt. De taal werd ook toegepast in het schrijven van applicatiesystemen.

forward delta *voorwaartse delta*

De delta waarmee in samenhang met een versie, een zoonversie kan worden gecreëerd. Zie wijzigingsbeheer (change management) en zoonversie (child version).

forward engineering *voorwaartse ontwikkeling*

Het traditionele proces waarin men van hoog-niveau abstracties en logische, implementatie-onafhankelijke ontwerpen naar de fysieke implementatie van een systeem toewerkt.

forward recovery *voorwaarts herstel*

1. De reconstructie van een bestand naar een gegeven toestand. Dit wordt gerealiseerd door een eerdere versie bij te werken en aan te passen, waarbij gegevens worden gebruikt die in een chronologisch verslag van de wijzigingen die in het bestand zijn aangebracht, zijn opgenomen.
2. Een soort herstel waarin een systeem, programma, database, of ander systeemelement naar een nieuwe, nog niet eerder bestaande toestand wordt gebracht waarin het de vereiste functies uit kan voeren.

four-address instruction *instructie met vier adressen*

Een computerinstructie die vier adresvelden heeft. Bijvoorbeeld, een instructie die de inhoud van de locaties A, B, en C optelt en het resultaat in locatie D plaatst.

four-plus-one address instruction *instructie met vier adressen en een extra adres*

Een computerinstructie die vijf adresvelden heeft, waarbij het vijfde adresveld het adres bevat van de volgende instructie die moet worden uitgevoerd. Bijvoorbeeld, een instructie die de inhoud van locaties A, B, en C optelt en het resultaat daarvan in locatie D plaatst, waarna de instructie in locatie E wordt uitgevoerd.

fourth generation language: 4GL *vierde-generatietaal*

1. Een hoog-niveautaal die onervaren gebruikers in staat stelt database-toepassingen te ontwikkelen en te programmeren. Een vierde-generatietaal is meestal niet-procedureel.
2. Een computertaal waarmee de productiviteit van het werken met derde-generatietalen kan worden verbeterd en vaak het computergebruik van onervaren gebruikers mogelijk maakt. Functies en elementen van een vierde-generatietaal omvatten een geïntegreerd databasemanagementsysteem, een opvraagtaal, een rapportgenerator, en schermdefinities. Daarnaast komen eveneens een grafische generator, een functie voor het ondersteunen van beslissingen, financiële modelleringsmogelijkheden, een spreadsheetprogramma, en statistische analysefuncties in een dergelijke taal voor.

framework *raamwerk*

Een verzameling klassen in de object-georiënteerde programmering die een abstract ontwerp voor oplossingen van een aantal onderling aan elkaar verwante problemen belichaamt.

Free Software Foundation: FSF *FSF*

Een stichting die de ontwikkeling en het gebruik van vrij verkrijgbare software ondersteunt en bevordert. Die software geldt met name het Gnu-systeem.

Fresco *Fresco*

Een object-georiënteerde applicatieprogramma-interface (API) voor grafische gebruikersinterfaces. Fresco wordt ontwikkeld door het X consortium als een open standaard die door een groot aantal leveranciers zal kunnen worden gehanteerd.

friend *vriend*

Een bepaalde relatie tussen klassen in de C++-taal.

function *functie*

1. Een specifiek doel of karakteristieke werking van een eenheid.
2. Een procedure die wordt opgeroepen als gevolg van het verschijnen van één van de bijbehorende ingangsnamen in een functieverwijzing en die tot gevolg heeft dat er een waarde naar het verwijzingspunt wordt teruggestuurd.
3. Een module die de waarde van één enkele variabele terugstuurt en die gewoonlijk één enkele uitgang heeft. Voorbeelden

hiervan zijn de wiskundige functies zoals sinus, logaritme, wortel enzovoort.

4. Een gedefinieerd doel of karakteristieke actie van een systeem of component.
5. Een softwaremodule die een specifieke actie uitvoert, door het verschijnen van de naam in een uitdrukking wordt geactiveerd, invoerwaarden kan ontvangen, en één waarde af kan leveren.

functional baseline *functionele voetlijn*

1. De versie van een asset die (in het voetlijndocument) wordt vastgelegd na de initiële voltooiing van de definitie van (1) de functies van het softwaresysteem en de daarmee geassocieerde gegevens; (2) de kenmerken en eigenschappen van de interfaces; (3) de functionele eigenschappen van de belangrijkste configuratie-items; en (4) de testen die nodig zijn voor het aantonen van de aanwezigheid van elke afzonderlijk gespecificeerde eigenschap. Een dergelijke voetlijn wordt gewoonlijk door de klant beheerd.
2. De term verwijst in het configuratiebeheer naar de initieel goedgekeurde (vrijgegeven) documentatie van een configuratie-item.

functional cohesion *functionele cohesie*

Een soort cohesie waarin de taken die door een softwaremodule worden verricht allen bijdragen tot de prestatie van één functie.

functional configuration audit *functionele configuratiecontrole*

Een administratieve controle waarin wordt nagegaan of de ontwikkeling van een configuratie-item naar voldoening is afgesloten, of het item de vereiste of gewenste prestaties levert en de functionele eigenschappen bezit die in de functionele of toegewezen configuratie-identificatie zijn gespecificeerd, en of de operationele en ondersteunende documenten compleet en voldoende zijn.

functional configuration identification *functionele configuratie-identificatie*

In het configuratiebeheer heeft de term betrekking op de momentane technische documentatie van een configuratie-item. Deze documentatie bevat de volgende onderwerpen: alle noodzakelijk geachte functionele eigenschappen en kenmerken; de testen die nodig zijn om de gespecificeerde en gerealiseerde functionele eigenschappen en kenmerken te kunnen demonstreren; de noodzakelijk geachte interface-eigenschappen en -kenmerken met verwante configuratie-items; de belangrijkste functionele eigenschappen en kenmerken van het configuratie-item en de belangrijkste lagere-niveau configuratie-items daarvan; en, voor zover aanwezig of van toepassing, de ontwerpbeperkingen.

functional decomposition *functionele decompositie*

Een soort modulaire decompositie waarin een systeem wordt opgebrouwen in componenten die met de systeemfuncties en deelfuncties corresponderen.

functional design *functioneel ontwerp*

1. Eén van de fasen bij de ontwikkeling van een automatiseringssysteem. In het functionele ontwerp worden de te automatiseren functies probleemgericht en gedetailleerd beschreven. De wijze waarop deze functies binnen het systeem zullen worden uitgevoerd, worden globaal aangegeven.
2. Het resultaat van het proces van de in punt 1 beschreven definitie.

functional diagram *functieschema*

Een schema dat de functionele relaties tussen de verschillende delen van een systeem weergeeft.

functional interfaces *functionele interfaces*

Interfaces die de functies definiëren die door gebruikers van een objectservice worden geïnitieerd. Deze interfaces vertegenwoordigen de functionaliteit (de nuttig bruikbare functies, werkwijze, en mogelijkheden) van de service.

functional language *functionele taal*

Een algemeen bruikbare hogere programmeertaal die gebaseerd is op de wiskundige notaties van functies. Een functioneel programma bestaat uit een verzameling (mogelijk recursieve) functiedefinities. De uitvoering ervan bestaat uit de evaluatie van de functies. Programma's die in een functionele taal zijn geschreven, zijn doorgaans compact en elegant. Ze hebben echter de neiging tamelijk langzaam in de verwerking te zijn en gebruiken over het algemeen veel geheugenruimte.

functional programming *functionele programmering*

Programmering die in een functionele taal wordt uitgevoerd.

functional requirement *functionele eis*

Een eis waarin een functie, die een systeem of systeemcomponent uit moet kunnen voeren, is gespecificeerd.

functional specification *functionele specificatie*

Het document waarin in detail is vastgelegd wat de eigenschappen van het product met betrekking tot de vereiste of gewenste functionaliteit zijn.

functional test *functionele test*

Een diagnostisch programma dat wordt gebruikt om de betrouwbaarheid van het totale systeem te testen.

functional test case design *ontwerp van een functioneel testproces*

De keuze van de testen, de testseries of overige beproevingsmogelijkheden die gebaseerd is op de analyse van de specificatie van het component, zonder dat daarbij naar de interne werkwijze van het component wordt verwezen.

functional testing *functioneel testen*

1. Testactiviteiten waarin het interne mechanisme van een systeem of component wordt genegeerd en waarin de aandacht uitsluitend wordt gericht op de uitvoer die als het resultaat van geselecteerde invoerwaarden en verwerkingscondities wordt gegeneereerd.
2. Testactiviteiten die worden verricht om de compliantie van een systeem of component met gespecificeerde functionele eisen te kunnen verifiëren.

function name *functienaam*

Een naam die een omschrijving geeft van logische eenheden, besturingstekens van, bijvoorbeeld, afdrukapparatuur, codes voor rapporten enzovoort. Indien een functienaam gekoppeld is aan een mnemonische naam kan deze mnemonische naam worden gesubstitueerd in elke opmaak waarin deze substitutie geldig is.

function procedure *functieprocedure*

Een procedure in programmeertalen die een bepaalde waarde oplevert en waarvan de procedure-oproep als een opdrachtelelement in een uitdrukking kan worden gebruikt. Zo levert de sinusfunctie de waarde van $\sin(x)$ op indien deze als deel van het programma wordt opgeroepen.

function reference *functieverwijzing*

Het verschijnen van een ingangsnaam, een naam van een ingebouwde functie of een ingangsvariabele in een uitdrukking in bepaalde programmeertalen.

function subprogram *functie-subprogramma*

Een externe functie die door een bepaalde opdracht vooraf wordt gegaan. De functie verzorgt het terugsturen van een waarde naar de oproepende programma-eenheid op het verwijzingspunt.

Fuse *Fuse*

Een software-ontwikkelingsomgeving voor ULTRIX van de firma DEC. Het bestaat uit een geïntegreerde gereedschapskist voor de ontwikkeling, het testen, het opsporen van fouten en het onderhoud van software.

G

garbage collection *sanering*

1. De automatische detectie en vrijgave van geheugen dat niet langer wordt gebruikt. Zo voert het Java-systeem tijdens de werking van het systeem een sanering uit zodat programmeurs nooit expliciet objecten vrij behoeven te maken.
2. Het proces waarin de geheugenruimte die niet wordt gebruikt, wordt opgeëist.
3. In het beheer van de computerelementen en -middelen is deze term synoniem met het creëren van compactheid.

garbage collector *saneerder*

Een programma waarmee saneringen kunnen worden uitgevoerd.

gateway *gateway*

Een verbinding tussen twee netwerken die meestal onderling verschillende eigenschappen hebben.

generality *alzijdigheid*

De mate waarin een systeem of component functies die een breed gebied bestrijken, uit kan voeren.

generate (to) *genereren*

1. Het vervaardigen van een programma door middel van de selectie van deelverzamelingen uit een raamwerk van coderingen. Dit vindt over het algemeen plaats onder de besturing van daarvoor geselecteerde parameters.
2. Het vervaardigen van opdrachten in een assembleer-codering op basis van modelopdrachten van macro-definities wanneer deze door macro-instructies worden opgeroepen.
3. Algemeen: vervaardigen, doen ontstaan, fabriceren, doen verwerken enzovoort.

generated address *gegenereerd adres*

Een adres dat tijdens de uitvoering van een computerprogramma is of wordt berekend.

genericity *algemeenheid*

De mogelijkheid van een taal in geparameteriseerde types of modules te kunnen voorzien. Bijvoorbeeld List(of:Integer) of List(of:People).

generic markup *generieke notatie*

Een methode die wordt toegepast in de opmaak van gecomputeriseerde documenten waarbij informatie aan de tekst wordt toegevoegd. Die informatie identificeert logische componenten van het document

zoals paragrafen, aanheffingen, voetnoten enzovoort. Specifieke instructies met betrekking tot de opmaak van de tekst op een pagina behoeft niet in de notatie voor te komen.

generic object *generiek object*

Een object waarvan de belangrijkste bestaansreden geen verband houdt met de objectdiensten (van de interface) van het object.

generic operation *generieke bewerking*

Een bewerking wordt generiek genoemd indien het op meer dan één enkele methode van toepassing kan zijn.

generic program unit *generieke programma-eenheid*

Een softwaremodule die op zeer algemene wijze is gedefinieerd en die de substitutie van specifieke gegevens, instructies of beide nodig heeft om in een computerprogramma te kunnen worden gebruikt.

generic reusable Ada components for engineers: GRACE *generieke herbruikbare Ada-componenten voor ontwikkelaars*

Een bibliotheek met meer dan 300 herbruikbare softwarecomponenten die op algemeen gebruikte en bruikbare gegevensstructuren zijn gebaseerd. Elke gevalideerde Ada-compiler kan met deze componenten werken.

ghostscript *ghostscript*

Het vertolkingsprogramma voor Gnu-postscript.

ghostview *ghostview*

Een X Window interface voor het ghostscript-vertolkingsprogramma.

glass box *glazen doos*

1. Een systeem of component waarvan de inhoud of implementatie bekend is.
2. De term verwijst naar een benadering die een systeem of component behandelt zoals in punt 1 van deze definitie is beschreven.

global compaction *globale compactheid*

Compactheid in de microprogrammering waarin micro-bewerkingen voorbij de grenzen van de sequentiële blokken met een enkelvoudige ingang/uitgang waarin ze voorkomen, kunnen worden verplaatst.

global data *globale gegevens*

Gegevens die voor twee of meer niet-geneste modules van een computerprogramma toegankelijk zijn zonder dat ze daarbij expliciet als parameters tussen de modules worden uitgewisseld.

global variable *globale variabele*

1. Een variabele die in een bepaald gedeelte van een programma is omschreven en in minstens één ander gedeelte van dat programma wordt gebruikt.
2. Een variabele die voor twee of meer niet-geneste modules van een computerprogramma toegankelijk is zonder dat deze daarbij expliciet als een parameter tussen de modules wordt uitgewisseld.

Gnu's not Unix: GNU *GNU*

Een populaire serie overdraagbare softwareprogramma's en -systemen van de Free Software Foundation. Deze software is in bovenwaartse richting compatibel met Unix.

go to *goto*

Een opdracht in een computerprogramma die een sprong tot gevolg heeft. Prof. Edsger Dijkstra heeft als eerste programmeurs op het hart gedrukt deze instructie te vermijden. Het gebruik van de 'goto' maakt programma's onleesbaar en onbewijsbaar. Zijn opmerking is in de jaren zestig in het vaktijdschrift 'Communications of the ACM' als een ingezonden brief opgenomen onder de titel: 'Goto's considered harmful'.

grammar *1. grammar; 2. grammatica*

1. Grammar is een wiskundig-georiënteerde methodiek voor de definitie van een taal.
2. De methode waarmee aan de zinnen in een taal een begrijpelijke structuur wordt gegeven.

graph *1. graaf; 2. grafiek; 3. schema; 4. diagram*

1. Een diagram of andere representatievorm die uit een eindige verzameling knooppunten en knooppuntverbindingen bestaat.
2. Een diagram dat de variaties van een variabele vergeleken met die van één of meer andere variabelen representeert; bijvoorbeeld, een grafiek die de vorm van een badkuip of een klokvorm heeft.
3. Zie 2.
4. Zie 2.

graphical user interface: GUI *grafische gebruikersinterface*

Een interface die de gebruiker in staat stelt om gegevens en informatie die op een beeldscherm worden afgebeeld te identificeren. Dit geschiedt, bijvoorbeeld, door van een muis in plaats van het toetsenbord gebruik te maken. De meest bekende GUI's voor Unix zijn Motif, Open Look, en het X Window systeem.

graphics interchange format: GIF *grafisch uitwisselingsformaat*

Een standaard voor gedigitaliseerde beelden die met behulp van het LZW-algoritme zijn gecomprimeerd.

GRAS *GRAS*

Een vrij verkrijgbaar graaf-georiënteerd databasesysteem voor toepassingen in de software-engineering. Het systeem is ontwikkeld aan de RWTH (Rheinland-Westfalische Technische Hochschule) te Aken.

GRASPIN *GRASPIN*

Een Esprit-project waarin een omgeving ten behoeve van de software-engineering voor persoonlijk gebruik wordt ontwikkeld. Het systeem moet de constructie en verificatie van gedistribueerde en niet-sequentiële softwaresystemen ondersteunen.

group item *groepsitem*

Een gegeven dat is samengesteld uit een reeks logisch met elkaar verbonden elementaire bestanddelen. Het kan een record of een gedeelte van een record zijn.

guidelines for the definition of managed objects: GDMO *richtlijnen voor de definitie van beheerste objecten*

Een ISO/IEC-standaard die van toepassing is op het definiëren van gegevensmodellen in ASN.1 (abstracte syntaxnotatie 1).

gunzip *gunzip*

Het gzip-programma voor het decomprimeren van gegevens.

gzip *gzip*

Een programma voor het comprimeren van gegevens. Het programma is met de Gnu-software geassocieerd.

H

h *h*

Een eenvoudige notatietaal (markup language) die bestemd is voor een snelle conversie van bestaande tekst naar hypertext.

halt *halt*

1. De term is meestal synoniem met 'stop'.
2. Soms is deze term een synoniem voor 'pauze'.

handle *handvat*

Een waarde die een object ondubbelzinnig identificeert.

hard failure *harde storing*

Een storing die tot een complete beëindiging van de werkzaamheden of verwerkingen in of van een systeem leidt.

hardware 1. *apparatuur*; 2. *hardware*

De fysieke componenten van een computersysteem. Die componenten zijn onder andere de computer, de terminal, het toetsenbord, het scherm, de modem, de printer(s), de muis, de volgbol, de schijf-eenheden, een scanner enzovoort.

hardware configuration item: HWCI *hardwareconfiguratie-item*

Een aggregatie van hardware die deel uit gaat maken van het configuratiebeheer en als een enkelvoudige entiteit in het configuratiebeheersproces wordt behandeld.

hardware description language: HDL *hardwarebeschrijvingstaal*

Een taal die wordt gebruikt voor het conceptuele ontwerp van geïntegreerde componenten.

hardware design language: HDL *hardware-ontwerptaal*

Een taal met speciale constructies die soms voorzien is van verificatieprotocollen en wordt gebruikt om een hardware-ontwerp te concipiëren, te analyseren, en te documenteren.

hardware loop *hardwarelus*

Een lus die ontstaat indien drie of meer apparaten onderling zodanig zijn verbonden dat hiermee een lus ontstaat.

hardware monitor *hardwaremonitor*

1. Een eenheid die gespecificeerde manifestaties of karakteristieke kenmerken van een computersysteem meet of vastlegt; bijvoorbeeld, een eenheid die de occurrence's van verschillende elektrische gebeurtenissen telt of de tijd tussen die gebeurtenissen meet.
2. Een softwaregereedschap dat manifestaties van de hardware tijdens de uitvoering van een programma vastlegt of analyseert.

hash function *hashfunctie*

Een routine die de adressen van objecten in een hashtabel ontcijfert en lokaliseert.

hash table 1. *hashtabel*; 2. *willekeurig bestand*

Een tabel waarvan de elementen unieke sleutelwaarden bevatten. Voor het opzoeken van de elementen wordt van wiskundige technieken gebruik gemaakt. Met deze technieken kunnen de adressen van de elementen worden bepaald.

hc *hc*

De compiler voor de h notatietaal. Zie h.

header *aanhef*

1. Een blok met commentaar dat aan het begin van een computerprogramma of -routine is geplaatst.
2. Identificatie- of besturingsinformatie die aan het begin van een bestand of boodschap is geplaatst.

hebbian *hebbiaans*

De term verwijst naar de meest algemene methode voor het leren (het doen openen of aanleren van kennis) door een neuronaal netwerk, namelijk die van het leren onder toezicht. Gebruik makend van een voorbeeld dat bekende antwoorden op moet leveren, worden de verbindingsgewichten van het netwerk zodanig aangepast dat ze de verschillen tussen de gewenste en de werkelijke uitvoer voor dit voorbeeld minimaliseren.

help function *hulpfunctie*

Eén of meer schermbeelden waarin wordt beschreven hoe bepaalde functies van de systeemprogrammatuur, de compilers, de macroverwerkingsprogramma's, de vertolkingsprogramma's of de algemene programmeertalen moeten worden gebruikt en wat de betekenis of het gebruik van aangeroepen instructies en commando's zijn. Dergelijke hulpfuncties kunnen zich eveneens uitstrekken tot toepassingsprogramma's.

HEPiX *HEPiX*

Een samenwerkingsverband tussen de instituten die zich met het onderzoek en de toepassing van deeltjes met hoge energie bezig houden. In dit samenwerkingsverband worden onderling compatibele versies van Unix gebruikt.

HEPnet *HEPnet*

Een vereniging die een standaard-netwerk voor HEP-instituten specificeert. Een HEP-instituut is een instituut dat zich met het onderzoek van deeltjes met hoge energie (high energy particles) bezighoudt.

HEPVM *HEPVM*

Een samenwerkingsverband tussen de instituten die zich met het onderzoek en de toepassing van deeltjes met hoge energie bezig houden. In dit samenwerkingsverband worden onderling compatibele versies van IBM's VM-MS bedrijfssysteem gebruikt.

Hermes *Hermes*

Een experimentele object-georiënteerde en gedistribueerde systeemtaal van IBM's Watson Research Centre.

heuristic *heuristisch*

Een methodiek waarin vuistregels worden gebruikt, vereenvoudigingen van statistische methoden worden toegepast, of schattingen worden gemaakt die het zoeken van oplossingen in domeinen die moeilijk zijn of maar matig worden begrepen, reduceren of beperken.

hidden file *verborgen bestand*

Een bestand waarvan de naam met een punt (.) begint. Deze bestanden komen niet in de normale directories voor. Men moet een bepaalde actie uitvoeren om ze zichtbaar te maken.

hierarchical decomposition *hiërarchische decompositie*

Een soort modulaire decompositie waarin een systeem wordt opgebroken in een hiërarchie van componenten. Dit geschiedt door middel van een serie stapsgewijze en van boven-naar-beneden uitgevoerde verfijningen.

hierarchical modelling *hiërarchische modellering*

Een techniek die in de evaluatie van computerprestaties wordt toegepast en waarin een computersysteem als een hiërarchie van deelsystemen wordt voorgesteld. De deelsystemen worden geanalyseerd om de voor die deelsystemen kenmerkende prestaties te kunnen bepalen waarna de resultaten worden gebruikt voor de evaluatie van de prestatie van het totale systeem.

hierarchical object oriented design: HOOD *hiërarchisch object-georiënteerd ontwerp*

Een methode die voor architectuur-ontwerp kan worden gebruikt en hoofdzakelijk bestemd is voor software die in Ada is geschreven. HOOD verzorgt de automatische controle, documentatie en broncodegeneratie van Ada-programma's.

hierarchy *hiërarchie*

1. Een structuur waarin componenten in niveaus van ondergeschiktheid worden gerangschikt; elk component heeft nul, één, of meer ondergeschikten; geen enkel component heeft meer dan één hiërarchisch superieur component.
2. Een classificatie van relaties waarin elk item, met uitzondering van het bovenste of hoogste item (dat als de wortel bekend staat) een specifieke vorm is van het item dat daar in de hiërarchie boven ligt. Elk item kan één of meer items hebben die er in de hiërarchie onder liggen. In de hiërarchie van de Java-klassen is de wortel de objectklasse.

high energy physics database: HEPDB *database voor de hoge energie fysica*

Een databasemanagementsysteem voor dat deel van de natuurkunde dat zich bezig houdt met het onderzoek en de toepassing van deeltjes met hoge energie.

high level language: HLL *hogere programmeertaal*

1. Een programmeertaal die geen weerspiegeling is van de structuur van een bepaalde computer, de opdrachten van die computer of een bepaald type computer.
2. Een probleemgerichte taal die veel op een natuurlijke taal lijkt of waarvan de notaties een sterke verwantschap met wiskundige of administratieve voorstellingen bezitten. Bij het gebruik van deze taal is weinig kennis vereist van de computer waarop een programma, dat in een hogere programmeertaal is geschreven, moet worden verwerkt. De taal kan in verschillende machinetalen worden vertaald, kent symbolische naamgevingen van bewerkingen en adressen, en bezit functies die het definiëren van gegevensstructuren en de programma-logica vergemakkelijken. Gewoonlijk resulteert dit in veel machine-instructies voor elke opdracht die in de programmeertaal wordt gehanteerd. Voorbeelden van hogere programmeertalen zijn: Ada, Algol, Fortran, Basic, Cobol, APL, Pascal, PL/1, C, C++ en Java.

high order language: HOL *hogere-orde taal*

Zie hogere programmeertaal (high level language).

home directory *eigen directory*

De directory of de index die actief is zodra er is ingelogd. De namen van persoonlijke bestanden worden over het algemeen in de eigen directory of subdirectories van de eigen directory bewaard.

homogeneous redundancy *homogene redundantie*

In de fouttolerantie verwijst de term naar de mogelijkheid dezelfde functie met identieke systeemelementen of hulpmiddelen te kunnen realiseren. Bijvoorbeeld, het gebruiken van twee identieke processoren.

Hopfield network *Hopfield-netwerk*

Een neuronaal netwerk waarin geen specifieke invoer- en uitvoerneuronen voorkomen. Het zijn beide zowel invoer- als uitvoerneuronen en in beide richtingen en met gelijke gewichten met alle andere neuronen verbonden. De invoer wordt gelijktijdig aan alle neuronen aangeboden die dan elkaar van uitvoer voorzien. Dit proces duurt net zo lang totdat er een stabiele toestand, dat de uitvoer van het netwerk representeert, is bereikt.

horizontal domain *horizontaal domein*

Een domein dat informatie of diensten aan meer dan één enkel domein levert. Voorbeelden van horizontale domeinen zijn communicaties, grafische gebruikersinterfaces en databases.

horizontal microinstruction *horizontale micro-instructie*

Een micro-instructie die een serie simultane bewerkingen specificeert die nodig zijn om een bepaalde opdracht van de machinetaal uit te kunnen voeren. **Opmerking:** Horizontale micro-instructies zijn relatief lang, vaak langer dan 64 bits en worden 'horizontaal' genoemd omdat de serie simultaan uit te voeren bewerkingen achter elkaar en op één lijn wordt geschreven, in plaats van dat ze onder elkaar op een pagina worden geplaatst.

horizontal reuse *horizontaal hergebruik*

Het hergebruik van assets in meer dan één verticaal domein.

host machine *gastheermachine*

1. Een computer die wordt gebruikt om software op te ontwikkelen die bestemd is voor een andere computer.
2. Een computer die wordt gebruikt om een andere computer op te emuleren.
3. De computer waarin een programma of bestand is geïnstalleerd.
4. Een computer die in een computernetwerk is opgenomen en die door de gebruikers van dat netwerk kan worden benut.

housekeeping operation *huishoudelijke bewerking*

Een computerbewerking die een reeks initiële condities of toestanden ten behoeve van de uitvoering van een computerprogramma instelt of opnieuw instelt. Voorbeelden daarvan zijn het initialiseren van de geheugenruimtes, het vrijmaken van vlaggen, het terugspoelen van banden, het openen en sluiten van bestanden enzovoort.

Huffman coding *Huffman-codering*

Een bijzonder knappe wijze voor het comprimeren van gegevens.

human-machine interface: HMI *mens-machine interface*

Zie gebruikersinterface (user interface).

hybrid coupling *hybride koppeling*

Een soort koppeling van de verschillende deelverzamelingen van het waardebereik dat een gegevensitem aan kan nemen en die voor verschillende en onderling niet-verwante doelen in verschillende softwaremodules kunnen worden gebruikt.

HyperBase *HyperBase*

Een experimentele actieve database voor hypertext-systemen van de Universiteit van Aalborg die in C++ is geschreven. Deze database maakt gebruik van het client-server model en creëert een gedistribueerde, gelijktijdige, en gedeelde toegang tot de werkstations in een lokaal netwerk.

Hypercard *Hypercard*

Een softwarepakket voor de MacIntosh-computer voor het opbergen en ophalen van informatie. De software kan afbeeldingen verwerken en is geschikt voor het zoeken of snuffelen (browsen) naar of in informatie. De krachtige klantgerichte en voor de klant aanpasbare interactieve gebruikersinterface maakt het toevoegen van nieuwe applicaties mogelijk. Deze kunnen gemakkelijk worden geconstrueerd met behulp van het manipuleren van objecten op het scherm waarbij meestal het schrijven van extra programmatuur niet nodig is.

Hyper-Man *Hyper-Man*

Een browser waarmee de hypertext-informatie voor de handleiding van Unix kan worden verkregen.

hypermedia *hypermedia*

Hypertext-systemen waarvan de knooppunten tekst, grafische afbeeldingen, geluid, videobeelden, broncode of andere vormen van gegevens bevatten.

hypermedia/time-based structuring language: HyTime *HyTime*

Een zich ontwikkelende ISO-standaard waarvan de ontwikkeling is geïnitieerd door een speciale groep van geïnteresseerden in hypertext en multimedia binnen de SGML-gebruikersgroep.

Hypertalk *Hypertalk*

De taal die kan worden gebruikt voor het schrijven van procedures die met de objecten van Hypercard worden of zijn geassocieerd.

hypertext *hypertekst*

Een methode die in het werken met informatie wordt gebruikt en waarin tekst in een netwerk met knooppunten, die met behulp van links onderling met elkaar zijn verbonden, is of wordt opgeslagen. De knooppunten kunnen met behulp van een interactieve browser worden geraadpleegd. Een link is een hulpmiddel of mechanisme dat een hoeveelheid tekst met een bestemmingsplaats of een plaats van herkomst van de tekst verbindt; de locaties van de plaats van herkomst en de bestemming worden gewoonlijk op een beeldscherm gemarkeerd door deze opvallende kenmerken te geven, bijvoorbeeld door deze in een andere dan de normale kleur af te beelden.

hyper text markup language: HTML *notatietaal voor hypertekst*

De taal bestaat eigenlijk uit een bestandsformaat dat gebaseerd is op SGML (Standardized Generalized Markup Language). De taal wordt gebruikt voor de compositie en weergave van documenten die in het hypertext-formaat zijn opgesteld en geschikt zijn voor de overdracht via het Internet of het World Wide Web. Het is een eenvoudige taal (of formaat) dat geschikt is voor de invoering van afbeeldingen, geluid, videobeelden en vormvelden en het formatteren van tekst. Er kunnen verwijzingen naar andere objecten of documen-

ten in de tekst worden opgenomen. Het activeren hiervan start het zoeken naar de URL waar dat object of dat document is op te halen.

hypertext transfer protocol: HTTP *overdrachtsprotocol voor hypertext*

1. Het Internet-protocol dat gebaseerd is op het TCP/IP communicatieprotocol en waarmee hypertext-objecten via het net van andere servers en gastheermachines kunnen worden opgehaald.
2. Het protocol dat in het World Wide Web tussen client en server wordt gebruikt.

I

icon *icoon*

Een redelijk kleine, vaak grappige afbeelding van iets dat in samenhang met een grafische gebruikersinterface wordt gebruikt. Een goed ontworpen icoon is meestal een duidelijk, veelzeggend en verklarend symbool van datgene wat het aan functies vertegenwoordigt; het neemt veel minder ruimte in dan de verklarende tekst die met de betekenis van het icoon overeenkomt. In de praktijk zijn iconen echter vaak kleine figuurtjes die geen duidelijke betekenis hebben.

identificer *identificatiesymbool*

1. Een teken dat of groep tekens die wordt gebruikt om een reeks bij elkaar behorende gegevens, zoals een naam, een adres, of een label van een object aan te duiden of van een naam te voorzien.
2. Een gegevensnaam die uniek is of uniek wordt gemaakt met behulp van een syntactisch juiste combinatie van kwalificerende elementen, onder-indexen en indexen.
3. De naam van een item in een Java-programma.

idle *inactief*

De term heeft betrekking op een systeem of component operationeel en in onderhoud, maar niet in gebruik is.

idle time *leeglooptijd*

De tijdsperiode gedurende welke een systeem of component operationeel en in onderhoud, maar niet in gebruik is.

IF-expression *IF-uitdrukking*

Een uitdrukking binnen een procedure of een module die wordt gebruikt om te testen of een conditie waar is of onwaar is. De actie die hiermee wordt verricht, is afhankelijk van het resultaat van deze test. Zie ook IF-opdracht (IF-statement).

IF-statement *IF-opdracht*

1. Een conditionele opdracht die een specificatie geeft van een conditie die moet worden getest en de actie die moet worden genomen indien aan deze conditie wordt voldaan.
2. Een opdracht die wordt gebruikt om een conditionele bewerking uit te voeren. IF wordt altijd door een THEN-clausule gevolgd en eventueel daarna door een ELSE-clausule.

IF-THEN-ELSE operation *IF-THEN-ELSE bewerking*

Een bewerking met één enkele ingang en een tweeweg-sprong met één enkele uitgang die in hogere programmeertalen wordt toegepast. Indien de IF-uitdrukking waar is, wordt de opdracht (of worden de opdrachten) in de THEN-clausule uitgevoerd. Indien deze bewering onwaar is wordt de opdracht (of de opdrachten) in de ELSE-clausule uitgevoerd. In beide gevallen wordt de besturing teruggegeven aan de bewerking die op de complete IF-THEN-ELSE bewerking volgt.

IF-THEN operation *IF-THEN bewerking*

Een programmabewerking waarvan, indien de IF-uitdrukking waar is, de opdracht (of opdrachten) in de THEN-clausule worden uitgevoerd. Indien de IF-uitdrukking onwaar is, worden geen verdere opdrachten die tot deze IF-THEN bewerking behoren, uitgevoerd.

IF-THEN-WHILE operation *IF-THEN-WHILE bewerking*

Een programmabewerking waarvan, indien de IF-uitdrukking waar is, de opdracht (of opdrachten) in de THEN-clausule worden uitgevoerd zolang aan de conditie in de WHILE-clausule wordt voldaan.

immediate data *onmiddellijke gegevens*

Gegevens die het adresveld van een computerinstructie bevatten.

immediate instruction *onmiddellijke instructie*

Een computerinstructie waarvan de adresvelden de waarden van de operanden in plaats van de adressen van de operanden bevatten.

immediate version *directe versie*

Zie zoonversie.

imperative construct *imperatieve constructie*

Een reeks van één of meer opeenvolgende stappen waarin geen sprongen of iteraties voorkomen.

implementation *implementatie*

1. Het proces waarin een ontwerp in haar hardware- of software-componenten, of beide, wordt vertaald.
2. Het resultaat van het proces dat in punt 1 van deze definitie is beschreven.

3. Een definitie die de informatie bevat die nodig is om een object te creëren en dat object in staat stelt een bijdrage te leveren aan het verlenen of verschaffen van daarvoor geëigende diensten (services).

implementation inheritance *implementatie-overerving*

De constructie van een implementatie met behulp van incrementele aanpassingen of wijzigingen van andere (bestaande) implementaties.

implementation phase *implementatiefase*

De tijdsperiode in de levenscyclus van software gedurende welke een softwareproduct, uitgaande van het ontwerp, wordt gerealiseerd (vervaardigd) en gedocumenteerd, alsmede foutvrij wordt gemaakt.

implementation requirement *implementatie-eis*

Een eis die de codering of de constructie (implementatie) van een systeem of systeemcomponent specificiert of beperkt.

implied addressing *impliciete adressering*

Een adresseringsmethode waarin het bewerkingsveld van een computer-instructie impliciet het adres van de operanden bevat. Bijvoorbeeld, indien een computer slechts één accumulator heeft, dan heeft een instructie die naar de accumulator verwijst geen adresinformatie nodig om dat adres te beschrijven. Herhalende en één-voorwaartse adresseringen zijn vormen van impliciete adresseringsmethoden.

implied DO *impliciete DO-opdracht*

Het gebruik van een index-specificatie die overeenkomt met een DO-opdracht zonder het woord 'DO' te specificeren. De specificatie bevat een lijst met gegevenselementen in plaats van een reeks opdrachten.

import (to) *importeren*

Het creëren van een object dat gebaseerd is op een beschrijving van een ander object. Dit laatste object wordt door een externe entiteit aangeleverd.

incipient failure *beginnende storing*

Een storing die op het punt staat plaats te vinden.

incremental backup *incrementele backup*

Een backup-kopie van slechts die bestanden die sinds de laatste volledige backup een wijziging hebben ondergaan.

incremental compiler *incrementele compiler*

Een compiler (compileerprogramma) dat zoveel mogelijk de vertaling van iedere bronopdracht tijdens de invoer of het scannen van de

bronopdracht vertaalt. Een dergelijke compiler wordt specifiek voor de on-line ontwikkeling en controle van programma's gebruikt.

incremental development *incrementele ontwikkeling*

Een software-ontwikkelingstechniek waarin de eis- en behoeftedefinitie, het ontwerp, de implementatie, en het testen in een elkaar overlappende en iteratieve (in plaats van een sequentiële) wijze geschiedt. Dit resulteert in een incrementele opbouw van het totale softwareproduct.

incremental testing *incrementeel testen*

Integratietesten waarbij de systeemcomponenten één voor één aan het systeem worden toegevoegd waarna het tot dusver opgebouwde systeem wordt getest. Dit proces gaat net zo lang door totdat alle componenten daarin correct zijn opgenomen.

independence *onafhankelijkheid*

De term verwijst naar de scheiding van verantwoordelijkheden met behulp waarvan een objectieve evaluatie kan worden bereikt.

independent compilation *onafhankelijke compilatie*

De compilatie van een programma-eenheid of module waarbij geen gebruik van de interface- en contextinformatie van verwante compilatie-eenheden wordt gemaakt. Wanneer onafhankelijk gecompileerde eenheden en modules tenslotte worden gecombineerd, kan het nodig zijn de interface en contextinformatie op correctheid te controleren.

independent verification and validation: IV&V *onafhankelijke verificatie en validatie*

1. Verificatie en validatie die door een organisatie wordt uitgevoerd die technisch, beslissingstechnisch en financieel onafhankelijk is van de organisatie waarin de ontwikkeling plaats vindt.
2. Een separate en objectieve evaluatie die in elke fase van de levenscyclus wordt uitgevoerd.

index *index*

1. Een geordende lijst met verwijzingen naar de inhoud van een bestand of een document, in combinatie met sleutels of verwijzingen om de plaats van die inhoud te (kunnen) bepalen.
2. Een symbool of een getal dat wordt gebruikt om een bepaalde grootheid in een reeks gelijksoortige grootheden te identificeren.
3. Een tabel die wordt gebruikt om de plaats van records of gegevenselementen in respectievelijk een bestand of een reeks te bepalen.

4. Een geheugenpositie of een register waarvan de inhoud de identiteit van een bepaald element in een tabel definieert.
5. Bij het programmeren is dit een onder-index met een gehele getalwaarde die de identiteit van de positie van een gegeven ten opzichte van een ander gegeven vastlegt.

index (to) *indiceren*

1. Het vervaardigen van een lijst met verwijzingen.
2. Het verplaatsen van een (onder)deel van een machine, een element, een object enzovoort naar een vooraf bepaalde positie of over een vooraf gespecificeerde afstand.

indexed *geïndiceerd*

Een adresseringsmethodiek waarbij het werkelijke adres wordt verkregen door een verschuiving aan een basisadres toe te voegen. De waarde van deze verschuiving is meestal in het indexregister opgeslagen; het basisadres bevindt zich in het instructieveld.

indexed address *geïndiceerd adres*

Een adres dat bij de inhoud van een indexregister moet worden opgeteld om het adres van de benodigde geheugenlocatie te verkrijgen.

indexed data name *geïndiceerde gegevensnaam*

Een identificatiesymbool van een gegevensnaam die met één of meer indexnamen wordt geïdentificeerd.

indexed data set *geïndiceerde gegevensverzameling*

Een gegevensverzameling waarbij records worden opgeslagen en opgezocht op basis van sleutels die zich binnen elk record bevinden en die deel uitmaken van het gegevensrecord.

indexed file *geïndiceerd bestand*

Een bestand waarvan de positie van het record in een afzonderlijk gedeelte van dit bestand wordt vastgelegd. Dit wordt de index genoemd en deze index bevat een sleutel en een adres op het schijfengeheugen voor elk record in het desbetreffende bestand.

indexed sequential access method: ISAM *geïndiceerde sequentiële toegangsmethodiek*

Een methode waarbij toegang tot bestanden kan worden verkregen en die wordt gebruikt om zowel de sequentiële als de geïndiceerde toegang te bepalen.

indexed sequential data set *geïndiceerde sequentiële gegevensverzameling*

Een gegevensverzameling waarin elk record een sleutel bevat die de positie bepaalt. De positie van elk record wordt berekend met gebruikmaking van een index waarin de sleutel is verwerkt of opgenomen.

index entry *indexingang*

Een catalogusingang die een beschrijving geeft van het deel van de index dat onder andere wachtwoorden, beschermende kenmerken en statische gegevens bevat.

indexing *indicering*

Een methode om een adres te wijzigen met behulp van indexregisters.

index name *indexnaam*

Een naam die door een programmeur aan een index van een bepaalde tabel wordt toegekend.

index record *indexrecord*

Een verzameling indexingen die als een groep wordt opgezocht en opgeslagen.

indicator *1. indicator; 2. aanwijzer*

Een eenheid of een variabele die in een voorgeschreven toestand of conditie kan worden geplaatst of ingesteld. Dit is gebaseerd op de resultaten van een proces of de occurrence van een gespecificeerde conditie. Voorbeelden zijn vlaggen en semaforen.

indigenous error *endemische fout*

Een fout in een computerprogramma die niet opzettelijk of met een bepaald oogmerk in een programma is opgenomen.

indirect *indirect*

Een adresseringsmethode waarbij het adres van de geheugenpositie die moet worden gelezen of beschreven, zich op een andere plaats in het geheugen of in een register bevindt.

indirect address *indirect adres*

Een adres dat de geheugenlocatie van een ander adres identificeert. De aangewezen geheugenlocatie kan het adres van de gewenste operand of een ander indirect adres bevatten; deze keten van adressen leidt uiteindelijk tot de gewenste operand.

indirect instruction *indirecte opdracht*

Een opdracht die de indirecte adressen van de opdrachtelelementen (operanden) voor de gespecificeerde bewerking bevat.

indirect referencing *indirecte verwijzing*

Een mechanisme in programmeertalen dat voor het verwijzen naar een taalobject wordt gebruikt. Dit geschiedt met behulp van een gegevensobject waarvan de waarde naar het verwezen taalobject wijst. Dit verwijzen kan via een ketting van gegevensobjecten plaatsvinden waarbij elk gegevensobject, behalve het laatste, naar het volgende wijst. Het laatste gegevensobject wijst naar het verwezen taalobject.

inductive assertion method *inductieve beweringsmethode*

Een techniek voor het bewijs van de juistheid (van programma's) waarbij beweringen (asserties) worden opgesteld die de programma-invoer, de uitvoergegevens, en tussentijdse condities beschrijven. Een serie stellingen wordt ontwikkeld waarin het voldoen aan de beweringen aangaande de invoer (de invoerasserties) wordt gerelateerd aan het voldoen aan de beweringen betreffende de uitvoer (de uitvoerasserties), terwijl de stellingen worden bewezen of worden afgekeurd door gebruik te maken van het proces van de volledige inductie.

infant mortality *kindersterfte*

Het aantal storingen dat tijdens de periode van aanvankelijke storingen (early-failure period) van een systeem of component optreedt.

infeasible path *onmogelijke route*

Een route of pad in een programma dat door geen enkele combinatie van mogelijke invoerwaarden kan worden afgelegd of doorlopen.

inference *inferentie*

Het logische proces waarmee nieuwe feiten uit bekende feiten worden afgeleid of gededuceerd.

inference engine *inferentiemachine*

Een programma dat feiten uit kennis of een verzameling ingevoerde waarden afleidt of daar conclusies uit trekt.

informal testing *informele test(en)*

Een test of testen die overeenkomstig opgestelde testplannen en testprocedures worden uitgevoerd, maar die niet zijn beoordeeld en goedgekeurd door een klant, gebruiker of het daarvoor aangewezen management.

information engineering facility: IEF *informatie-ontwikkelings-faciliteit*

Een CASE-gereedschap van de firma Texas Instruments. IEF genereert code vanuit de grafische modellen van bedrijfsprocessen.

information hiding *informatieverberging*

De techniek waarbij software-ontwerpmodules op een zodanige wijze worden opgeborgen (ingekapseld) dat de interfaces van de modules zo weinig mogelijk informatie over de interne werking van het module prijs geven; op deze wijze is een module een black box voor de andere modules van het systeem. De discipline van deze informatieverberging verbiedt het gebruik van informatie van of over een module die niet in de specificatie van de interface van de module is opgenomen.

information resource dictionary system: IRDS *dictionaire-systeem voor informatiebronnen*

Een serie ISO-standaarden voor vergaarbakken (repositories) die in CASE-systemen kunnen worden gebruikt. IRDS regelt de definitie van gegevensdictionaries die op relationele databases kunnen worden geïmplementeerd.

inheritance *overerving*

1. De term verwijst in de object-georiënteerde programmering naar de mogelijkheid om nieuwe klassen van bestaande klassen af te leiden. Een afgeleide klasse erft geïnstantieerde variabelen en methoden van een bepaalde basisklasse en kan daar nieuwe geïnstantieerde variabelen en methoden aan toevoegen. Een nieuwe methode kan dezelfde naam hebben als een methode uit die basisklasse. In dat geval wordt de oorspronkelijke methode overschreven.
2. Het concept waarin klassen automatisch de variabelen en methoden die in de klasse van waaruit ze worden afgeleid, bevat.

inherited error *geërfde fout*

Een fout die vanuit een eerdere stap in een sequentieel proces wordt meegenomen.

initialize (to) *initialiseren*

Een variabele, register, of andere geheugenlocatie van een beginwaarde voorzien.

initial program loader *initieel laadprogramma*

Een zelfstartend laadprogramma (bootstrap loader) dat wordt gebruikt om een deel van een besturingssysteem te laden. Dat deel is nodig om de rest van het besturingssysteem in het geheugen te plaatsen en de werking ervan te initiëren.

inline code *inline-code*

Instructies die op volgorde worden uitgevoerd zonder naar modules, subroutines, of andere programma's te springen.

inline procedures *inline-procedures*

De reeks opdrachten waaruit de hoofdstroom of besturende stroom van het programma bestaat en die opdrachten uitsluit die onder besturing van het asynchrone besturingssysteem worden uitgevoerd.

inlining *inlining*

Het kopiëren van de inhoud van een methode in die van een andere methode. Bijvoorbeeld, indien methode A methode B aanroept, kan een programmeur of compiler de code van methode B in die van methode A kopiëren. Op die wijze is het niet nodig dat A tijdens de werking B aanroept. Dit komt de verwerkingssnelheid ten goede.

input *invoer*

1. Het invoerstadium.
2. Een eenheid, proces, kanaal of gegevens die verband houden met of geassocieerd zijn met een invoerproces.
3. Een variabele die zowel binnen als buiten een component kan zijn opgeslagen en die door het component wordt ingelezen.
4. De term verwijst naar gegevens die vanuit een externe bron zijn ontvangen.
5. Invoergegevens of ingevoerde gegevens.

input and output: I/O *invoer en uitvoer*

Informatie en/of gegevens die een programma, computer of andere verwerkingseenheid worden aangeboden en daaruit als resultaat van bepaalde ver- of bewerkingen worden afgeleverd.

input assertion *invoerassertie*

Een logische uitdrukking die één of meer condities specificceert en waaraan de invoergegevens voor een programma moeten voldoen om te kunnen worden verwerkt.

input domain *invoerdomein*

De verzameling van alle mogelijke invoerwaarden.

input line *invoerregel*

Een reeks tekens die door het systeem als één enkel blok gegevens dat van een terminal afkomstig is, wordt aanvaard. Dit zijn, bijvoorbeeld, alle tekens die worden ingevoerd voordat naar de eerste positie van een regel wordt gesprongen of alle tekens die worden ingevoerd voordat de terminalgebruiker de attentietoets indrukt.

input mode *invoerwerkwijze*

In de invoertoestand worden gegevens ingevoerd en in een buffer of een bestand geplaatst. Dit in tegenstelling tot de verwerkingstoestand (executietoestand) waarin opdrachten, die eventueel van de invoergegevens gebruik kunnen maken, worden uitgevoerd.

input/output control *in- en uitvoerbesturing*

Het geheel of het gedeelte van de verwerkingsomgeving waarin de invoer en de uitvoer van gegevens wordt geregeld.

input/output statement *in- en uitvoeropdracht*

Een opdracht waarmee gegevens tussen het hoofdgeheugen en de in- en uitvoereenheden worden getransporteerd.

input-process-output: IPO *invoer-proces-uitvoer*

Een software-ontwikkelingstechniek die uit de identificatie bestaat van de stappen die in elk proces moeten worden uitgevoerd en waarin de invoer naar en uitvoer van iedere afzonderlijke stap worden geïdentificeerd. Opmerking: Een verfijning, genoemd de hiërarchische invoer-proces-uitvoer identificeert de stappen, de invoer, en de uitvoer van zowel algemene als gedetailleerde detailniveaus.

input-process-output (IPO) chart *IPO-schema*

Een schema van een softwaresysteem of module met een rechthoek die aan de linkerkant van het schema is gepositioneerd en de invoer uitbeeldt, een rechthoek in het midden waarin de processtappen zijn opgenomen, en een rechthoek aan de rechterkant van het schema waarin de uitvoer is geplaatst. Pijlen verbinden de rechthoek van de invoer met de rechthoek van de verwerkingsstappen en deze laatste rechthoek met de rechthoek van de uitvoer.

input program *invoerprogramma*

Een functioneel programma dat het invoerproces van een computer organiseert en voor de invoerbewerkingen zorg draagt.

input routine *invoerroutine*

Het deel van een programmafunctie dat het invoerproces van een computer organiseert.

input value *invoerwaarde*

1. De waarde van een invoergegeven.
2. Een geïnstantieerd invoergegeven.

inspection *inspectie*

1. Een statische analysetechniek waarin gebruik wordt gemaakt van een visuele controle van ontwikkelde producten om fouten, overtredingen van ontwikkelingsstandaarden, en andere problemen te detecteren. Deze vorm van controle omvat onder andere code-inspecties en ontwerpinspecties.
2. Een proces dat bedoeld is om de kwaliteit van geschreven documenten te verbeteren en in groepsverband wordt uitgevoerd. Het proces bestaat uit twee benaderingen; verbetering van het product (het document zelf) en verbetering van het proces

(zowel van de vervaardiging van het product als de kwaliteitsborging ervan).

installability testing *testen van installatiemogelijkheden*

Het testen dat betrekking heeft op de installatieprocedures van een te plaatsen systeem.

installation and checkout phase *installatie- en controlefase*

De tijdsperiode in de levenscyclus van software gedurende welke een softwareproduct in haar operationele omgeving wordt geïntegreerd en in deze omgeving wordt getest. Dat laatste geschiedt om de vereiste prestaties van de software te kunnen garanderen.

installation manual *installatiehandleiding*

Een document dat de informatie bevat die nodig is om een systeem of een component te kunnen installeren, de initiële parameters in te kunnen stellen, en het systeem of het component geschikt te maken voor operationeel gebruik.

instance *1. exemplaar; 2. geïntantieerde kopie*

- a. Een object dat wordt gecreëerd door een klasse te instantiëren, dat wil zeggen actuele waarden, kenmerken of eigenschappen aan die klasse en de objecten van die klasse toe te kennen.
- b. Een object uit een bepaalde klasse. In Java-programma's wordt deze gecreëerd door de nieuwe bewerkingsinstructie gevolgd door de naam van de klasse voor dit doel te gebruiken.

instance (to) *1. instantiëren; 2. een exemplaar maken; 3. iets actueel maken*

Van het Engelse werkwoord 'to instance' heeft de IT-gemeenschap gemeend een nieuw werkwoord te moeten maken: 'to instantiate' hetgeen in het Nederlands is verbasterd tot instantiëren. De betekenis van dit begrip is het toekennen van waarden, kenmerken of eigenschappen aan iets dat minder nauwkeurig is gedefinieerd. Het toekennen van actuele parameters aan formele parameters is een voorbeeld van een dergelijke instantiatie. Ook geldt dit het detailleren van objecten die minder nauwkeurig zijn omschreven of een abstracte betekenis hebben.

instance method *methode van een exemplaar*

Elke methode die met betrekking tot een exemplaar van een klasse wordt geactiveerd. Dit wordt meestal simpelweg aangeduid met methode.

instance variable *instantiatievariabele*

Elk gegevensitem dat met een specifiek object is geassocieerd. Elk exemplaar van een klasse heeft een eigen kopie van de instantiatie-

variabelen die in die klasse zijn gedefinieerd. Dit wordt ook wel een veld (field) genoemd.

instantiation *instantiatie*

1. Het creëren van een object. Zie verder instantiëren (to instance).
2. Een meer nauwkeurig gedefinieerde versie van een bepaald object dat reeds gedeeltelijk was gedefinieerd, creëren.
3. In de object-georiënteerde programmering heeft de term betrekking op een specifiek exemplaar of voorbeeld van een object dat op basis van de mal (template) van de klasse wordt geproduceerd.
4. Het proces waarin specifieke gegevens, instructies, of beide in een generieke programma-eenheid worden gesubstitueerd met het doel deze eenheid in een computerprogramma te gebruiken.

instruction *1. instructie; 2. opdracht*

Een opdracht in een programmeertaal die een specificatie geeft van een uit te voeren bewerking.

instruction address *opdrachtadres*

Het adres dat moet worden gebruikt om een instructie of opdracht op te halen.

instruction address register *opdrachtadresregister*

Een register, uit de inhoud waarvan het adres van de volgende instructie of opdracht wordt afgeleid.

instruction code *opdrachtcode*

Een code waarmee instructies kunnen worden gerepresenteerd.

instruction counter *1. opdrachtteller; 2. opdrachtregister*

Een register dat de locatie van de eerstvolgende computeropdracht die moet worden uitgevoerd, aangeeft.

instruction cycle *opdrachtcyclus*

Het proces waarin een computeropdracht vanuit het geheugen wordt opgehaald en uitgevoerd.

instruction format *opdrachtformaat*

Het aantal en de rangschikking van de velden in een computeropdracht.

instruction length *opdrachtlengte*

Het aantal woorden, bytes, of bits dat nodig is voor de opslag van een computeropdracht.

instruction modifier *opdrachtmodificatie-eenheid*

Een woord of een deel van een woord dat wordt gebruikt om een instructie of opdracht te veranderen.

instruction set *1. opdrachtset; 2. opdrachtreeks*

De reeks opdrachten of instructies van een computer, een programmeertaal, of specifieke programmeringsystemen.

instruction statement *opdrachtdefinitie*

Een opdracht in een programmeertaal die een specificatie geeft van een uit te voeren bewerking en die de identiteit van eventueel aanwezige opdrachtelelementen bepaalt.

instruction time *opdrachttijd*

De tijd die nodig is om een computeropdracht vanuit het geheugen op te halen en uit te voeren.

instruction word *opdrachtwoord*

Een woord dat een instructie of opdracht voorstelt.

instrument (to) *instrumenteren*

In het testen van software en systemen heeft de term betrekking op het installeren of het toevoegen van eenheden of opdrachten aan de hardware of software voor het controleren en volgen van de werking van een systeem of component.

instrumentation *instrumentatie*

Het toevoegen of invoegen van additionele code aan of in een programma om met behulp daarvan informatie betreffende het gedrag van het programma gedurende de uitvoering ervan te verzamelen.

instrumenter *instrumentatiegereedschap*

Softwaregereedschap waarmee een instrumentatie kan worden uitgevoerd.

integer *geheel getal*

Een geheel positief of negatief getal of nul.

integer constant *constante bestaande uit een geheel getal*

Een reeks cijfers waarin geen decimaalteken voorkomt.

integer type *integer type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens slechts uit constanten en gehele getallen kunnen bestaan en slechts kunnen worden bewerkt door rekenkundige bewerkingen op gehele getallen zoals optellingen, aftrekkingen en vermenigvuldigingen.

integrated project support environment: IPSE *omgeving voor geïntegreerde projectondersteuning*

Een term voor een serie technische en managementgereedschappen waarmee de software-ontwikkeling kan worden ondersteund. Deze gereedschappen zijn meestal in een coherent raamwerk ondergebracht.

integrated software *geïntegreerde programmatuur*

1. Toepassingsprogrammatuur zoals spreadsheets, tekstverwerkers en databasebeheerprogramma's die beurtelings voor dezelfde gegevens kunnen worden gebruikt.
2. Een grote verscheidenheid aan programma's en systemen die allen via interfaces aan elkaar zijn gekoppeld en waarbij de indruk wordt gewekt dat men slechts met één compleet programma of systeem werkt. Tussentijdse ingrepen om afzonderlijke programma's te starten of te beëindigen zijn hierbij niet nodig.

integrated systems architecture: ISA *geïntegreerde systeemarchitectuur*

Een Esprit-project dat een vervolg is op het ANSA-project.

integration *integratie*

1. Het proces waarmee componenten worden gecombineerd en in grotere samenstellingen of systemen worden ondergebracht. Deze samenstelling van componenten of systemen dient daarbij aan gestelde regels te voldoen en dient op basis van gespecificeerde voorwaarden en eisen correct te moeten kunnen werken.
2. De eigenschap die demonstreert dat verschillende componenten goed met elkaar samen kunnen werken en wel speciaal in de situatie waarbij verschillende CASE-gereedschappen vanuit een gemeenschappelijke gebruikersinterface worden aangestuurd en waarbij gegevens en toegang tot de functies van die gereedschappen worden gedeeld.

integration testing *integratietesten*

1. Testen waarin de softwarecomponenten, de hardwarecomponenten, of beide worden gecombineerd en getest met het doel de interacties tussen deze componenten te evalueren.
2. Testen die worden uitgevoerd om fouten in de interfaces en de interacties tussen geïntegreerde componenten te detecteren.

integrity *integriteit*

De mate waarin een systeem of component de niet gemachtigde toegang tot, of modificatie van programma's of gegevens kan voorkómen.

interactive *interactief*

De term heeft betrekking op een systeem of werkwijze waarin iedere gebruikersingang of actie van een gebruiker op het systeem een antwoord, reactie, of actie van het systeem veroorzaakt.

interactive data language: IDL *interactieve gegevenstaal*

Een pakket voor de interactieve reductie, analyse, en visualisatie van wetenschappelijke informatie en/of gegevens. Het pakket is afkomstig van de firma Research Systems, Inc.

interactive language *interactieve taal*

Een niet-procedurele taal waarin een programma wordt vervaardigd als resultaat van een interactieve dialoog tussen de gebruiker en het systeem. Het systeem stelt, bijvoorbeeld, vragen en voorziet de gebruiker van formulieren om de gebruiker daarmee te assisteren bij het formuleren van de beoogde resultaten.

interblock gap *bloktussenruimte*

1. Een gebied op of in een gegevensmedium dat het einde van een blok of een fysiek record aangeeft.
2. De ruimte tussen twee opeenvolgende blokken op of in een gegevensmedium.

interface *interface*

1. Een beschrijving van een serie mogelijkheden waarmee een object kan worden gebruikt. Een interface beschrijft specifiek een aantal potentiële vragen die een indicatie zijn van de wijze waarop een object op betekenisvolle wijze met andere objecten of de omgeving kan participeren.
2. Een gemeenschappelijk grensgebied waarover of kanaal waarlangs informatie wordt uitgewisseld.
3. Een hardware- of softwarecomponent dat twee of meer andere componenten met elkaar verbindt met het doel informatie tussen de aldus verbonden componenten uit te wisselen.
4. Een interface is in Java een groep methoden die door verschillende klassen kunnen worden geïmplementeerd. Dit is onafhankelijk van de plaats waar die klassen zich in de hiërarchie van klassen bevinden.

interface (to) *interfacen*

1. Twee of meer componenten met elkaar verbinden met het doel informatie tussen de aldus verbonden componenten uit te wisselen.
2. Als het grensgebied of kanaal, beschreven in punt 2 van de vorige term, te kunnen fungeren of dienst te kunnen doen.

Interface Architect *Interface Architect*

Een programma waarmee een interface voor Motif kan worden geconstrueerd. Het wordt door Hewlett-Packard gedistribueerd.

interface control *interfacebeheer*

1. Het proces in het configuratiebeheer waarin (a) alle functionele en fysieke kenmerken en eigenschappen worden geïdentificeerd die relevant zijn voor het interfacen van twee of meer configuratie-items die door één of meer organisaties worden verschaft, en (b) de verzekering wordt gegeven dat voorgestelde wijzigingen van de eerder genoemde kenmerken en eigenschappen voorafgaande aan de implementatie worden geëvalueerd en goedgekeurd.
2. De administratieve en technische procedures en documentatie in het configuratiebeheer die nodig zijn om functionele en fysieke kenmerken en eigenschappen tussen en in configuratie-items die door verschillende ontwikkelaars worden aangeboden, te identificeren en problemen met betrekking tot de gespecificeerde interfaces op te lossen.

interface definition language of Java: IDLJava *interactieve Java-definitietaal*

Java applicatie programma-interfaces (API's) die een op standaarden gebaseerde connectiviteit en onderlinge operationele verwerkingsmogelijkheden (interoperability) met CORBA verzorgen.

interface requirement *interface-eis*

Een eis waarin een extern item wordt gespecificeerd dat een systeem of systeemcomponent wederzijds beïnvloedt of daarmee moet kunnen communiceren, dan wel een extern item dat de beperkingen definieert van de formaten, de timing, en andere factoren die het gevolg van een dergelijke beïnvloeding, communicatie of interactie zijn.

interface specification *interfacespecificatie*

Een document dat de kenmerken en andere eigenschappen van een interface van een bestaand of gepland systeem of component bevat.

interface test *interfacetest*

1. Test die wordt uitgevoerd om na te gaan in hoeverre systemen of componenten gegevens en besturingsinformatie correct aan elkaar doorgeven.
2. Integratietest waarin de interfaces tussen de componenten van een samenstelling of een systeem worden getest.

interface type *interfacetype*

Een type dat geldt voor elk object (letterlijk, waarvoor elke waarde geldt die een object identificeert) dat aan specifieke interface voldoet.

interleave (to) *afwisselend tussenvoegen*

1. Delen van een reeks zodanig in een andere reeks invoegen dat de reeksen elkaar afwisselen. Daarbij blijft de volgorde van elke reeks gehandhaafd.
2. Het toekennen van opeenvolgende geheugenposities aan fysiek verschillende geheugenmodules.
3. De elementen die een bepaalde volgorde hebben, afwisselen met de elementen die op een andere wijze zijn gerangschikt. Dit dient zodanig te geschieden dat de identiteit van iedere afzonderlijke reeks elementen gehandhaafd blijft; bijvoorbeeld het afwisselend uitvoeren van de stappen van twee verschillende taken met het doel een gelijktijdige uitvoering van die taken te bewerkstelligen.

interleaving *gelijktijdigheid van bewerking*

1. Het gelijktijdig verkrijgen van toegang tot twee of meer bytes of gegevensstromen van verschillende geheugeneenheden.
2. Het afwisselen van twee of meer bewerkingen of functies door overlappend van een computervoorziening gebruik te maken.

Intermedia *Intermedia*

Een hypertext-systeem dat door IRIS (Institute for Research in Information and Scholarship) van de Brown Universiteit (Rhode Island) is ontwikkeld.

Intermedia interchange format: IIF *uitwisselingsformaat voor Intermedia*

Een standaard hypertext-uitwisselingsformaat van IRIS (Zie Intermedia).

intermittent fault *intermitterende fout*

Een tijdelijke of onvoorspelbare fout in een component.

internal name *interne naam*

Een naam die niet buiten het blok, waarin deze naam wordt benoemd, bekend is.

internal text *interne tekst*

De gehele tekst in een bepaald blok.

International Electrotechnical Commission: IEC *Internationale Elektrotechnische Commissie*

Een standaardisatie-lichaam dat kan worden vergeleken met de ISO, maar zich speciaal bezig houdt met normen en standaarden voor elektronische, elektrische en elektrotechnische eenheden, componenten, systemen en daarmee geassocieerde activiteiten en toepassingen.

International Organization for Standardization: ISO *Internationale Standaardisatie Organisatie*

Het internationale instituut dat de officiële standaardisatielichamen of -instituten van de ledenlanden bundelt. De ISO heeft een groot aantal technische commissies en werkgroepen waarin normen en standaarden worden gedefinieerd. De ISO publiceert deze. De ISO werkt nauw samen met haar zusterorganisatie IEC en de officiële Europese standaardisatielichamen (CEN en CEN-CENELEC).

Internet *Internet*

Internet wordt meestal aangeduid als het 'Net'. Internet verbindt een steeds groter wordend aantal computers en computernetwerken. Hiervan maken onder andere Unix-systemen en -netwerken, systemen die gebaseerd zijn op Windows, en een groot aantal andere computersystemen deel van uit. Het meest gebruikte communicatieprotocol voor Internet is TCP/IP.

Internet address *Internetadres*

Een getal van 32 bits dat een Internet-gastheercomputer of -server op unieke wijze definieert. Het getal wordt meestal voorgesteld als vier getallen van 8 bits die door punten onderling van elkaar zijn gescheiden, bijvoorbeeld 135.187.3.2. Het getal bestaat uit een getal dat aan het netwerk is toegekend en een getal dat de gastheer identificeert en kan verder op verschillende manieren worden onderverdeeld.

Internet Engineering Task Force: IETF *IETF*

Een groep experts die technische bijdragen levert aan de ontwikkeling en verdere evolutie van het Internet en de daaraan geassocieerde technologieën. Het is het belangrijkste lichaam dat in de ontwikkeling van nieuwe specificaties van de Internet-standaard een rol speelt.

Internet protocol: IP *Internetprotocol*

Het basisprotocol van Internet. Het verzorgt de onbetrouwbare aflevering van afzonderlijke pakketten van de ene gastheer(computer) naar de andere. Dit protocol verschaft geen enkele garantie dat het pakket zal worden afgeleverd, hoe lang het transport en de aflevering zal duren, of dat meerdere bij elkaar behorend pakketten in de volgorde waarin ze zijn verstuurd ook op dezelfde wijze zullen worden afgeleverd.

interoperability *onderlinge verwerkingswijze*

1. De potentie die twee of meer systemen of componenten bezitten om onderling informatie uit te kunnen wisselen en die informatie te kunnen gebruiken.
2. De mogelijkheid gemeenschappelijke functies of processen uit te voeren over de grenzen die door de verbindingen tussen

heterogene en homogene asset-bibliotheken op gastheercomputers worden gecreëerd.

3. Objecten werken onderling met elkaar samen indien de methoden van het ene object diensten (services) van een ander object vraagt.

interpret (to) *vertolken*

Het vertalen en uitvoeren van elke opdracht in een programmeertaal. Dit geschiedt voordat de volgende opdracht wordt geïnterpreteerd en uitgevoerd.

interpreter *vertolkingsprogramma*

1. Een programma of programmamodule dat iedere instructie of opdracht in een gecodeerd programma afwisselend decodeert en uitvoert; bijvoorbeeld, het Java-vertolkingsprogramma decodeert en voert Java-bytecode uit.
2. Een programma dat elke opdracht in een programmeertaal vertaalt en uitvoert voordat de volgende opdracht wordt vertaald.

interpretive code *interpretatieve code*

Computerinstructies en gegevensdefinities die in een vorm worden uitgedrukt die door een vertolkingsprogramma kan worden herkend en verwerkt.

interrupt *onderbreking*

1. Het verzoek om een proces te onderbreken.
2. De actie waarmee een proces wordt onderbroken.

interrupt (to) *onderbreken*

1. Het afbreken van een proces om een gebeurtenis die geen deel van dat proces uitmaakt, te kunnen verwerken.
2. De afbreking van een proces veroorzaken.

interrupt latency *onderbrekingswachttijd*

De vertragingstijd tussen het verzoek dat een computersysteem ontvangt om een proces te onderbreken en het afhandelen van een dergelijk verzoek.

interrupt mask *onderbrekingsmasker*

Een masker dat wordt gebruikt om onderbrekingen te starten of ongedaan te maken door bits die een verzoek om een proces te onderbreken representeren, vast te houden of te onderdrukken.

interrupt priority *onderbrekingsprioriteit*

Het gewicht dat aan een bepaald verzoek om een proces te onderbreken, wordt toegekend. Dit gewicht bepaalt of het verzoek de onderbreking van het op dat moment in uitvoering zijnde proces zal

veroorzaken en, indien er verschillende verzoeken zijn die op een verdere afhandeling wachten, welk verzoek het eerst zal worden afgewerkt.

interrupt request *onderbrekingsverzoek*

Een signaal of enig andere vorm van invoer waarmee wordt verzocht het op dat moment in uitvoering zijnde proces op te schorten ten behoeve van de uitvoering van een ander proces.

interrupt service routine *onderbrekings-besturingsroutine*

Een routine die op onderbrekingsverzoeken reageert door de inhoud van kritieke registers te bewaren en op te slaan, de verwerking die ten grondslag ligt aan het onderbrekingsverzoek uit te voeren, de inhoud van de registers te herstellen en het onderbroken proces te herstarten.

intervariant *intervariant*

Een bewering (assertie) die altijd 'waar' is voor een gespecificeerd segment of in een gespecificeerd punt van een computerprogramma.

InterViews *InterViews*

Een object-georiënteerde gereedschapskist voor de constructie van grafische gebruikersinterfaces. Het is in C++ geprogrammeerd en bezit een bibliotheek met objecten en een serie protocollen voor het werken met en de constructie van objecten. InterViews is een ontwikkeling van de Stanford Universiteit in Californië.

intrinsic function *intrinsieke functie*

Een functie die door een programma (zoals een Basic- of Fortran-programma) wordt geleverd. Dit in tegenstelling tot een functie die in het programma zelf wordt geschreven, vertaald en uitgevoerd.

Intrinsics *Intrinsics*

Een bibliotheekprogramma dat de basisfuncties van het X Window systeem uitbreidt. Het bezit mechanismen voor het construeren van verzamelingen apparaatjes (widgets) en applicatieomgevingen. Het programma is in feite een uitbreiding van de X bibliotheek (Xlib).

invocation *aanroeping*

De activering van een procedure bij één van de ingangspunten van de procedure.

invoked procedure *geactiveerde procedure*

Een procedure die bij één van de ingangspunten van de procedure is geactiveerd.

isolation testing *isolatietesten*

Het testen van afzonderlijke componenten die van de onmiddellijke omgeving zijn geïsoleerd en waarbij de werking van de direct aangrenzende componenten eventueel wordt gesimuleerd.

iterate (to) *itereren*

Het herhaald uitvoeren van een reeks bewerkingen. De herhalingen worden net zo lang uitgevoerd totdat een bepaald criterium of een gespecificeerde conditie is bereikt.

iteration *iteratie*

1. Het proces waarin een reeks stappen herhaald wordt uitgevoerd.
2. Eén specifieke stap in de reeks stappen die genoemd zijn in punt 1 van deze definitie.

iterative operation *iteratieve bewerking*

1. Het herhalen van een algoritme met het doel een reeks vergelijkingen op te lossen door het opeenvolgend combineren van de aanvangscondities of andere parameters; elke opeenvolgende combinatie wordt gekozen aan de hand van een hulpberekening op een vooraf bepaalde reeks herhalingsregels.
2. Het herhalen van een proces door de uitgangswaarden van het proces al dan niet aangepast, als ingangswaarden van de volgende herhaling van het proces te gebruiken. Deze herhalingen worden beëindigd op het moment dat een bepaalde conditie of een gespecificeerd criterium is bereikt.

ITHACA *ITHACA*

Een Esprit-project dat beoogt een vierde generatie object-georiënteerd systeem bruikbaar te maken voor industriële doeleinden. De ITHACA-omgeving verschaft de gebruiker een applicatieondersteuning waarin geavanceerde technologieën voor de object-georiënteerde programmering, programmeertalen, databasetechnologieën, gebruikersinterfaces en gereedschappen voor de software-ontwikkeling zijn opgenomen.

J**Jackson method** *methode van Jackson*

Een particuliere gestructureerde methode voor de analyse, het ontwerp en de programmering van software. De theorie die aan deze methode ten grondslag ligt, gaat er van uit dat indien de structuur van een programma een weerspiegeling is van de gegevens die door het programma worden verwerkt, het programma gemakkelijk kan worden gelezen en begrepen.

Java *Java*

Een object-georiënteerde programmeertaal die door Sun Microsystems is ontwikkeld. Het is een taal die de mogelijkheid biedt om, nadat men er eenmaal een programma in heeft geschreven, dit programma op ieder platform en iedere computer te kunnen verwerken.

Java application environment: JAE *Java-applicatieomgeving*

De vrijgave van de broncode afkomstig van het Java-ontwikkelingspakket.

Java archive: JAR *Java-archief*

Een bestandsformaat waarin veel bestanden in één enkel bestand zijn samengebundeld (geaggregeerd).

Java archive file *JAR-bestand*

Het geaggregeerde bestand van een Java-archief.

Java archive file format *JAR-bestandsformaat*

De term heeft betrekking op een platform-onafhankelijk bestandsformaat waarin veel bestanden in één enkel bestand zijn samengebundeld (geaggregeerd). Meerdere Java-applets en de daarvoor benodigde en geassocieerde componenten (klassebestanden, beelden, geluid en andere bestanden) kunnen in een JAR-bestand worden opgenomen en daaropvolgend naar een browser in één enkele HTTP-transactie worden overgeheveld ('gedownload'). Deze techniek ondersteunt verder het comprimeren van bestanden en digitale handtekeningen.

Java bean *Java-bean*

Een portabel, platform-onafhankelijk, herbruikbaar en in Java geschreven component dat voldoet aan de specificaties van een rigoureuus gedefinieerd model. De werking van Java-beans kan met behulp van bepaalde gereedschappen worden geanalyseerd.

Java blend *Java-blend*

Een product dat ontwikkelaars in staat stelt de ontwikkeling van database-applicaties te vereenvoudigen door database-records op Java-objecten af te beelden en omgekeerd Java-objecten met databases te laten corresponderen.

JavaCheck™ *JavaCheck*

Een gereedschap voor het controleren of en in hoeverre applicaties en applets aan hun specificaties voldoen.

Java chip *Java-chip*

Een verwerkingseenheid van de firma Sun Microsystems die Java-bytecode direct met het instructiepakket van de verwerkingseenheid uitvoert. Met een Java-chip zal de bytecode de virtuele machine of de JIT-compiler omzeilen en direct in de verwerkingseenheid worden geladen.

Java computing *Java-verwerking*

Een op computers gebaseerde architectuur waarin standaard netwerk-protocollen worden gebruikt. Deze benutten de universele beschikbaarheid van netwerken, gegevens, en Java-applicaties en leveren op een dynamische wijze diensten aan een grote verscheidenheid aan eenheden en systemen die met Java kunnen werken of door Java worden geactiveerd.

Java database connectivity: JDBC *Java-databaseconnectie*

Een industriestandaard voor de database-onafhankelijke verbinding van Java met een groot aantal databases. De JDBC bezit een API (applicatieprogramma-interface) die op oproepbasis (call-level) werkt voor de toegang tot databases die op de standaard-gegevensopvraagtaal voor databases (SQL) zijn gebaseerd.

Java developer connection: JDC *Java-ontwikkelaarconnectie*

Een service die bestemd is voor individuele ontwikkelaars. De service omvat on-line opleiding, productkortingen, artikelen, informatie betreffende fouten, en de mogelijkheden tot het verkrijgen van snelle toegang.

Java development kit: JDK *Java-ontwikkelingsgereedschap*

Een software-ontwikkelingsomgeving voor het schrijven van applets en applicaties in de Java-taal.

Java electronic commerce framework: JECF *Java's elektronisch commercieel raamwerk*

Een gestructureerde architectuur voor de ontwikkeling van elektronische commerciële applicaties in Java.

Java enterprise application programming interface *Java's ondernemingsgerichte applicatieprogramma-interface*

Deze applicatieprogramma-interface (API) zorgt voor een redelijk eenvoudige ontwikkeling van grote commerciële en database-applicaties die multimedia-gegevens met andere toepassingen binnen een organisatie of over het Internet delen.

Java foundation class: JFC *Java's basisklasse*

Een uitbreiding die klasse-bibliotheken van grafische gebruikersinterfaces aan de AWT (Abstract Windowing Toolkit) toevoegt.

Java interface definition language: JIDL *Java's interface-definitietaal*

Java-applicatieprogramma-interfaces (Java API's) die de op standaarden gebaseerde onderlinge werkwijze en verbindingen met CORBA (Common Object Request Broker Architecture) verzorgen.

Java management application program(ming) interface: JMAPI *Java's management applicatieprogramma-interface*

Een serie Java-klassen en -interfaces waarmee ontwikkelaars systemen, netwerken, en managementapplicaties voor services kunnen construeren.

Java media API's *Java's media-applicatieprogramma-interfaces*

Een serie applicatieprogramma-interfaces die de integratie van geluid- en videoknipsels, 2D-lettertypen, grafische beelden en plaatjes, alsmede 3D-modellen en telefoongesprekken ondersteunt.

Java media framework: JMF *Java-mediaraamwerk*

Dit centrale raamwerk ondersteunt klokken voor de synchronisatie van verschillende media zoals audio en video. Dit standaard raamwerk zorgt er voor dat een volledige stroomlijning en synchronisatie van beeld en geluid kan worden gerealiseerd.

Java naming and directory interface *Java-interface voor naamgevingen en directories*

Een serie applicatieprogramma-interfaces die de interfaces naar meervoudige benamingen en directory-diensten verzorgt.

JavaOS *JavaOS*

Een platform dat zodanig is geoptimaliseerd dat Java-programma's er op een verscheidenheid aan computers in kunnen draaien. JavaOS heeft een uitvoeringstijd-omgeving waarin Java-applicaties direct en zonder tussenkomst van een bedrijfssysteem van een gastheercomputer kunnen draaien.

JavaPlan *JavaPlan*

Een object-georiënteerd gereedschap voor de ontwikkeling en het maken van diagrammen en schema's. Het is in Java geschreven.

Java platform *Java-platform*

De virtuele Java-machine (JVM) en de kernklassen (core classes) vormen tezamen het Java-platform. Dit platform zorgt voor een uniforme programma-interface naar een 100% zuiver Java-programma ongeacht het bedrijfssysteem dat wordt gebruikt.

Java remote method invocation: JRMI *Java's op afstand uitgevoerde methode-aanroeping*

Een gedistribueerd objectmodel voor Java-naar-Java toepassingen die is voorzien van een methode waarin op afstand gelegen Java-objecten vanaf andere virtuele Java-machines, die mogelijkterwijs op verschillende gastheercomputers zijn geïmplementeerd, kunnen worden aangeropen.

Java runtime environment: JRE *Java's uitvoeringstijd-omgeving*

Een deelverzameling van het Java-ontwikkelingsgereedschap (JDK) voor eindgebruikers en ontwikkelaars die de JRE willen distribueren. De JRE bestaat uit de virtuele Java-machine (JVM), de kernklassen van Java en ondersteunende bestanden.

JavaSafe *JavaSafe*

Een gereedschap voor het volgen en het beheren van wijzigingen in de bronbestanden. JavaSafe is in Java geschreven.

JavaScript™ *JavaScript*

Een taal waarmee web-scenario's kunnen worden geschreven. De taal wordt zowel in browsers als web-servers gebruikt. JavaScript is slechts zijdelings aan Java verwant en de naam zorgt voor de nodige verwarring. Het wordt hoofdzakelijk gebruikt om componenten te koppelen of de invoer van gebruikers te verwerken. JavaScript is een product van de firma Netscape en kan als een open, platformoverschrijdende object-georiënteerde taal worden beschouwd.

Java Studio™ *Java Studio*

Het eerste Java-programma waarmee op eenvoudige wijze applicaties en applets kunnen worden geconstrueerd zonder de Java-taal te moeten kennen.

Java virtual machine: JVM *virtuele Java-machine*

Dat deel van Java's uitvoeringstijd-omgeving dat de interpretatie van bytecodes verzorgt.

Java wallet *Java-wallet*

Een gebruikersinterface die voor Java's elektronisch commercieel - raamwerk is gebouwd en die on-line winkelen, betalen en administratieve functies verricht.

Java web server *Java-webserver*

Een gemakkelijk te gebruiken, uitbreidbare, eenvoudig te administreren, veilige, en platform-onafhankelijke oplossing voor snel uit te voeren en te verwerken transacties die het gebruik en beheer van Internet- en Intranet-websites vereenvoudigt.

Java Workshop™ *Java Workshop*

Een complete serie gereedschappen die in één omgeving zijn geïntegreerd en met behulp waarvan de programmering in Java kan worden ondersteunt. Java Workshop heeft een sterk modulaire structuur waarmee op eenvoudige wijze nieuwe gereedschappen aan de bestaande kunnen worden toegevoegd.

job 1. job; 2. karwei

- a. Een programma dat vanuit een bedrijfssysteem of schil (shell) wordt gestart en dat met de verwerking aan kan vangen, kan beëindigen, en tussen de voor- en achtergrond kan worden verplaatst.
- b. Een reeks gegevens die een complete omschrijving van een eenheid werk voor een computer inhoudt. Een bewerking omvat gewoonlijk alle noodzakelijke programma's, verbindingen, bestanden en instructies voor het besturingssysteem.
- c. Een door de gebruiker gedefinieerde eenheid van werkzaamheden die door een computer moeten worden uitgevoerd.

job batch *gegevensgewijs invoeren van bewerkingen*

Een reeks bewerkingsomschrijvingen die achter elkaar in een groep worden geplaatst en met een minimale vertraging tussen de bewerkingsstappen wordt ingevoerd.

job control *1. bewerkingsbesturing; 2. karweibesturing*

Een programma dat in het geheugen wordt geladen om elke bewerking of bewerkingsstap gereed te maken voor verwerking. De functies zijn onder andere het toekennen van in- en uitvoereenheden aan symbolische namen, het instellen van switches voor gebruik door het programma, het registreren of afdrukken van bewerkings- en/of besturingsopdrachten enzovoort.

job control language *karweibesturingstaal*

1. Een taal die wordt gebruikt om een reeks karweien te definiëren, hun eisen ten aanzien van een besturingssysteem te beschrijven, en de uitvoering van die karweien te (be)sturen.
2. Een probleemgerichte taal die wordt gebruikt om bewerkingen en bewerkingsstappen (karweien) te coderen.

job definition *bewerkingsomschrijving*

Een reeks bewerkings- en besturingsopdrachten die een bewerking definiëren.

job function *karweifunctie*

Een reeks processen die als één eenheid voor de organisatie, de toekenning of de evaluatie van de werkzaamheden wordt geïdentificeerd. Voorbeelden hiervan zijn ontwerpen, testen, en configuratie-beheer.

job step *bewerkingsstap*

Een door de gebruiker gedefinieerd deel van een karwei dat expliciet met behulp van een besturingsstap wordt geïdentificeerd. Een karwei bestaat uit één of meer bewerkingsstappen.

job stream *1. jobstroom; 2. verwerkingsstroom*

Een serie programma's of karweien die zodanig zijn gearrangeerd dat een computer zonder de interventie van een operateur de karweien of programma's één voor één kan verwerken.

Joint Photographic Experts Group: JPEG *Groep van gezamenlijke fotografie-experts*

De naam van de oorspronkelijke groep die de JPEG-standaard ontwikkelde. Deze standaard wordt gebruikt voor de comprimatie van digitale kleurenbeelden of beelden met grijstonen van natuurlijke echte taferelen of voorstellingen. De standaard is minder geschikt voor niet-realistische beelden zoals cartoons of lijntekeningen. JPEG verwerkt geen zuivere zwart-wit informatie of filmbeelden. Standaarden daarvoor worden door andere commissies ontwikkeld. De standaard zorgt voor een zeer snelle compressie van de beelden. Dit gaat echter ten koste van de nauwkeurigheid van de beelden.

Jovial *Jovial*

Een algemene hogere programmeertaal.

jump *sprong*

Bij de uitvoering van een programma verwijst deze term naar het afwijken van de normale volgorde waarin instructies worden uitgevoerd.

jump (to) *springen*

Van de impliciete of gedefinieerde volgorde afwijken waarin de opdrachten van een computerprogramma worden uitgevoerd.

jump instruction *spronginstructie*

Een instructie die een specificatie is van een sprongopdracht. De volgende instructie die moet worden uitgevoerd, is de instructie op het adres dat door de sprong wordt gespecificeerd.

just-in-time compiler *JIT-compiler*

Een compiler die de gehele bytecode omzet in machinecode van de computer tijdens het verwerken van een Java-programma. Dit resulteert in een grote verbetering van de uitvoeringstijd ten opzichte van de code die door de virtuele Java-machine wordt geïnterpreteerd.

K

Kerberos *Kerberos*

Een beveiligings- en machtigingssysteem van het Athena-project. Dit is overgenomen door de Open Software Foundation als de basis voor de beveiliging van het Distributed Management Environment (DME).

Kermit *Kermit*

Een protocol voor de bestandsoverdracht. Het wordt hoofdzakelijk gebruikt voor de overdracht van bestanden naar en van PC's.

kernel *1. kernel; 2. kern*

- a. Het belangrijkste deel van een bedrijfssysteem zoals Unix. De kernel is onder andere verantwoordelijk voor de toekenning of toewijzing van systeemelementen.
- b. Het belangrijkste deel van een besturingssysteem dat standaardfuncties verricht zoals het toewijzen van systeemcomponenten of -elementen.
- c. Een programma dat in verschillende besturingssystemen of omgevingen kan worden verwerkt.
- d. Een gedeelte van een programma dat zich in het hoofdgeheugen bevindt en andere delen van het programma kan laden.

keyword *sleutelwoord*

1. Eén van de vooraf gedefinieerde woorden van een kunstmatige taal.
2. Eén van de significante woorden in de titel van een document waarmee de inhoud van dit document wordt gespecificeerd.
3. Een symbool dat de identiteit van een parameter bepaalt.
4. Een gedeelte van een opdrachtelemeent dat uit een specifieke tekenrij bestaat.
5. Een gereserveerd woord waarvan het gebruik van essentieel belang is voor de betekenis en de structuur van een opdracht en/of functie.

keyword parameter *sleutelwoordparameter*

Een parameter in de vorm van een sleutelwoord dat door één of meer waarden wordt gevolgd.

kill (to) *beëindigen*

De verwerking van een proces stoppen.

kludge *zootje ongeregeld*

De term verwijst naar een functie of een programma dat wel werkt maar waarvoor de auteur zich schaamt.

knowledge management system: KMS *kennisbeheerssysteem*

Een gedistribueerd hypermedia-systeem voor het beheer van kennis in organisaties. KMS is een commercieel systeem van de firma Knowledge Systems Inc. en draait op werkstations. Het is gebaseerd op onderzoek dat eerder werd uitgevoerd aan de Universiteit van Carnegie Mellon.

Kohonen network *Kohonen-netwerk*

Een neuronaal netwerk ontwikkeld door T. Kohonen van de Universiteit van Helsinki waarvan het algoritme het gedrag van biologische systemen nabootst. In een dergelijk netwerk bestaan twee soorten neuronen, invoerneuronen en andere. Alle invoerneuronen zijn met alle andere neuronen verbonden en de overige neuronen slechts met de meest nabijgelegen burens. Het leeralgoritme van een dergelijk netwerk is relatief eenvoudig en is gebaseerd op de geometrische rangschikking van de neuronen.

L

label *label*

1. Eén of meer tekens binnen of geassocieerd met een reeks gegevens met informatie betreffende deze reeks en die eveneens de identificatie van de reeks bevatten.
2. Een identificatiesymbool van een instructie bij het programmeren.
3. Een identificatierecord voor een bestand op een opslagmedium.
4. Een naamingang.
5. Een naam die wordt gebruikt om de identiteit van een opdracht te bepalen.
6. Een symbolische naam voor een adres.

language *taal*

1. Een verzameling tekens, conventies en regels waarmee informatie kan worden overgedragen. De drie belangrijkste aspecten van een taal zijn het lexicon, de syntaxis en de semantiek.
2. Een systematische wijze voor het uitwisselen van ideeën met behulp van conventionele signalen, geluiden, gebaren, of merktekens en regels voor het vormen (samenstellen, opstellen, genereren) van verwerkbaar en toegestaan uitdrukkingen.
3. Een communicatiemiddel met een syntaxis en semantiek die uit een serie representaties, conventies, en daarmee geassocieerde regels bestaat en die wordt gebruikt om informatie uit te wisselen of over te dragen.

language based editor: LBE *taal-georiënteerd opmaakprogramma*

Een opmaakprogramma (editor) dat rekening houdt met de syntactische, semantische en structurele regels van een bepaalde programmeertaal. Het staat de programmeur toe broncode aan de instructies toe te voegen. Programma's of wijzigingen in eerder opgeslagen programma's worden op een incrementele wijze ontleed in een abstracte syntaxboom en automatisch op juistheid gecontroleerd.

language construct *taalconstructie*

Bij een programmeertaal verwijst deze term naar een syntactisch toegestaan programma of subprogramma dat in overeenstemming met de

reeks regels waaruit de grammatica van de desbetreffende taal is opgebouwd, kan zijn samengesteld.

language processor *taalverwerkingsprogramma*

Een programma dat functies verricht zoals het vertalen, het vertolken of interpreteren en andere taken die nodig zijn om een programmeertaal te verwerken.

language standard *taalstandaard*

Een standaard die de kenmerken en eigenschappen van een taal beschrijft en die wordt gebruikt om daarmee, bijvoorbeeld, de eisen- en behoeftespecificaties, een ontwerp, of testgegevens te definiëren of te specificeren.

language statement *taalopdracht*

Een opdracht waarbij informatie wordt overgebracht naar een verwerkingsprogramma zoals een vertaalprogramma, een algemeen programma of een besturingsprogramma. Een dergelijke opdracht kan een verzoek zijn om een bepaalde bewerking uit te voeren of kan gegevens bevatten die aan het verwerkingsprogramma moeten worden doorgegeven.

language translator *vertaalprogramma*

Een algemene aanduiding van een assembleerprogramma, een compilerprogramma of enig ander programma waarmee opdrachten van een bepaalde taal kunnen worden omgezet in overeenkomstige opdrachten van een andere taal.

latency *wachttijd*

Het tijdsinterval tussen het moment waarop een opdrachtbesturingseenheid een verzoek om gegevens afgeeft en het moment waarop de overdracht van gegevens aanvangt.

lateral compression *laterale compressie*

Een vorm van demodularisatie in het ontwerp van software waarin twee of meer modules die na elkaar worden uitgevoerd, in één module worden gecombineerd.

LaTeX *LaTeX*

Een documentpreparatiesysteem dat op TeX is gebaseerd. Het breidt TeX met een aantal instructies uit die de typografie van het document vergemakkelijken. De gebruiker kan zich concentreren op de structuur van de tekst in plaats van veel aandacht aan de formatteringsinstructies te moeten besteden.

LCSAJ coverage *LCSAJ-dekking*

Het percentage LCSAJ's in een (software)component dat door testen of testseries wordt onderzocht. Zie lineaire code met sprong (linear code sequence and jump).

LCSAJ testing *LCSAJ-testen*

Een test of een testserie die specifiek is ontwikkeld om LCSAJ's te kunnen testen.

leading decision *vooruitlopend besluit*

Een lusbesturing die vóór die van de lus-body wordt uitgevoerd.

legacy *erfenis*

De term heeft betrekking op oude softwaresystemen die nog steeds worden gebruikt maar die zouden kunnen worden verbeterd met behulp van een herziene ontwikkeling (re-engineering) waarbij meer modernere methoden worden toegepast.

Lempel-Zich-Welch: LZW *LZW*

Een gegevenscompressietechniek die door Lempel, Zich en Welch is ontwikkeld.

lex *lex*

Een gereedschap voor de lexicale analyse in Unix-omgevingen.

lexical *lexicaal*

De term verwijst naar de wijze waarop de tekens in broncode worden vertaald in symbolische begrippen of beschrijvingswijzen die door een compiler kunnen worden begrepen.

library *bibliotheek*

1. Een verzameling bestanden.
2. Een opslagplaats voor verwisselbare geheugenmedia.
3. Een gesegmenteerde gegevensverzameling.
4. Een verzameling gegevensstructuren die afzonderlijk kunnen worden opgezocht, geraadpleegd of opgehaald.
5. Een reeks programma's en/of procedures die op systematische wijze in een geheugenmedium zijn opgeslagen en van daaruit kunnen worden opgehaald.

library directory *bibliotheekindex*

Het gedeelte van de bibliotheek dat informatie over de bibliotheek bevat, zoals de naam en de locatie van elk onderdeel of element van de bibliotheek.

library list *bibliotheeklijst*

Een geordende lijst met bibliotheeknamen die wordt gebruikt om een object te kunnen vinden. Deze lijst geeft aan welke bibliotheken

moeten worden geraadpleegd en de volgorde waarin dat moet geschieden.

library metric *bibliotheekmetriek*

Een standaard-waardebepaling of -maatanduiding waarmee kwantitatieve vergelijkingen en evaluaties die betrekking hebben op het werken met bibliotheken voor hergebruik, kunnen worden uitgevoerd.

library object *bibliotheekobject*

Een benoemd systeemelement dat in het bibliotheeksysteem van een (gastheer)computer is opgeslagen.

library of efficient data types and algorithms: LEDA *bibliotheek van efficiënte gegevenstypen en algoritmen*

Een bibliotheek van C++-klassen die eveneens is voorzien van klassen met grafische informatie. De bibliotheek is ontwikkeld aan de Universiteit van Saarbrücken.

library program *bibliotheekprogramma*

Een programma in een programmabibliotheek.

library routine *bibliotheekroutine*

Een module die in een programmabibliotheek is opgeslagen en daarin wordt bijgehouden.

licensing standard *licentiestandaard*

Een standaard met de kenmerken van een machtiging, die door een officieel of wettelijk erkend lichaam aan een persoon of organisatie wordt afgegeven en waarin wordt beschreven dat die persoon of organisatie een specifieke zaak kan ondernemen of uit kan voeren, of een specifiek ding kan bezitten.

life-cycle *levenscyclus*

De levenscyclus van software bestaat uit de volgende fasen: definitie van de eisen en/of behoeften, analyse, ontwerp, codering, testen, en onderhoud. Het ontwikkelingsproces heeft de neiging op iteratieve wijze in plaats van lineair het merendeel van deze fasen te doorlopen. Dit proces kan met behulp van verschillende methoden (watervalmethode, spiraalvormig model) worden beschreven. Zie verder software-levenscyclus.

life-time *levensduur*

Bij programmeertalen heeft deze term betrekking op het gedeelte van de uitvoeringstijd gedurende welke een taalobject bestaat.

Linda *Linda*

Een portabele parallele taal die de parallele programmering vereenvoudigt. De taal bevat uitbreidingen op C en Fortran en is een product van Scientific Computing Associates Inc.

linear code sequence and jump: *LCSAJ* *lineaire code met sprong*

Een instructie die uit de drie volgende items bestaat: het begin van de lineaire serie van uitvoerbare instructies, het einde van deze serie, en het regelnummer van de instructie waar het programma de verdere afwerking vervolgt. De drie items worden door regelnummers in de lijst van de broncode geïdentificeerd.

linear list *lineaire lijst*

Een lineair geordende reeks gegevenselementen die dezelfde structuur hebben en waarvan de volgorde in het geheugen wordt beschermd door van een sequentiële toewijzing gebruik te maken.

line editor *regelopmaakprogramma*

Een programma voor de opmaak van tekst dat de tekst lijn voor lijn toont waarna de tekst in die lijn kan worden gewijzigd. De meeste opmaakprogramma's zijn schermopmaakprogramma's die een scherm met de tekst vullen (20 of meer lijnen). Die tekst kan dan op willekeurige plaatsen op het scherm worden gewijzigd. Regelopmaakprogramma's zijn nauwelijks nog in gebruik.

line number *regelnummer*

1. Een nummer dat betrekking heeft op een regel in een afgedrukte of afgebeelde tekst.
2. Een nummer dat betrekking heeft op een regel in een verzameling logische records.

link *link*

1. Een relatie tussen twee objecten.
2. Een additionele naam die aan een bestand is toegekend. Indien een bestand wordt gecreëerd, wordt de inhoud van dat bestand ergens opgeslagen en van een naam voorzien. Die naam wordt in een index geplaatst. Men kan additionele bestandsnamen creëren die dan naar dezelfde inhoud verwijzen.
3. Een deel van een computerprogramma (dit is meestal één enkele opdracht of adres) dat de besturing en parameters tussen afzonderlijke modules van een programma overdraagt of doorgeeft.

link (to) *1. linken; 2. monteren*

- a. Een laadmodule uit twee of meer afzonderlijk vertaalde object-modules creëren of modules laden door de kruisverwijzingen tussen die modules aan te passen en de verschillen daartussen te verwijderen.

- b. Een link, zoals beschreven in punt 3 van de vorige definitie, tot stand brengen.

linkage *1. verbinding; 2. koppeling*

Bij het programmeren is dit de codering die twee onafhankelijk van elkaar gecodeerde modules met elkaar doet verbinden dan wel aan elkaar koppelt.

linkage editor *montageprogramma*

Een programma dat wordt gebruikt om één laadmodule van één of meer onafhankelijk vertaalde programmamodules of laadmodules tot stand te brengen door middel van het herleiden van de verwijzingen en naamgevingen tussen deze modules.

linkage instruction *verbindingsinstructie*

Een instructie die de besturing en de parameters tussen de afzonderlijke delen van een programma doorgeeft.

linked list *verbonden lijst*

Een lijst waarin gegevens in een willekeurige volgorde voor kunnen komen, maar waarin elk gegeven een verwijzing bevat om met behulp daarvan de positie van het volgende gegeven te kunnen bepalen.

linker *linker*

Een programmamodule die een volledig uitvoerbaar programma opbouwt met behulp van componenten die uit machinecode-modules bestaan. De Java-linker bouwt op die wijze een uitvoerbaar programma op door van de gecompileerde klassen gebruik te maken.

linking loader *koppel-laadprogramma*

Een computerprogramma dat één of meer objectmodules in het geheugen laadt ter voorbereiding van de uitvoering van die modules, één enkel laadmodule creëert door de kruisverwijzingen tussen afzonderlijke modules aan te passen en, in sommige gevallen, de adressen aan te passen om de correcte geheugenplaatsen waarin de code is geplaatst, te verkrijgen.

lint *lint*

Een voorverwerkingsprogramma voor de C-taal dat meer controles op de gemaakte code uitvoert dan de C-compiler zelf doet.

Linux *Linux*

Een compleet vanuit scratch geïmplementeerde versie van Unix. De code is vrij verkrijgbaar onder een voor iedereen toegankelijke Gnu-licentie. Linux is een product van Linus Torvalds.

Lisp *Lisp*

Een lijst-verwerkende taal die uitermate geschikt is voor de symbolische en logische programmering. Lisp wordt als de eerste vierde-generatie taal beschouwd en wordt veel toegepast in toepassingen waarin kunstmatige intelligentie is verwerkt.

list *lijst*

1. Een geordende reeks gegevens.
2. Een verzameling gegevensitems waarvan elk element dezelfde definitie heeft.

list (to) *1. afdrukken; 2. weergeven*

Een verzameling gegevensitems of een geordende reeks gegevens afdrukken of op een beeldscherm projecteren.

listing *1. lijst; 2. afdruk; 3. uitdraai*

Een geordende weergave of afdruk van gegevensitems, programma-instructies, of andere informatie.

list processing *lijstverwerking*

Een methode om gegevens in de vorm van lijsten te verwerken. Gewoonlijk worden daarbij kettinglijsten gebruikt zodat de logische volgorde van de gegevens kan worden gewijzigd zonder veranderingen in de fysieke posities aan te brengen.

list processing language *lijstverwerkingstaal*

Een programmeertaal die is ontwikkeld om het manipuleren van gegevens die in de vorm van lijsten zijn weergegeven, te vergemakkelijken.

literal *1. literaal; 2. letterwaarde*

- a. Een waarde die een entiteit, die geen object is, voorstelt.
- b. De representatie van elk geheel getal, drijvende-kommagetal, of teken.
- c. Een expliciete weergave van de waarde van een gegeven. Deze waarde mag niet worden gewijzigd gedurende het vertalen van een programma waarin een letterwaarde voorkomt.
- d. Een tekenrij waarvan de waarde impliciet gegeven is in de tekens die de rij constitueren.
- e. Een expliciete representatie van de waarde van een item zoals het woord 'Fout' in de instructie: IF x = 0 then print "FOUT".

literate programming *gedocumenteerde programmering*

De combinatie van het gebruik van een tekstverwerkingssysteem zoals TeX en een conventionele programmeertaal. Het doel hiervan is de documentatie en de broncode tegelijkertijd en in samenhang met elkaar te ontwikkelen.

load (to) *laden*

1. Het invoeren van gegevens of een programma in een geheugen of in registers ter voorbereiding van de uitvoering van het programma of de verwerking van de gegevens en, in sommige gevallen, het aanpassen van de adressen en het monteren (linken) van modules.
2. Het overbrengen van een te laden module van een hulpgeheugen naar een hoofdgeheugen (secundair geheugen naar primair geheugen) om te worden uitgevoerd.
3. Het kopiëren van computeropdrachten of gegevens vanuit externe geheugens naar het interne geheugen of van het interne geheugen naar de registers van de computer.

load-and-go *laden en uitvoeren*

Een verwerkingsmethode waarbij geen oponthoud plaats vindt tussen de laad- en uitvoeringsfase van een programma. De laadfase kan ook het assembleren of compileren van het programma omvatten.

loaded origin *geladen oorsprong*

Het adres van de initiële geheugenplaats van een computerprogramma op het moment dat het programma in het hoofdgeheugen is geladen.

loader *laadprogramma*

1. Een computerprogramma dat machinecode in het hoofdgeheugen plaatst (leest of laadt) voor de uitvoering van het programma en, in sommige gevallen, de aanpassing van de adressen en het monteren (linken) van de modules. Er bestaan een aantal verschillende soorten laadprogramma's: absoluut laadprogramma, koppel-laadprogramma, en laadprogramma met adresberekening.
2. Elk programma dat programma's of gegevens in het hoofdgeheugen plaatst.

load map *laadoverzicht*

Een door de computer gegenereerd overzicht dat of lijst die de locatie of de grootte van alle of geselecteerde delen van code of gegevens die zich in het geheugen bevinden, identificeert.

load module *laadmodule*

Een programma of deelprogramma dat geschikt is om in het hoofdgeheugen te worden geladen en daarna te laten (ver)werken; dit is meestal de uitvoer van een montageprogramma (linkage-editor).

local compaction *lokale compactheid*

Een compactheid in de microprogrammering waarin microbewerkingen niet over de grenzen van de sequentiële blokken met bewerkingen met een enkelvoudige ingang en een enkelvoudige uitgang worden verplaatst.

local data *lokale gegevens*

Gegevens waar slechts één module of een reeks geneste modules in een computerprogramma toegang toe hebben.

local mount *lokale combinatie*

De logische verbinding van verschillende schijfgeheugens aan één en dezelfde machine. Dit wekt de indruk dat er van één groot bestands-systeem sprake is.

local variable *lokale variabele*

Een gegevensitem dat binnen een logisch blok bekend is maar waar de code van buiten dat blok geen toegang toe heeft. Zo is, bijvoorbeeld, elke variabele die in een Java-methode is gedefinieerd een lokale variabele en kan niet buiten die methode worden gebruikt of aangeroepen.

locked name *vergrendelde naam*

Een naam die op een bepaald moment niet beschikbaar hoeft te zijn voor alle taken die deze naam kennen.

lockout *blokkering*

Een techniek voor de toewijzing van computerelementen waarin gedeelde elementen (dat geldt speciaal voor gegevens) zijn beschermd door op een gegeven moment slechts aan één eenheid of proces toegang tot die elementen te verlenen.

Locus *Locus*

Een project voor de ontwikkeling en/of toepassing van een gedistribueerd systeem dat de transparante toegang tot gegevens in een bestandssysteem binnen een netwerk verzorgt.

logical address *logisch adres*

Een geheugenadres dat aan of door een programma wordt verstrekt tijdens het ophalen en uitvoeren van een instructie of als een wijzer (pointer) tijdens een in- en uitvoerbewerking wordt gebruikt.

logical cohesion *logische cohesie*

Een soort cohesie waarin de taken die door een softwaremodule worden uitgevoerd, functies verrichten die logisch gezien gelijksoortig zijn; bijvoorbeeld, de verwerking van verschillende soorten invoergegevens.

logical constant *logische constante*

Een constante waarmee wordt gespecificeerd of iets waar of onwaar is.

logical expression *logische uitdrukking*

1. Een uitdrukking die logische bewerkingstekens en opdrachtele-
menten bevat en die kan worden herleid tot een waarde die waar
of onwaar is.
2. Een conditionele uitdrukking die uit een combinatie van
logische termen, logische bewerkingstekens en (dubbele)
haakjes bestaat.

logical file *logisch bestand*

Een verzameling records die met elkaar zijn verbonden en die als
een eenheid wordt behandeld. Zo kan een regel van een factuur een
gegevenselement vormen, een volledige factuur een record, en een
volledige verzameling van dergelijke records, een bestand.

logical operation *logische bewerking*

1. Een bewerking die volgens de regels van de symbolische logica
wordt uitgevoerd.
2. Een bewerking waarbij elk teken van het resultaat afhankelijk
is van niet meer dan één teken van elk opdrachtele-
ment.

logical operator *logisch bewerkingsteken*

1. Een bewerkingsteken of een aantal bewerkingstekens dat in een
logische uitdrukking kan worden gebruikt om de bewerking aan
te geven die op de termen van de uitdrukking moeten worden
uitgevoerd. De logische bewerkingstekens zijn symbolen of
woorden die de logische bewerkingen En, Of, Niet, Noch
enzovoort voorstellen.
2. Een woord dat de logische verbanden tussen relationele
bewerkingstekens omschrijft. De logische bewerkingstekens zijn
symbolen of woorden (termen) die synoniem zijn met logische
bewerkingen zoals: En, Of, Niet, Noch, Niet-En, Niet-Of
enzovoort.

logical relation *logische relatie*

Een logische term waarin twee uitdrukkingen van elkaar worden
gescheiden door middel van een vergelijkingsteken. Deze vergelij-
kingstekens zijn: =, <, >, ≤, ≥, en ¬.

logical trace *logische trace*

Een naspeuring of tracering van een in uitvoering zijnd programma
waarin slechts de spronginstructies worden vastgelegd.

logical type *logisch type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende
gegevens slechts logische waarden (meestal 'waar' of 'onwaar') aan
kunnen nemen en die slechts met behulp van logische bewerkings-
tekens, zoals 'En', 'Of' en 'Niet' kunnen worden verwerkt.

logic control statement *lus-besturingsopdracht*

Een opdracht die wordt gebruikt voor het specificeren van opdrachten die onder besturing van een programmalus moeten worden uitgevoerd. De parameters die hierbij worden gebruikt en de omstandigheden voor het beëindigen van de lus maken eveneens deel van deze opdracht uit.

logic control variable *lusbesturingsvariabele*

Een variabele die invloed uitoefent op het uitvoeren van instructies in de lus en die door de lusbesturing worden gemodificeerd.

logic design *logisch ontwerp*

De specificatie van de wijze waarop de samenstellende elementen van een systeem ten opzichte van en met elkaar werken. Dit wordt uitgedrukt in symbolische logica zonder acht te slaan op de fysieke uitvoering van de apparatuur.

logic diagram *logisch schema*

Een grafische uitbeelding van een logisch ontwerp.

logic element *logisch element*

Een eenheid die een elementaire logische functie verricht.

logic instruction *logische instructie*

Een instructie waarvan het gedeelte dat een bewerkingsteken bevat, een logische bewerking specificceert.

logic operator *logische bewerking*

1. Een bewerking die volgens de regels van de symbolische logica wordt uitgevoerd.
2. Een bewerking waarbij elk teken van het resultaat slechts afhankelijk is van het overeenkomstige teken van elk opdracht-element.

logic programming *logische programmering*

De programmering in een taal zoals Prolog of Lisp waarin de programmeur een serie bewerkingen of verklaringen opstelt die door een inferentiemachine worden geïnterpreteerd.

logic programming language *logische programmeertaal*

Een programmeertaal met programma-opdrachten en -uitdrukkingen die de vorm van besturingsconstructies en een beperkte predikatencalculus hebben.

logic symbol *logisch symbool*

Een symbool dat een bewerkingsteken, een functie, of een functionele relatie voorstelt.

log in (to) *inloggen*

Toegang tot een systeem (hardware/software) verkrijgen. Een dergelijke toegang vereist dat de gebruiker die om die toegang verzoekt, één of meer identificaties aan het systeem verstrekt: paswoord, login-identificatie, gebruiker-identificatie, naam van de gebruiker enzovoort.

Logiscope *Logiscope*

Een gereedschap voor de analyse van de kwaliteit van software met behulp waarvan deze zowel statisch als dynamisch kan worden geëvalueerd. Logiscope is een product van de firma Verilog SA.

loop *lus*

1. Een reeks instructies die steeds opnieuw wordt uitgevoerd totdat aan een bepaalde conditie of een gespecificeerd criterium is voldaan of de toestand wordt bereikt waarin een gegeven voorwaarde waar is.
2. Een hardwarelus (kring) of softwarelus (zie punt 1 van deze definitie).
3. De elektrische verbindingsweg tussen een station en een kanaal.

loop (to) *een lusbewerking uitvoeren*

De reeks instructies zoals genoemd onder punt 1 van de vorige definitie uitvoeren.

loop assertion *lusassertie*

Een logische uitdrukking waarin één of meer voorwaarden zijn gespecificeerd waaraan op elk moment dat een specifiek punt in een programmalus wordt uitgevoerd, moet worden voldaan.

loopback test *rondkoppeltest*

Testactiviteit waarin signalen of gegevens van een testeenheid de invoer voor een systeem of component vormen en de resultaten terug worden gegeven aan de testeenheid voor het uitvoeren van metingen of het maken van vergelijkingen.

loop body *lus-body*

Het deel van een lus waarin het belangrijkste doel van de lus wordt gerealiseerd.

loop construct *lusconstructie*

Een constructie in programmeertalen die een herhaling van de uitvoeringsvolgorde specificeert, bijvoorbeeld een DO-lus, een DO WHILE-lus enzovoort.

loop control *lusbesturing*

Het deel van een lus waarin wordt bepaald of de lus al dan niet moet worden beëindigd en verlaten.

loop-control variable *lusbesturingsvariabele*

Een programmavariabele die wordt gebruikt om te bepalen of een lus al dan niet moet worden beëindigd en verlaten.

loop invariant *lusinvariant*

Zie lusassertie (loop assertion).

low level language *lagere programmeertaal*

Een programmeertaal waarbij iedere instructie in de symbolische taal overeenkomt met één machine-instructie.

Lukasiewicz notation *prefixnotatie*

Een methode om wiskundige uitdrukkingen te vormen waarbij elk bewerkingsteken voorafgaat aan de bijbehorende opdrachtelelementen en de bewerkingen aangeeft die op deze opdrachtelelementen of de daaropvolgende tussenresultaten moeten worden verricht.

LynxOS *LynxOS*

Een real-time besturingssysteem dat zich conformeert aan de door POSIX (Portable Operating System based on Unix) opgestelde eisen. Dit bedrijfssysteem heeft een Unix-achtige interface voor applicatieprogramma's.

M

mach *mach*

De kern van een besturingssysteem dat door de Universiteit van Carnegie-Mellon is ontwikkeld. Dit verzorgt gedistribueerde en gelijktijdig uit te voeren bewerkingen en berekeningen. Mach is zodanig ontworpen dat het in staat is de computeromgevingen die uit netwerken van enkel- en meervoudige verwerkingseenheden bestaan, kan ondersteunen. Mach is de kernel van het OSF/1 systeem.

machine code *machinencode*

Computerinstructies en gegevensdefinities die in een zodanige vorm zijn opgesteld dat ze door de verwerkingseenheid van een computer kunnen worden herkend.

machine dependent *machine-afhankelijk*

De term heeft betrekking op software die afhankelijk is van functies en eigenschappen van een specifiek type computer. Deze software kan derhalve slechts op een computer van dat type worden verwerkt.

machine independent *machine-onafhankelijk*

De term heeft betrekking op software die niet afhankelijk is van de functies en eigenschappen van een bepaald type computer. Deze software kan derhalve op meer computers dan die van dat ene type worden verwerkt.

machine language *machinetaal*

1. Een reeks binaire codes die instructies voorstellen en die direct door een computer kunnen worden herkend en uitgevoerd. De taal kent dus geen symbolische namen van bewerkingen en adressen.
2. Bij het verwerken van tekst is de machinetaal de taal die wordt gebruikt om tekst en programma-instructies in het geheugen te plaatsen en die daarna voor het verwerken en het afdrukken worden gebruikt.

machine-oriented language *machinegerichte taal*

1. Een programmeertaal die een weerspiegeling is van de structuur van een bepaalde computer.
2. Een programmeertaal waarvan de woorden en de syntaxis voor een bepaald soort computer zijn geconcipieerd.

machine readable *machine-leesbaar*

De term verwijst naar gegevens in een vorm die als automatische invoer voor een computer kan dienen; bijvoorbeeld, gegevens die op een diskette zijn gecodeerd.

macro *macro*

Een groep instructies die met een symbolische naam wordt aangeduid. Wanneer deze naam in een programma voorkomt, vervangt het assembleerprogramma deze naam door de groep instructies. Het is ook mogelijk dat een dergelijke naam in hogere programmeertalen of toepassingsgerichte talen voorkomt. In dat geval wordt de naam eveneens door de in de macro opgenomen instructies vervangen.

macro assembler *macro-assembleerprogramma*

Een assembleerprogramma waarin de functies van een macrogenerator zijn opgenomen of waarin dergelijke functies kunnen worden uitgevoerd.

macro call *macro-oproep*

1. Een instructie in een bepaalde taal die wordt vervangen door een gespecificeerde reeks instructies in dezelfde taal. De macro-oproep kan eveneens waarden specificeren voor parameters in de instructies die in de macrodefinitie voorkomen en daarin moeten worden vervangen.
2. Een opdracht die tot gevolg heeft dat een vooraf bepaalde reeks opdrachten, de macrodefinitie genaamd, wordt verwerkt.

macro declaration *macrodefinitie*

1. Een verklaring die de code bevat die een macrogenerator gebruikt om een macro-instructie of macro-oproep te vervangen.
2. Een reeks opdrachten die de naam en de opmaak van en de omstandigheden voor het genereren van een reeks opdrachten in een taal omschrijft.

macro expansion *macro-expansie*

De reeks opdrachten die ontstaat uit een bewerking waarin macro's worden toegepast.

macro generator *macrogenerator*

Een programma dat of routine die meestal een deel van een compiler of een assembleerprogramma is en die macro-instructies in een brontaal tijdens de assemblage of compilatie vervangt door een bepaalde reeks instructies in diezelfde taal.

macro instruction *macro-instructie*

Een instructie die, wanneer deze wordt geactiveerd, de uitvoering van een vooraf omschreven reeks instructies in dezelfde brontaal veroorzaakt. Een macro-instructie specificceert ook de actuele parameters voor alle formele parameters in de vervangende reeks instructies. Zie ook macro-oproep (macro call).

macro language *macrotaal*

De regels voor het schrijven van macro-instructies en macrodefinities en de symbolen die hiervoor worden gebruikt.

macro library *macrobibliotheek*

Een bibliotheek van macrodefinities die door een macrogenerator en tijdens het ontwikkelen van macro's kunnen worden gebruikt.

macro processor *macroprocessor*

Eén of meer routines die in bepaalde assembleerprogramma's of compilers voorkomen en die kunnen worden benut voor de definitie en het gebruik van macro's.

macro programming *macroprogrammering*

De programmering waarin macro-instructies worden gebruikt.

macro statement *macro-opdracht*

Zie macro-instructie (macro instruction).

macro substitution *macrosubstitutie*

Tijdens het opmaken (het editen) is dit de substitutie van besturingswoorden, symbolen en tekst door een macro.

main program *hoofdprogramma*

1. Het programma van het hoogste niveau dat bij een programmastap is betrokken.
2. Een programma-eenheid die geen functie- of subroutine-opdracht bevat en tenminste één uit te voeren opdracht heeft.
3. Een softwarecomponent dat door het besturingssysteem van een computer wordt opgeroepen en meestal andere softwarecomponenten oproept of de werking daarvan initieert.

maintainability *onderhoudbaarheid*

1. Het gemak waarmee een softwaresysteem of component kan worden gemodificeerd voor het herstellen van fouten, het verbeteren van de prestatie of andere attributen, of het aanpassen van het systeem of component aan een gewijzigde omgeving.
2. Het gemak waarmee een hardwaresysteem of component in een toestand kan worden gehouden of hersteld waarbij het de vereiste functies uit kan voeren.

maintainability testing *testwerkzaamheden ten behoeve van het onderhoud*

Testwerkzaamheden waarin wordt nagegaan indien het systeem de gespecificeerde doelstellingen met betrekking tot het onderhoud verzorgt of kan verzorgen.

maintenance *onderhoud*

1. Het proces waarin een softwaresysteem of component na aflevering wordt gemodificeerd om fouten te herstellen, de prestatie of andere attributen te verbeteren, of aan een gewijzigde omgeving aan te passen.
2. Het proces waarin een hardwaresysteem of component in een toestand kan worden gehouden of hersteld waarin het de vereiste functies uit kan voeren.
3. Alle werkzaamheden die zijn bedoeld om fouten te elimineren of apparatuur en/of programmatuur in een goede conditie te houden. Onderhoud is een belangrijk stadium in de levenscyclus van een softwaresysteem of -product. Met behulp van bepaalde software-ontwikkelingsmethoden is men in staat de grote onderhoudskosten tot een acceptabel niveau te reduceren.

Make *Make*

Een populair gereedschap in Unix-systemen met behulp waarvan het hercompileren en het onderling koppelen van modules kan worden geautomatiseerd. Daarbij wordt rekening gehouden met de onderlinge relaties en afhankelijkheden van de modules van het totale systeem of programma.

Makedoc *Makedoc*

Een programma van de Canadese Carleton Universiteit. Met dit programma kan de documentatie van object-georiënteerde C-programma's worden verzorgd. Het programma genereert eveneens een schema dat de klassehiërarchie bevat.

man-machine interface: MMI *mens-machine interface*

Zie gebruikersinterface.

manufacture (to) *fabriceren*

Het proces in de ontwikkeling van software waarin software naar diskettes, CD's, chips, of andere media wordt gekopieerd met het doel deze onder klanten of gebruikers te distribueren.

manufacturing phase *fabricagefase*

De tijdsperiode in de levenscyclus van software waarin de basisversie van een softwareproduct aan een gespecificeerd aantal operationele omgevingen wordt aangepast en naar klanten wordt gedistribueerd.

map program *overzichtsprogramma*

Een softwaregereedschap dat vaak deel van een compileer- of assembleerprogramma uitmaakt en waarin een laadoverzicht wordt gegenereerd.

mark *merkteken*

Eén of meer symbolen die het begin of het einde van een rubriek, een woord, een gegeven, een reeks gegevens, een bestand, een bericht, een record, een blok enzovoort aangeven.

marker *markeringsteken*

Een symbool dat wordt gebruikt om een bepaalde positie aan te geven.

markup *1. markup; 2. opmaak*

Een methode die wordt gebruikt bij de opmaak van automatisch en via computertechnieken te verwerken documenten. Er wordt extra informatie aan de tekst van een document toegevoegd. Deze extra informatie bevat opdrachten en aanwijzingen betreffende de logische elementen van een document en instructies ten aanzien van de wijze waarop de automatische opmaak (layout) van een document moet worden verzorgd.

mask *masker*

Een bitpatroon of tekenpatroon dat zó is geconstrueerd dat het op logische wijze met een onbekend gegevensitem kan worden gecombineerd met het doel delen van dat gegevensitem vast te houden of te onderdrukken. Indien, bijvoorbeeld, de bitrij 00000011 met behulp van een logische En-bewerking met een uit acht bits bestaand

gegevensitem wordt bewerkt, worden de laatste twee bits van dat item vastgehouden en bevatten alle andere bitposities van dat item een 0.

master library *hoofdbibliotheek*

Een softwarebibliotheek die moederkopieën van software en documentatie bevat en waarvan kopieën kunnen worden gemaakt ten behoeve van de distributie en het gebruik.

match (to) *overeenkomen*

Het bepalen van de overeenkomst tussen verschillende (gegevens)elementen.

mathematical check *wiskundige controle*

Een geprogrammeerde controle waarbij van wiskundige relaties gebruik wordt gemaakt.

mathematical function *wiskundige functie*

Een wiskundige uitdrukking die een relatie tussen twee of meer variabelen beschrijft.

McCulloch-Pitts neuron *McCulloch-Pitts neuron*

Het McCulloch-Pitts neuron is het fundamentele bouwblok van neuronale netwerken. Een dergelijk neuron ontvangt één of meer elementen met informatie en levert één of meer identieke elementen af, waarvan elk een eenvoudige niet-lineaire functie van de som van de ontvangen elementen is. De niet-lineaire functie is typisch een drempel- of een stapfunctie die meestal moet worden geëffend om het leerproces te vergemakkelijken.

mean time between failures: MTBF *gemiddelde tijd tussen storingen*

De verwachte of waargenomen tijd tussen de opeenvolgende storingen van een systeem of component.

mean time to repair: MTTR *gemiddelde reparatietijd*

De verwachte of waargenomen tijd die nodig is om een systeem of component te repareren en het weer in de normale operationele werkwijze te plaatsen.

measurement standard *meetstandaard*

Een standaard die de karakteristieke kenmerken en eigenschappen van het evalueren van een proces of een product beschrijft.

member *lid*

Een veld of een methode van een klasse. Een lid is niet statisch tenzij anders gespecificeerd.

member function *lidfunctie*

De naam die in de programmeertaal C++ aan een methode wordt gegeven.

memory *geheugen*

Het geheugengebied of de opslagruimte waar de computer de informatie die op een bepaald moment wordt verwerkt, in opslaat. Dit is slechts nuttig voor het kortstondig opslaan van informatie of gegevens. Voor langdurige opslag worden over het algemeen magneetschijfgeheugens gebruikt.

memory capacity *geheugencapaciteit*

Het maximale aantal items dat in een computergeheugen kan worden opgeslagen; dit wordt meestal uitgedrukt in woorden of bytes.

memory compaction *geheugencompactie*

1. Een geheugenopslagtechniek waarin de inhoud van alle toegekende geheugenlocaties naar het begin van het geheugengebied wordt verplaatst en de overblijvende geheugenblokken in één enkel blok worden gecombineerd.
2. Een geheugenopslagtechniek waarin aaneensluitende blokken met niet-toegekende geheugenplaatsen zodanig worden gecombineerd dat ze één enkel blok vormen.

memory dump *geheugendump*

Een weergave van de inhoud van alle of een deel van het interne computergeheugen. Dit geschiedt meestal in een binaire, octale, of hexadecimale vorm.

memory map *geheugenoverzicht*

Een schema of diagram dat aangeeft waar programma's en gegevens in het computergeheugen zijn opgeslagen.

menu *menu*

Een lijst met gegevens of informatie waaruit keuzes kunnen worden gemaakt.

menu by-pass *menu-omzeiling*

Een mogelijkheid in een menu-gestuurd systeem die ervaren gebruikers de gelegenheid biedt om functies in een opdracht-gestuurde werkwijze uit te voeren zonder dat ze daarbij functies uit het menu behoeven te selecteren.

menu-driven *menu-gestuurd*

De term heeft betrekking op een systeem of werkwijze waarin de gebruiker het systeem met behulp van keuzes uit een menu dirigeert of (aan)stuurt.

merge (to) *samenvoegen*

Het samenvoegen van de gegevens van twee of meer op dezelfde wijze geordende reeksen tot een nieuwe reeks die in dezelfde volgorde is gerangschikt.

message *1. bericht; 2. boodschap*

Indien in een object-georiënteerd programma een bericht of boodschap naar een object wordt verzonden of een methode wordt geactiveerd, kan dit als gelijkwaardig worden beschouwd met het oproepen van een procedure in de meer traditionele programmeertalen. Echter, de actuele code die wordt uitgevoerd kan gedurende de uitvoeringstijd worden geselecteerd. Dit is afhankelijk van de klasse van het object. Als reactie op het bericht "teken" zal de geactiveerde code anders zijn indien het doelobject een cirkel dan wel een vierkant is.

meta-CASE tool *meta-CASE gereedschap*

Een term die soms voor bepaalde softwarepakketten wordt gebruikt. Hiermee kunnen gebruikers hun eigen CASE-gereedschappen aanpassen of zelfs ontwikkelen.

metaclass *metaklasse*

De klasse van een klasse. Een metaklasse waarvan de afgeleiden, de exemplaren of de kopieën zelf weer klassen zijn.

metaclass system: MCS *MCS*

Een portabele, object-georiënteerde uitbreiding van Common Lisp. MCS integreert de functionaliteiten van Common Lisp Object System (CLOS).

metadata *metagegevens*

1. Gegevens waarmee gegevens worden beschreven. Modellen kunnen in feite als metagegevens worden beschouwd; ze beschrijven dingen of zaken die gemodelleerd moeten worden of zijn.
2. Gegevensdefinities die aspecten van de actuele gegevensitems of -elementen beschrijven zoals een naam, een formaat, specifieke eigenschappen enzovoort.

metafile *metabestand*

Een bestand dat grafische informatie bevat. Deze informatie kan met behulp van een metabestand tussen verschillende machines worden getransporteerd.

metalanguage *metataal*

Een taal waarmee bepaalde of alle aspecten van één of meer andere talen kunnen worden beschreven en eventueel geanalyseerd.

meta-object *meta-object*

Een object dat een type, een bewerking, een klasse, een methode, of een andere entiteit van een objectmodel dat objecten beschrijft, vertegenwoordigt.

method *methode*

1. Code die kan worden uitgevoerd (verwerkt) met het doel een gevraagde dienst te verrichten.
2. De naam die in de programmeertaal Smalltalk en in bepaalde andere object-georiënteerde programmeertalen aan een procedure of een routine die met een object is geassocieerd, wordt toegekend.
3. Een functie die in een klasse is gedefinieerd. Tenzij anders is gespecificeerd, moet een methode nooit als statisch worden beschouwd.

method binding *methodebinding*

Zie binding.

methodology *methodologie*

Een term die wordt geassocieerd met een gecodificeerde verzameling procedures die in bepaalde fasen van de software-ontwikkeling worden toegepast. Voorbeelden daarvan zijn de analyse en het ontwerp.

method standard *standaardmethode*

Een standaard die de karakteristieke kenmerken en eigenschappen van een ordelijk proces of procedure beschrijft die in de ontwikkeling van een product of de uitvoering van een dienst wordt gebruikt.

metric *metriek*

1. Een kwantitatieve opgave van de mate waarin een systeem, een component, een asset, of een proces een bepaald attribuut bezit.
2. De definitie, het algoritme, of de wiskundige functie die wordt gebruikt om een kwantitatieve schatting te verkrijgen van onderwerpen zoals het aantal malen dat software opnieuw wordt gebruikt, de betrouwbaarheid van assets, de inspanningen die samenhangen met het hergebruik van assets, of andere karakteristieke eigenschappen of kenmerken van een domein of een softwaresysteem.

microarchitecture *micro-architectuur*

De definities van microwoorden, gegevensstromen, tijdconsequenties, en precedentiebependingen die een met micro-instructies geprogrammeerde computer karakteriseert.

microcode *microcode*

1. Een verzameling van één of meer micro-instructies die alle, een deel van alle, of een serie microprogramma's omvatten.
2. Een code die de reeks instructies representeert die in een gedeelte van het geheugen is geïmplementeerd. Dit deel van het geheugen kan niet door het programma worden geadresseerd. Microcode heeft betrekking op de micro-instructies die in een bepaald product worden gebruikt als een alternatief voor een in hardware geïmplementeerd circuit, een bepaalde functie van een verwerkingseenheid, of een ander deel van een systeem.

microcode assembler *microcode-assembleerprogramma*

Een computerprogramma dat microprogramma's vanuit de symbolische vorm vertaalt naar de binaire vorm.

micro-instruction *micro-instructie*

Een opdracht in de microprogrammering die één of meer basisbewerkingen specificeert die nodig zijn om een opdracht in machinetaal uit te voeren. Specifieke types zijn de diagonale, horizontale en verticale micro-instructies.

microkernel *microkernel*

Een besturingssysteem dat de nadruk legt op kleine modules waarmee de belangrijkste functies en kenmerken van een systeem kunnen worden geïmplementeerd en geconfigureerd.

micro-operation *microbewerking*

Eén van de basisbewerkingen in de microprogrammering die nodig is om een opdracht in de machinetaal uit te voeren.

microprogram *microprogramma*

Een serie opeenvolgende opdrachten (de micro-instructies) die de basisbewerkingen specificeren die nodig zijn om een opdracht in de machinetaal uit te voeren.

microprogrammable computer *microprogrammeerbare computer*

Een met behulp van micro-instructies geprogrammeerde computer waarin microprogramma's door de gebruiker kunnen worden opgesteld en gewijzigd.

microprogrammed computer *met micro-instructies geprogrammeerde computer*

Een computer waarin opdrachten in de machinetaal met behulp van microprogramma's in plaats van bedrade logica worden geïmplementeerd. Opmerking: Een dergelijke computer kan al dan niet een microcomputer zijn; de concepten zijn verschillend, ondanks het feit dat de termen een zekere verwantschap vertonen.

microprogramming *microprogrammering*

Het proces van het ontwerp en de implementatie van de besturingslogica van een computer. Dit geschiedt door de basisbewerkingen te identificeren die nodig zijn om elke afzonderlijke opdracht in de machinetaal uit te voeren en deze bewerkingen als series opeenvolgende opdrachten in een speciaal geheugen, het besturingsgeheugen, te plaatsen. Deze methode is een alternatief voor de methode waarin de besturingssignalen, die nodig zijn om elke opdracht in de machinetaal te verwerken, met behulp van hardware-circuits wordt gerealiseerd.

microword *microwoord*

Een adresseerbaar element in het besturingsgeheugen van een computer die met behulp van micro-instructies is geprogrammeerd.

migrate (to) *migreren*

1. Het verplaatsen van gegevens van een bepaalde opslaghiërarchie naar een andere.
2. Het overgaan naar een gewijzigde verwerkingsomgeving. Deze overgang vindt gewoonlijk naar een nieuwe versie van een systeem plaats.

minimum delay programming *programmering met minimale vertragingstijden*

Een programmeringstechniek waarin geheugenlocaties voor computeropdrachten en gegevens zodanig worden gekozen dat de toegangstijd tot opdrachten en gegevens wordt geminimaliseerd.

mistake *fout*

Een menselijke actie die tot een foutief resultaat leidt. Opmerking: In de discipline van de fouttolerantie maakt men in de Engelse taal een onderscheid tussen een menselijke fout (mistake), de manifestatie van die menselijke fout in de hardware of software (manifestation), het resultaat van een fout (failure), en de mate waarin dat resultaat incorrect is (error).

mixed mode *gemengde modus*

De term verwijst naar een uitdrukking die twee of meer verschillende gegevenstypen bevat. Bijvoorbeeld, $Y := X + N$, waarbij X en Y de variabelen van drijvende-kommagetallen voorstellen en N de variabele van een geheel getal voorstelt.

mnemonic *mnemonische naam*

1. Een symbolische voorstelling van een functie of een verwerkingscode.
2. Een acroniem of korte en gemakkelijk te onthouden naam of afkorting.

mnemonic symbol *mnemonisch symbool*

Een symbool dat als geheugensteun is gekozen, bijvoorbeeld de afkorting van een woord.

mode *werkwijze*

1. Een bewerkingmethode, bijvoorbeeld de binaire of alfanumerieke werkwijze.
2. De statistisch meest voorkomende waarde.
3. Een kenmerk van rekenkundige gegevens, bijvoorbeeld reëel of complex.

modelling *modellering*

De evaluatie van een softwareconcept. Dit geschiedt met behulp van de constructie van een model van de mogelijkheden die de software biedt of te bieden heeft, waarna deze gemodelleerde mogelijkheden in een grote verscheidenheid aan configuraties en gesimuleerde operationele omgevingen kunnen worden uitgevoerd.

model statement *modelopdracht*

Een opdracht in een macrodefinitie of in een open code waaruit, voorafgaande aan het assembleren, een opdracht kan worden gegenereerd. In een modelopdracht kunnen op één of meer punten waarden worden vervangen. Uit dezelfde modelopdracht kunnen één of meer identieke of verschillende opdrachten worden gegenereerd.

modified condition/decision coverage *gemodificeerde conditie/beslissingsdekking*

Het percentage van de resultaten van alle takken, zijpaden of sprongen die de uitkomst van een beslissing onafhankelijk van elkaar beïnvloeden. Dit wordt door een test of een testserie geverifieerd.

modified condition/decision testing *gemodificeerde conditie/beslissingstest*

Het ontwerp van een test of een testserie waarin de testen zodanig zijn geconcipieerd dat de resultaten van takken, zijpaden of sprongen in een programma de uitkomst van een beslissing onafhankelijk van elkaar beïnvloeden.

modifier *1. modificatie-element; 2. wijzigingselement*

Een informatie-eenheid die wordt gebruikt om een programma-instructie te wijzigen zodat deze instructie bij elke toepassing een andere bewerking uit kan voeren.

modify (to) *modificeren*

Het wijzigen van een gedeelte van een instructie of een deel van een programma.

MODSIM *MODSIM*

MODSIM is het acroniem voor modular simulator. MODSIM is een algemeen toepasbare modulaire blok-gestructureerde taal die geschikt is voor de object-georiënteerde programmering en de simulatie van discrete gebeurtenissen. De taal kan worden gebruikt voor de realisatie van grote proces-georiënteerde discrete modellen van discrete gebeurtenissen en object-georiënteerde mechanismen zoals die welke met behulp van Modula-2 kunnen worden gerealiseerd.

Modula-2 *Modula-2*

Een hoog-niveau programmeertaal die als een verbetering van Pascal kan worden beschouwd. De taal is beter geschikt voor de administratieve en bestuurlijke informatieverwerking dan Pascal en kan eveneens als een ontwikkelingsgereedschap voor software worden gebruikt.

modular *modulair*

Opgebouwd uit delen die elk op zich als één geheel kunnen worden beschouwd. De term heeft zowel betrekking op apparatuur als programmatuur.

modular decomposition *modulaire decompositie*

Het proces waarin een systeem in haar componenten wordt opgebroken of opgedeeld met het doel het ontwerp en de ontwikkeling te vereenvoudigen; dit is een aspect van de modulaire programmering.

modularity *modulariteit*

De mate waarin een systeem uit discrete modules of componenten bestaat of daaruit is opgebouwd en waarbij geldt dat die modules of componenten andere modules of componenten zo weinig mogelijk beïnvloeden.

modular programming *modulaire programmering*

Programmeren waarbij discrete modules worden gecodeerd om bepaalde functies te verrichten. Deze modules zijn onderling consistent.

module *module*

1. Een deel van een programma dat een afzonderlijke en discrete eenheid vormt en als zodanig kan worden geïdentificeerd ten behoeve van het compileren, het combineren ervan met andere modules en het laden.
2. Een functionele en op zich zelf staande eenheid die is ontworpen om in combinatie met andere componenten, modules, eenheden, of elementen te worden gebruikt.
3. Een component dat òf uit code òf uit een ontwerp bestaat. Voorbeelden van modules zijn broncodemodules, doelcodemodules, en ontwerpmodules.

Opmerking: De termen 'module', 'component' en 'eenheid' zijn meestal onderling uitwisselbaar of zijn zodanig gedefinieerd dat ze op verschillende manieren sub-elementen van elkaar kunnen zijn, afhankelijk van de context waarin ze worden gebruikt of voorkomen. De relaties tussen deze termen zijn echter (nog) niet gestandaardiseerd.

module strength *modulecohesie*

Zie cohesie (cohesion).

monadic selective construct *monadische selectieve constructie*

Een IF-THEN-ELSE bewerking waarin de verwerking slechts voor één resultaat van de tak is gespecificeerd; het resultaat van de andere tak heeft het overslaan van deze verwerking tot gevolg.

monitor *monitor*

Een softwaregereedschap dat of hardware-eenheid die tezamen en tegelijkertijd met een systeem of component operationeel is en toezicht houdt op de werking van het systeem of component, of die werking vastlegt, analyseert of verifieert.

Motif *Motif*

Een grafische gebruikersinterface die gebaseerd is op het X Window systeem en door de Open Software Foundation wordt gedistribueerd.

mounting directories *koppelindexen*

Indexen (directories) die de wortelindex van een bepaald bestand aan de index van een ander bestand koppelen. Alle bestanden kunnen op deze wijze in het bestandssysteem als deelindexen (sub-directories) worden gehanteerd.

move (to) *verplaatsen*

1. Het lezen van gegevens uit een bron, waarbij de inhoud van de locatie van de bron wordt gewijzigd, en dezelfde gegevens op een andere plaats in een bepaalde fysieke voorstellingswijze of vorm worden weggeschreven. Daarbij kan die laatste voorstellingswijze of vorm verschillend zijn van die waarin de gegevens in de bron waren opgeslagen.
2. Soms is deze term synoniem met die van 'kopiëren' (to copy).

multi-address instruction *opdracht met een meervoudig adres*

Een computeropdracht die meer dan één adresveld bevat.

multilevel address *meervoudig-niveau adres*

Zie indirect adres.

multiple exclusive selective construct *meervoudig-exclusieve selectieve constructie*

Zie CASE-clausule (CASE-clause).

multiple inclusive selective construct *meervoudig-inclusieve selectieve constructie*

Een speciale vorm van de CASE-clausule waarin twee of meer verschillende waarden van de besturende opdracht dezelfde verwerking tot gevolg hebben.

multiple inheritance *meervoudige overerving*

1. In de meervoudige overerving kan een klasse meer dan één enkele superklasse hebben en de eigenschappen en kenmerken van alle ouders erven.
2. De mogelijkheid in de object-georiënteerde programmering dat een deelklasse van meerdere ouders kan worden afgeleid. Deze ouders zijn echter nooit van elkaar afgeleid.

multiple instruction multiple data: MIMD *meervoudige instructies en meervoudige gegevens*

Een bepaalde vorm van gelijktijdigheid die in computers met meervoudige verwerkingseenheden wordt toegepast en waarmee verschillende instructiestromen (programma's) gelijktijdig verschillende gegevensstromen verwerken.

multiprocessing *meervoudige verwerking*

Het gelijktijdig of parallel en onder centrale besturing uitvoeren van twee of meer programma's of instructiereeksen door een computer, een verwerkingseenheid, een netwerk enzovoort.

multiprogramming *1. meervoudige programmering; 2. multiprogrammering*

- a. Het programmeren en het daaropvolgend gelijktijdig uitvoeren van twee of meer programma's door een computer.
- b. Het programmeren en het daaropvolgend afwisselend uitvoeren van twee of meer programma's door één enkele verwerkingseenheid.

multiprogramming system *multiprogrammeringssysteem*

Een computersysteem dat gelijktijdig twee of meer programma's kan verwerken door de uitvoering ervan afwisselend plaats te laten vinden.

multiprogramming with a fixed number of tasks *multiprogrammering met een vast aantal taken*

Een besturingsprogramma dat toezicht houdt op de uitvoering van een vast aantal taken dat zich in het hoofdgeheugen bevindt en dat systeemfaciliteiten en -elementen aan die taken toewijst.

multiprogramming with a variable number of tasks *multiprogramming met een variabel aantal taken*

Een besturingsprogramma dat toezicht houdt op de uitvoering van een variabel aantal taken dat zich in het hoofdgeheugen bevindt en dat systeemfaciliteiten en -elementen aan die taken toewijst.

multitasking *meervoudige verwerking van taken*

1. Het gelijktijdig uitvoeren van twee of meer taken die zich in hetzelfde geheugengebied of dezelfde geheugenpartitie bevinden.
2. Multiprogrammering met het gelijktijdig of beurtelings uitvoeren van twee of meer taken.

multitask operation *meertakenverwerking*

1. Een vorm van multiprogrammering waarbij twee of meer taken gelijktijdig worden verwerkt.
2. Het gebruik van één enkel programma door meerdere taken. Dit programma wordt steeds opnieuw ingevoerd of geladen.

multithreaded *meervoudig aaneengeregen*

De term verwijst naar een programma dat zodanig is ontworpen dat delen van de code tegelijkertijd kunnen worden uitgevoerd.

mutation analysis *afwijkingsanalyse*

Een methode voor het bepalen van de kwaliteit van een test of testserie door de mate waarin een testserie of een test het programma kan onderscheiden van kleine afwijkingen van dat oorspronkelijke programma, waar te nemen en vast te leggen.

mutation test *mutatietest*

Een testmethodiek waarin twee of meer van elkaar afwijkende programma's worden verwerkt en waarbij van dezelfde testen of testreeksen gebruik wordt gemaakt om te evalueren in hoeverre de testen of testreeksen de verschillen tussen de mutaties van de programma's detecteren.

N

n-address instruction *n-adresinstructie*

Een computerinstructie die *n* adresvelden bevat; *n* kan elk niet-negatief geheel getal zijn.

name *naam*

1. Een combinatie van alfanumerieke tekens waarin over het algemeen tenminste één letter voor dient te komen. Een naam bepaalt de identiteit van een gegevensverzameling, een besturingsopdracht of een procedure.
2. Een woord dat uit een voorgeschreven aantal tekens bestaat en dat een opdrachtelelement omschrijft.

name constant *naamconstante*

Een gegevenstype dat de variabele van de interne naam van een object voorstelt.

nano-instruction *nano-instructie*

In een implementatie van een microprogramma dat uit twee niveaus bestaat, is dit een instructie die één of meer basisbewerkingen specificeert die nodig zijn om een micro-instructie uit te voeren.

nanostore *nano-geheugen*

In de microprogrammering met twee niveaus verwijst de term naar een secundair besturingsgeheugen waarin de nano-instructies zijn opgeslagen.

national database language: NDL *NDL*

Een Amerikaanse standaard voor de overdraagbaarheid van databasedefinities en applicatieprogramma's.

national software reuse directory: NSRD *nationale index van herbruikbare software*

Een index van de herbruikbare software in het ASSET-systeem, een project van het Amerikaanse Ministerie van Defensie dat het hergebruik van software bevordert.

native language system: NLS *oorspronkelijke-taalsysteem*

Een serie interfaces die door de X/Open organisatie zijn gespecificeerd en die gebruikt kunnen worden voor de ontwikkeling van applicaties die in omgevingen, waarin verschillende natuurlijke talen worden gehanteerd, kunnen worden verwerkt.

natural language *natuurlijke taal*

Een taal waarvan de regels zijn gebaseerd op het algemeen gangbare gebruik zonder dat deze specifiek zijn omschreven. Een natuurlijke taal is een taal die nimmer gedefinieerd of gespecificeerd wordt voordat de taal als algemeen bruikbaar kan worden beschouwd.

natural language system *natuurlijke-taalsysteem*

Een systeem waarmee men, zonder gebruik te maken van een programmeertaal, gegevens uit een computer kan verkrijgen door vragen in een natuurlijke taal te stellen.

negative testing *negatief testen*

Het uitvoeren van testen die er op zijn gericht om aan te tonen dat de software (of een systeem) niet werkt.

Neptune *Neptune*

Een hypertext-systeem ten behoeve van computer-ondersteunde software-ontwikkeling. Neptune is een product van de firma Tektronix.

nest (to) *nesten*

1. In een programma heeft de term betrekking op het insluiten van een structuur of structuren van een bepaald type in een structuur van hetzelfde type; bijvoorbeeld, het nesten van een subroutine in een andere subroutine, een blok in een ander blok, een lus in een andere lus, of het nesten van een module in een ander module.
2. Het opsluiten of inkapselen van programma-onderdelen of gegevens in andere programma-onderdelen of gegevens die een verschillend hiërarchisch niveau hebben, zodat de programma-onderdelen of gegevens op de verschillende niveaus herhaald kunnen worden uitgevoerd respectievelijk toegankelijk kunnen worden gemaakt.

nested DO-group *geneste DO-groep*

Een DO-groep die zich binnen een andere DO-groep bevindt.

nested procedure *geneste procedure*

Een procedure die door een andere procedure wordt opgeroepen.

nested subroutine *geneste subroutine*

Een subroutine die vanuit een andere subroutine wordt opgeroepen.

nesting level *nestend niveau*

1. Het niveau waarop een term of een deeluitdrukking in een uitdrukking voorkomt.
2. Het niveau waarop een macrodefinitie, die een ingesloten macro-instructie bevat, door een vertaalprogramma wordt verwerkt.

network *netwerk*

Een serie computers die met behulp van kabels, telefoonlijnen of satellieten onderling zijn verbonden.

network application support: NAS *netwerkaplicatie-ondersteuning*

De benadering van de firma DEC voor de integratie van applicaties in een gedistribueerde omgeving van apparatuur die door verschillende leveranciers wordt of is geïnstalleerd.

network extensible Window system: NeWS *netwerk-uitbreidbaar Window-systeem*

Een systeem ontwikkeld door Sun Microsystems dat functies en mogelijkheden biedt die overeenkomen met die van het X Window systeem. De communicaties zijn op Postscript gebaseerd, terwijl de functies van de server kunnen worden uitgebreid.

neural network *neuronaal netwerk*

Een verwerkingseenheid die of bepaald soort computer dat één of meer invoersignalen omzet in één of meer uitvoersignalen. Dit geschiedt met behulp van een onderling gekoppelde serie elementaire niet-lineaire verwerkingseenheden die neuronen worden genoemd. De hersenen van dieren vormen een voorbeeld van een biologisch neuronaal netwerk. Kunstmatige neuronale netwerken zijn door de mens vervaardigde rekeneenheden waarvoor hun biologische tegenhangers model hebben gestaan. De kenmerkende eigenschappen die neuronale netwerken doen onderscheiden van de traditionele Von Neumann-computers zijn: (a) de elementaire processoren zijn in sterke mate niet-lineair; (b) de neuronen zijn onderling sterk gekoppeld, iets waarmee een hoge graad van paralleliteit kan worden bereikt; en (c) er bestaat geen vrij geheugen dat gegevens en programma's bevat, elk neuron is vooraf geprogrammeerd en steeds actief.

neuron *neuron*

Zie neuronaal netwerk (neural network).

NewWave *NewWave*

Een grafische gebruikersinterface en object-georiënteerde omgeving van de firma Hewlett-Packard. De interface is gebaseerd op Windows en is beschikbaar op UNIX-werkstations.

NeXTstep *NeXTstep*

Een systeem waarmee grafische interfaces kunnen worden ontwikkeld, object-georiënteerde toepassingen kunnen worden geschreven en window-achtige software voor NeXT en IBM AIX-systemen bevat.

n-level address *n-niveau-adres*

Een indirect adres dat de eerste van een keten van n geheugenlocaties specificeert; elk van de eerste $n-1$ geheugenlocaties van die keten bevat het adres van de volgende locatie in de keten en de laatste daarvan bevat de gewenste operand.

node *knooppunt*

1. Een punt, cirkel of andere geometrische figuur die in een schema wordt gebruikt om een toestand, een gebeurtenis, of een ander significant item voor te stellen.

2. Zie graaf (graph).

nomenclature standard *nomenclatuurstandaard*

Een standaard die de karakteristieke kenmerken en eigenschappen van een systeem of een reeks namen, toewijzingen, of symbolen beschrijft.

nondestructive read *niet-destructieve leesopdracht*

Een leesbewerking die de gegevens in de locatie die wordt uitgelezen, niet wist.

non-functional requirements testing *testen van niet-functionele eisen*

Het uitvoeren van testen van die eisen die geen relatie met de functionaliteit van een systeem hebben. Dit omvat de bruikbaarheid, de uitvoering, de snelheid enzovoort.

non-procedural language *niet-procedurele taal*

Een taal waarin de gebruiker vastlegt wat hij/zij met de taal wenst te bereiken zonder dat daarbij de specifieke instructies worden gebruikt die de computer in een bepaalde volgorde uit moet voeren.

no-operation: no-op *zonder effectieve bewerking*

Een computerbewerking waarvan de uitvoering geen enkel effect op de uitvoering van een reeks instructies heeft anders dan het ophogen van de opdrachtteller naar de volgende instructie. Een dergelijke bewerking wordt gebruikt om ruimte in een programma te reserveren of, indien deze herhaaldelijk wordt uitgevoerd, op een bepaalde gebeurtenis te wachten.

notation standard *notatiestandaard*

Een standaard die de karakteristieke kenmerken en eigenschappen van de formele interfaces binnen een bepaald beroep beschrijft.

NoteCards *NoteCards*

Een ambitieus hypertext-systeem dat door het Palo Alto Research Centrum van Xerox is ontwikkeld. Het systeem voert de transformatie uit van chaotische denkbeelden die geen onderlinge relaties bezitten en werkt deze om tot een geordende en geïntegreerde interpretatie van ideeën en de samenhang tussen deze ideeën.

notice of revision: NOR *wijzigingsnota*

Een formulier dat deel uitmaakt van het configuratiebeheer waarin voorstellen voor het herzien of wijzigen van een tekening, lijst of overzicht worden voorgesteld en na goedkeuring daarvan de gebruikers inlicht over het feit dat de tekening, de lijst of het overzicht als zodanig is of zal worden gewijzigd.

n-plus-one address instruction *n-plus-één adresinstructie*

Een computerinstructie die $n+1$ adresvelden bevat; het laatste veld bevat het adres van de instructie die als eerstvolgende instructie moet worden uitgevoerd.

N-switch coverage *N-switch dekking*

Het percentage van de opeenvolgende N-overgangen die door een test of een testserie zijn ontdekt.

N-switch testing *N-switch testen*

Het testen van toestandsovergangen waarin de test of de testseries zodanig zijn geconcipieerd dat ze alle geldige opeenvolgende N-overgangen uitvoeren.

N-transitions *N-overgangen*

Een reeks $N+1$ overgangen.

O

object *object*

1. Een logisch systeemelement of -component dat door een supervisieprogramma wordt beheerd. Aan elk object wordt een naam toegekend zodat naar dit element of component kan worden verwezen.
2. Een combinatie van een toestand en een serie methoden die een abstractie, gekenmerkt door het gedrag van relevante verzoeken, op expliciete wijze belichaamt. Een object is een entiteit die zowel een toestand als functionaliteit bezit.
3. In de object-georiënteerde programmering verwijst de term naar een exemplaar of een kopie van een gegevensstructuur die met behulp van een model (template) wordt gedefinieerd. Dit model wordt bepaald door de klasse van het model en de mogelijkheid om op berichten die door de klasse zijn gedefinieerd, te kunnen reageren.
4. Het belangrijkste bouwblok van de object-georiënteerde programmering. Elk object bestaat uit een programma-eenheid die uit gegevens en functies bestaat en een inkapseling is van gegevens, functies en/of services (methoden) die die gegevens kunnen bewerken.
5. De term heeft betrekking op het resultaat van een assembleer- of compilatieproces. Object wordt hier gebruikt in termen zoals objectcode, objectmodule, en objectprogramma.
6. Een programmaconstante of -variabele.

object broker *objectbemiddelaar*

Een gedistribueerd objectsysteem van de firma DEC dat gebaseerd is op de CORBA-standaard.

object code *objectcode*

De uitvoer van een compileer- of een assembleerprogramma die uit een code bestaat die de computer uit kan voeren of die geschikt is om te worden omgezet in een code die de computer kan verwerken.

object compatibility standard: OCS *object-compatibiliteit-standaard*

Een open standaard voor compilers en link-programma's.

object creation *objectrealisatie*

Een gebeurtenis die het ontstaan van een object tot gevolg heeft. Het op deze wijze gerealiseerde object onderscheidt zich van elk ander object.

object destruction *objectvernietiging*

Een gebeurtenis die de vernietiging van een object tot gevolg heeft. Zowel het vernietigde object als de daarmee geassocieerde (hulp)middelen, systeemelementen of hulpbronnen zijn niet meer voor hergebruik beschikbaar.

object interface *objectinterface*

Een beschrijving van een aantal mogelijke toepassingen of gebruiksmogelijkheden van een object.

objective C *objective C*

Een Smalltalk-achtige uitbreiding van de C-taal waarmee de mogelijkheid wordt gecreëerd om object-georiënteerde programma's te ontwikkelen.

objective Pascal *objective Pascal*

Een uitbreiding van de Pascal-taal ten behoeve van de object-georiënteerde programma-ontwikkeling.

object language *1. doeltaal; 2. objecttaal*

Een taal die door een metataal wordt gespecificeerd.

object linking and embedding: OLE *objectkoppeling en inbedding*

Een gedistribueerd objectsysteem van de firma Microsoft.

Object Management Group: OMG *objectbeheersgroep*

Een consortium dat zich bezig houdt met het opstellen van standaarden in de object-georiënteerde programmering.

object management system *object-beheerssysteem*

Een systeem in een omgeving voor de geïntegreerde programmeringsondersteuning (IPSE) dat informatie betreffende het in ontwikkeling zijnde systeem bijhoudt.

object management tool: OMTool *OMTool*

Een grafisch gereedschap voor het ontwerp en de analyse van systemen die met behulp van de object-georiënteerde methodologie worden vervaardigd. Het gereedschap bevat bepaalde codegeneratiefaciliteiten voor C++/SQL.

object module *1. objectmodule; 2. werkmodule*

Een module die de uitvoer van een compileer- of assembleerprogramma en de invoer van een montageprogramma (link-programma) vormt.

object name *objectnaam*

1. De naam die aan een object is toegekend.
2. Een waarde die een object identificeert.

object-oriented *object-georiënteerd*

De term is van toepassing op de analyse, het ontwerp en de vervaardiging van programma's. Het fundamentele concept in deze benadering is dat van objecten welke uit gegevensstructuren en een aantal routines of services (of methoden) bestaan. Deze kunnen de gegevens die in het object zijn verpakt, bewerken of verwerken, of kunnen communicaties tussen de objecten en eventuele externe eenheden verzorgen. Bewerkingen op de gegevens moeten met behulp van deze methoden worden uitgevoerd. Daarbij zijn de methoden van toepassing op alle exemplaren of kopieën van objecten die zich een speciale klasse bevinden. Op deze wijze is de interface naar de objecten goed gedefinieerd en kan de code, waarmee de methoden zijn geïmplementeerd, worden gewijzigd. Daarbij geldt echter wel de eis dat de interfaces ongewijzigd blijven.

object-oriented analysis: OOA *object-georiënteerde analyse*

Analyse van een probleem waarbij men gebruik maakt van de methodologieën uit de object-georiënteerde technologie.

object-oriented database *object-georiënteerde database*

Een systeem dat database-management faciliteiten in een object-georiënteerde omgeving heeft.

object-oriented database management system: OODBMS *object-georiënteerd databasebeheerssysteem*

Een beheerssysteem dat geheel op de object-georiënteerde technologie is afgestemd.

object-oriented decomposition *object-georiënteerde decompositie*

De term heeft betrekking op het opdelen van een systeem in objecten die toegankelijk zijn voor andere systemen of objecten die voldoen aan een voor dat object ontworpen standaard protocol.

object-oriented design *object-georiënteerd ontwerp*

Een ontwerpmethode voor softwarecomponenten en -systemen waarin de karakteristieke kenmerken en eigenschappen van abstracte of reële objecten kunnen worden gemodelleerd en waarbij gebruik wordt gemaakt van klassen en objecten. Een component of een systeem wordt in termen van objecten en connecties tussen die objecten uitgedrukt.

object-oriented design diagrams *object-georiënteerde ontwerpdigrammen*

Diagrammen of schema's die worden gebruikt om de aanwezigheid van objecten en hun onderlinge relaties in het logische ontwerp van een systeem aan te tonen. Eén enkel diagram vertegenwoordigt het geheel of een deel van de objectstructuur van een systeem. Voor het totale ontwerp van een systeem is een serie van deze diagrammen nodig.

object-oriented language *object-georiënteerde taal*

Een programmeertaal die de gebruiker ervan in staat stelt een programma in termen van objecten, berichten tussen die objecten en/of functies die van toepassing zijn op gegevens van die objecten uit te drukken. Voorbeelden van object-georiënteerde talen zijn Smalltalk, Eiffel en C++.

object-oriented programming: OOP *object-georiënteerde programmering*

Programmering waarbij gebruik wordt gemaakt van object-georiënteerde methoden en technieken. Zie object-georiënteerd.

object-oriented programming language: OOPL *object-georiënteerde programmeertaal*

Een programmeertaal (zoals C++, Eiffel, of Objective C), die ontworpen is ten behoeve van de programmering in de object-georiënteerde technologie.

object-oriented structured design: OOSD *object-georiënteerd gestructureerd ontwerp*

Een ontwerpmethode die is afgeleid van de methodologie voor gestructureerd ontwerp waarin de essentiële eigenschappen en mogelijkheden van de object-georiënteerde benadering zijn verwerkt.

object oriented technology: OOT *object-georiënteerde technologie*

Een programma-ontwikkelingsmethodiek waarbij de begrippen 'procedure' en 'algoritme', 'proces', of 'functie' zijn vervangen door 'objecten' en 'berichten' of 'methoden'. De objecten bestaan uit oproepbare en uitvoerbare functies of services (methoden) die de gegevens, die tot een object behoren, kunnen verwerken of bestaan uit berichten waarmee de objecten onderling en met de buitenwereld

kunnen communiceren en transacties kunnen plegen. Zie ook: object-georiënteerd.

object owner *objecteigenaar*

Degene die een object realiseert of aan wie het eigenaarschap van een object is overgedragen. De objecteigenaar heeft de volledige controle en de verantwoordelijkheid over het object.

object program *1. objectprogramma; 2. werkprogramma*

1. Een geheel gecompileerd of geassembleerd programma dat gereed is om in de computer voor verdere verwerking of uitvoering te worden geladen.
2. Het gedeelte van een object- of werkmodule dat een deel van een besturing vormt. Het vormt eveneens de uitvoer van een macro-assembleertaal.

object request broker: ORB *bemiddelingsfunctie bij verzoeken van en aan objecten*

Een term die verwijst naar de middelen die worden gebruikt bij vragen die aan objecten worden gesteld en de antwoorden die objecten geven, alsmede de vragen die door objecten worden gesteld en de antwoorden die worden ontvangen.

object services *objectdiensten*

Basisfuncties ten behoeve van het beheer van de levenscyclus van objecten en het opslaan van objecten, zoals realisatie/vernietiging, activering/passivering, identificatie/locatie.

object type *objecttype*

1. De gegevenskenmerken die het doel van een object binnen een systeem omschrijven. Elk objecttype is geassocieerd met een reeks opdrachten waarmee dat objecttype wordt verwerkt.
2. Een type waarvan de uitbreiding een serie objecten is (letterlijk, een serie waarden waarmee die objecten kunnen worden geïdentificeerd).

Occam *Occam*

Een programmeertaal voor transputers waarmee parallele programma's kunnen worden geschreven. De taal verschaft de programmeur de mogelijkheid te specificeren of processen sequentieel dan wel parallel dienen te worden uitgevoerd.

occupational title standard *beroepenstandaard*

Een standaard waarin de karakteristieke kenmerken van algemene werkterreinen of beroepen zijn beschreven.

off-line *off line*

De term heeft betrekking op een eenheid of een proces dat niet direct door de centrale verwerkingseenheid van een computer wordt bestuurd.

offset *1. verschuiving; 2. verschil*

- a. Het verschil tussen de geladen oorspronkelijke versie en de geassembleerde oorspronkelijke versie van een computerprogramma.
- b. Een getal dat moet worden opgeteld bij een relatief adres om het adres van de geheugenlocatie te verkrijgen. Dit getal kan het in punt 1 van deze definitie genoemde verschil zijn of kan een ander getal zijn dat in het programma is gedefinieerd.

one-address instruction *één-adres opdracht*

Een computerinstructie die één adresveld bevat. Bijvoorbeeld, een opdracht voor het laden van de inhoud van een register.

one-ahead addressing *één-voorwaartse adressering*

Een impliciete adresseringsmethode waarin wordt aangenomen dat de operanden van een computerinstructie zich in de geheugenlocaties bevinden die volgen op de locaties van de operanden die werden gebruikt voor de laatst uitgevoerde opdracht.

one-level address *één-niveau adres*

Zie absoluut adres en direct adres.

one-plus-one address instruction *één-plus-één adresinstructie*

Een computerinstructie met twee adresvelden; het tweede veld bevat het adres van de instructie die na de eerste instructie moet worden uitgevoerd. Bijvoorbeeld, een opdracht voor het laden van de inhoud van locatie A, gevolgd door de uitvoering van de opdracht in locatie B.

on-line *on line*

1. De term verwijst naar een systeem of een werkwijze waarin invoergegevens de computer direct vanaf het punt van oorsprong binnenkomen of uitvoergegevens die direct naar het punt worden getransporteerd waar ze zullen worden gebruikt.
2. De term heeft betrekking op een eenheid of een proces dat direct door de centrale verwerkingseenheid van een computer wordt bestuurd.

Open Look *Open Look*

Een grafische gebruikersinterface die gebaseerd is op het X Window systeem en ontwikkeld is door het Unix Systems Laboratory (USL). Open Look is eigendom van AT&T en Sun Microsystems.

Open Software Foundation: OSF *OSF*

Een samenwerkingsverband dat door negen computerfabrikanten in het leven is geroepen. De oorspronkelijke leden waren: Apollo, DEC, Hewlett-Packard, IBM, Bull, Nixdorf, Philips, Siemens en Hitachi. Inmiddels zijn Apollo, DEC en de computerdivisie van Philips in andere handen overgegaan. Het doel van het samenwerkingsverband was de mogelijkheid open systemen en open informatieverwerking te steunen en uit te dragen. Het is nog steeds de bedoeling van dit consortium om een gemeenschappelijk besturingssysteem (Unix) en grafische interface (X Window Systeem) op een grote serie computers van verschillend makelij beschikbaar te maken.

open subroutine *open subroutine*

Een subroutine die op elke plaats waar deze subroutine wordt opgeroepen, in het computerprogramma wordt gekopieerd.

OpenWindows *OpenWindows*

Een server voor SUN-werkstations die voorzien is van protocollen voor, onder andere, het X Window systeem.

operand *operand*

Een variabele, constante, of functie waarop een bewerking kan worden uitgevoerd. Zo zijn in de uitdrukking $A = B + 3$, B en 3 de operanden.

operating system *1. besturingssysteem; 2. bedrijfssysteem*

- a. Een verzameling software-, firmware-, en hardware-elementen die de uitvoering van computerprogramma's besturen en bepaalde diensten in een computersysteem verzorgen zoals het toewijzen van systeemelementen, de besturing van karweien, de invoer en de uitvoer, alsmede het beheren van bestanden.
- b. Een speciaal programma dat de wijze waarop computer, toetsenbord, beeldscherm, en randgeheugen samenwerken, bestuurt. Bekende besturings- of bedrijfssystemen zijn DOS, UNIX, AIX, OS/2, enzovoort.

operation *1. bewerking; 2. bedrijfsproces*

- 1a. De actie die door een bewerkingsteken op één of meer operanden wordt uitgevoerd. Bijvoorbeeld, in de uitdrukking $A = B + 3$, is dit het proces waarin 3 bij B wordt opgeteld om A te verkrijgen.
- 1b. Een gedefinieerde actie die door een computer wordt uitgevoerd, zoals bijvoorbeeld een optelbewerking, vergelijkingsbewerking, schuifbewerking enzovoort. Een dergelijke actie wordt gewoonlijk in en door het opdrachtgedeelte van een computerinstructie gespecificeerd.
2. Het proces waarin een computersysteem in de voor dit systeem geëigende omgeving werkt met het doel dit computersysteem de daarvoor bestemde functies uit te laten voeren.

operational *operationeel*

1. De term heeft betrekking op een systeem of component dat in de voor dat systeem of component geëigende omgeving voor gebruik gereed is.
2. Betrekking hebbende op een systeem of component dat in een daarvoor geëigende omgeving is geïnstalleerd.
3. De term verwijst naar een omgeving waarin een systeem of component volgens de specificaties van het systeem of component kan functioneren.

operational testing *operationeel testen*

Testen die worden uitgevoerd om na te gaan hoe een systeem, component of element van een systeem zich in een operationele omgeving gedraagt.

operation and maintenance phase *bedrijfs- en onderhoudsfase*

De tijdsperiode van de levenscyclus van software gedurende welke een softwareproduct in de voor dat product geëigende omgeving werkt, op de prestaties ervan wordt gecontroleerd, en afhankelijk van de noodzaak problemen te moeten corrigeren of op gewijzigde omstandigheden te moeten reageren, wordt gemodificeerd.

operation code: op code *bewerkingscode*

Een code die wordt gebruikt om instructies weer te geven. Zo heeft de code BNZ de betekenis van 'branch if not zero'.

operation control statement *bewerkingsbesturingsopdracht*

Een opdracht in een bewerking of in een bewerkingsstap die wordt gebruikt om de identiteit van deze bewerking te bepalen of de bijbehorende eisen van de opdracht aan het besturingssysteem te omschrijven.

operation exception *bewerkingsexceptie*

Een exceptie die optreedt indien een programma een niet-juiste bewerkingscode tijdens de verwerking van het programma tegenkomt.

operation expression *bewerkingsuitdrukking*

Een uitdrukking die één of meer bewerkingstekens bevat.

operation field *bewerkingsveld*

Het veld in een computerinstructie waarin de bewerking die moet worden uitgevoerd, is gespecificeerd.

operation part *bewerkingsdeel*

Een gedeelte van een instructie dat een expliciete specificatie van de te verrichten bewerking bevat.

operator *1. bewerkingsteken; 2. bedieningsfunctionaris; 3. operateur*

- 1a. Een symbool dat de actie weergeeft die in een wiskundige bewerking moet worden verricht.
- 1b. Bij het beschrijven van een proces verwijst de term naar het teken dat de actie aangeeft die op de opdrachtelelementen moet worden verricht.
2. Een persoon die een computersysteem bedient.
3. Zie punt 2 van deze definitie.

operator manual *handboek van de bedieningsfunctionaris*

Een document dat de informatie bevat die noodzakelijk is voor het activeren (initiëren) en de werking van een systeem of component. In een dergelijk document worden de volgende procedures beschreven: procedures voor de voorbereiding, de werking, het in bedrijf houden, de controle, en het eventueel herstellen van een systeem of component. Opmerking: Deze handleiding verschilt van de gebruikershandleiding voor zover er een onderscheid wordt gemaakt tussen hen die een systeem bedienen (opzetten, wegnemen van magneetbanden en -schijven) en hen die het systeem gebruiken voor het doel waarvoor het is geïnstalleerd.

opportunistic reuse *opportunistisch hergebruik*

Het ad hoc hergebruik van assets bij de ontwikkeling van softwaresystemen. Daarbij wordt een programma-ontwikkelingsproces gebruikt dat niet is aangepast aan het systematisch hergebruik van bestaande programmatuur. De ontwikkelaar bepaalt in het geval van het opportunistische hergebruik waar dit hergebruik kan worden toegepast voor de ontwikkeling van een softwaresysteem. Een dergelijke ontwikkeling vindt plaats zonder dat daarbij gebruik wordt gemaakt van de producten van de domeinontwikkeling in de opeenvolgende stadia van het ontwikkelingsproces.

option *optie*

De term heeft betrekking op de mogelijkheid een instructie in een (besturings)taal van extra informatie te voorzien. Hiermee kan het programma een andere route volgen of andere functies oproepen. Zo kan aan de ls-instructie in Unix de optie -l worden toegevoegd. Het resultaat van een dergelijke instructie is dat een bestand wordt afgedrukt met meer informatie dan het geval was geweest indien die optie niet aan deze instructie was toegevoegd.

oracle *oracle*

Een mechanisme waarmee de voorspelde resultaten van een test kunnen worden vergeleken met de werkelijke resultaten van een test.

order clash *sorteringsconflict*

Een soort structureel conflict in de ontwikkeling van software waarin een programma met twee gegevensverzamelingen te maken krijgt die op verschillende manieren zijn gesorteerd.

origin *1. oorsprong; 2. beginadres*

Het adres van de initiële geheugenlocatie die aan een computerprogramma in het hoofdgeheugen wordt toegewezen.

outcome *testresultaat*

Het werkelijke of het voorspelde resultaat van een test.

output *uitvoer*

1. Het uitvoerstadium.
2. Een eenheid, proces of kanaal dat bij een uitvoerproces is betrokken.
3. De gegevens of omstandigheden die verband houden met een uitvoerproces.
4. Een variabele die door een component of element wordt afgeleverd. De variabele kan zowel binnen als buiten het component of element worden opgeslagen.
5. De term heeft betrekking op gegevens die naar een externe eenheid zijn getransporteerd.

output (to) *verzorgen van de uitvoer*

Gegevens naar een externe eenheid of bestemming transporteren.

output assertion *uitvoerassertie*

Een logische uitdrukking die één of meer condities specificceert waaraan de uitvoer van een programma moet voldoen om dat programma als correct te kunnen beschouwen.

output domain *uitvoerdomein*

De verzameling van alle mogelijke uitvoergegevens of af te leveren gegevens.

output file *uitvoerbestand*

1. Een bestand dat de resultaten van het verwerken bevat.
2. Een bestand dat is geopend om het schrijven van records mogelijk te maken.

output line *uitvoerregel*

Een tekstregel die door een opmaakprogramma wordt gerealiseerd.

output mode *uitvoerwerkwijze*

Een werkwijze waarbij records naar een bestand kunnen worden geschreven.

output program *uitvoerprogramma*

Een functioneel programma dat het uitvoerproces van een computer organiseert en realiseert.

output value *uitvoerwaarde*

Een exemplaar of kopie van een uitvoer.

overflow exception *overloopexceptie*

Een exceptie (uitzonderingstoestand) die optreedt indien het resultaat van een rekenkundige bewerking de grootte van de geheugenlocatie(s) die aangewezen is (zijn) om dat resultaat in op te slaan, overschrijdt.

overhead time *overheadtijd*

De tijd die een computersysteem besteedt aan de uitvoering van taken die niet direct bijdragen aan de afwerking of verdere voortgang van de taak van een gebruiker; bijvoorbeeld, de tijd die wordt gebruikt om het gebruik van de computerelementen te verzamelen en/of de doorbelastingen te administreren.

overlay *overlay*

1. Een geheugentoe wijzingstechniek waarin segmenten van computerprogramma's vanuit het hulpgeheugen op het moment dat ze nodig zijn, in het hoofdgeheugen worden geladen. Daarbij worden segmenten die op dat moment niet in gebruik zijn, overschreven.
2. Een segment van een computerprogramma dat in het hulpgeheugen is opgeslagen en op het moment dat dit segment nodig is in het hoofdgeheugen wordt geladen en niet in gebruik zijnde segmenten overschrijft.

overlay (to) *overschrijven*

Een segment van een computerprogramma vanuit het hulpgeheugen in het hoofdgeheugen laden en wel op een zodanige wijze dat andere segmenten van dat programma worden overschreven.

overlay supervisor *overlay-stuurprogramma*

Een routine die de volgorde en de plaats van overlays regelt.

overload (to) *overbelasten*

1. Aan een bewerkingsteken, identificatiesymbool of letterwaarde (literaal) meer dan één betekenis toekennen. Dit is afhankelijk van de gegevenstypen die op een bepaald moment tijdens de uitvoering van een programma met die gegevens zijn geassocieerd.
2. Het gebruiken van een identificatiesymbool voor het verwijzen naar meervoudige items die zich in hetzelfde bereik bevinden.

In de programmeertaal Java kunnen methoden worden overbelast. Niet echter de variabelen of de bewerkingstekens (operatoren).

override (to) *overschrijven*

De term heeft betrekking op het vervangen van de implementatie van een methode in een deelklasse van de klasse waarin de methode oorspronkelijk was gedefinieerd.

P

pack (to) *inpakken*

Gegevens in een compacte vorm in of op een geheugenopslagmedium plaatsen. Hierbij dienen de karakteristieke kenmerken en eigenschappen van de gegevens bekend te zijn zodat die gegevens weer kunnen worden opgehaald en in de oorspronkelijke vorm kunnen worden hersteld.

package *pakket*

1. Een afzonderlijk te compileren softwarecomponent dat uit onderling verwante gegevenstypen, gegevensobjecten en deelprogramma's bestaat.
2. Een groep met gegevens, programma's, assets, objecten enzovoort die van hetzelfde type zijn.

packaging *verpakking*

De term heeft in de software-ontwikkeling betrekking op de toekenning van modules aan segmenten die als duidelijk te onderscheiden en herkennen fysieke eenheden voor een computerverwerking moeten worden beschouwd.

padding *opvulling*

1. De techniek van het opvullen van blokken met een vaste lengte met dummy tekens, woorden of records.
2. Dummy tekens, woorden of records die worden gebruikt om een blok met een vaste lengte op te vullen.

page *pagina*

1. Een gegevenssegment met een vaste lengte of een segment met een vaste lengte dat een computerprogramma bevat en dat als een eenheid in de toekenning van geheugenlocaties wordt behandeld.
2. De term heeft in een virtueel opslagsysteem betrekking op een segment met een vaste lengte of een segment met een vaste lengte dat een computerprogramma bevat, dat een virtueel adres heeft en als een eenheid tussen het hoofd- en hulpgeheugen en vice-versa wordt getransporteerd.

3. Een scherm van een beeldscherm eenheid dat gevuld is met informatie.

page breakage *pagina-overgang*

Een deel van het hoofdgeheugen dat ongebruikt is indien de laatste pagina met gegevens of een computerprogramma niet het gehele blok van de toegewezen geheugenruimte vult.

page frame *paginakader*

Een blok uit het hoofdgeheugen dat de omvang van een pagina heeft en waarin de pagina kan worden opgeslagen.

pager *pagineringsroutine*

Een routine die de uitwisseling van pagina's tussen hoofd- en hulpgeheugen initieert en regelt.

page swapping *paginawisseling*

Het uitwisselen van pagina's tussen hoofd- en hulpgeheugen(s).

page table *paginatablel*

Een tabel die de locatie van pagina's in het geheugen identificeert en belangrijke attributen van die pagina's vermeldt.

page zero *nul-pagina*

In de pagineringsmethode die bij het toekennen van geheugenruimte wordt toegepast, verwijst deze term naar de eerste pagina van een aantal pagina's.

paging *paginering*

Een geheugenallocatietechniek waarin programma's of gegevens in blokken met een vaste lengte, de pagina's, worden opgedeeld. Het hoofdgeheugen is eveneens opgedeeld in blokken met dezelfde vaste lengte, de paginakaders. De pagina's worden in paginakaders opgeslagen. Dit behoeft niet in dezelfde of een logische volgorde te gebeuren. De pagina's worden indien dat nodig of gewenst is, tussen hoofd- en hulpgeheugen uitgewisseld.

parallel *parallel*

1. De term heeft betrekking op de gelijktijdige overdracht, uitwisseling, occurrence, of verwerking van de individuele delen van een geheel, zoals de bits van een teken. Daarbij worden afzonderlijke faciliteiten voor de verschillende delen toegepast.
2. Zie gelijktijdig (concurrent).

parallel construct *parallele constructie*

Een programmaconstructie die uit twee of meer procedures bestaat die gelijktijdig voor kunnen komen.

parallel processing *parallelverwerking*

Het gelijktijdig uitvoeren van twee of meer processen in één enkele verwerkingseenheid.

parallel run *parallelle verwerkingsgang*

Bij de ontwikkeling van systemen verwijst deze term naar een verwerkingsgang van een nieuw of een gewijzigd informatieverwerkend systeem dat met dezelfde brongegevens werkt als die welke bij andere systemen worden gebruikt; de andere systemen kunnen als vergelijkingssystemen worden beschouwd.

parameter *parameter*

1. Een variabele waaraan voor een bepaald doel, een bepaalde applicatie of een bepaalde verwerking, een constante waarde wordt toegekend.
2. Bij het verwerken van tekst is een parameter een keuzemogelijkheid in een menu waarvoor men een waarde kan kiezen of waarvoor het systeem een waarde verschaft wanneer dit menu wordt geïnterpreteerd.
3. Een naam in een procedure die wordt gebruikt om naar een argument te verwijzen dat aan een procedure is doorgegeven.
4. Een constante, variabele of uitdrukking die wordt gebruikt om waarden tussen softwaremodules door te geven.

parameter list *parameterlijst*

Een lijst met waarden die een middel biedt om de adresseerbaarheid van gegevens die in een opgeroepen procedure zijn omschreven, te koppelen met de gegevens in het oproepende programma. Deze lijst bevat parameternamen en de volgorde waarin deze gegevens in het oproepende en het opgeroepen programma met elkaar moeten worden geassocieerd.

parameter word *parameterwoord*

Een woord dat direct of indirect één of meer parameters verschaft of aanwijst.

parent *ouder*

Een knooppunt met één of meer ondergeschikte knooppunten in een boomstructuur.

parent directory *hoofdindex*

De index of de directory die de index bevat van het bestand waar op een bepaald moment mee wordt gewerkt. Deze laatste index is dus een deelindex of subdirectory van de hoofdindex.

Parlog++ *Parlog++*

Een object-georiënteerd systeem dat de object-georiënteerde programmering met de parallelle logische programmering combineert en als zodanig de voordelen van beide paradigma's in één complete en coherente ontwikkelingsomgeving demonstreert.

parse (to) *ontleden*

1. Het analyseren van een zin of een zinsdeel in de constituerende (taal)elementen.
2. Het analyseren van de opdrachtelelementen die gelijktijdig met een opdracht worden ingevoerd en het op basis van deze informatie opstellen van een parameterlijst voor het werkwingsprogramma.
3. Het bepalen van de syntactische structuur van een taal door deze in meer elementaire taalconstructies op te delen en de relaties tussen die constructies vast te stellen. Bijvoorbeeld, blokken ontleden in deelprogramma's, deelprogramma's ontleden in uitdrukkingen, uitdrukkingen ontleden in opdrachten en operanden.

parser *ontledingsprogramma*

1. Een programma of een deel van een programma dat belast is met de analyse en het decoderen van een programma-instructie en het verifiëren van de syntax aan de hand van de gespecificeerde syntax van de programmeertaal waarin de programma-instructie voorkomt.
2. Een functie die geldige zinnen van een taal herkent door de syntax van die taal te controleren. Daarbij wordt van een lexicaal analyseprogramma gebruik gemaakt.

partial correctness *gedeeltelijke correctheid*

Bij het bewijzen van de correctheid van programma's verwijst de term naar de aanname dat de uitvoerasserties van een programma het logische gevolg zijn van de invoerasserties en de daarop volgende verwerkingsstappen van dat programma.

participate (to) *deelnemen*

Een object neemt deel aan een (communicatie- of transactie)verzoek indien één of meer actuele parameters van het verzoek het object identificeert.

partitioning *partitionering*

Decompositie; opdeling; het scheiden van een geheel in haar (samenstellende) delen.

Pascal *Pascal*

Een hogere programmeertaal die wordt gebruikt voor het schrijven van wiskundig georiënteerde, wetenschappelijke en in mindere mate

administratieve programma's en die eveneens goed kan worden toegepast bij de specificatie, de definitie, en de implementatie van softwaresystemen.

pass 1. *verwerkingsgang*; 2. *cyclus*

Eén enkele cyclus in de verwerking van een reeks gegevens waarbij meestal een deel van een compleet proces wordt afgewerkt. Bijvoorbeeld, de verwerkingsgang van een assembleerprogramma door een bronprogramma; een verwerkingsgang van een sorteerprogramma door een reeks gegevens.

pass/fail criteria *slagen/mislukken criteria*

Beslissingsregels die worden gebruikt om te bepalen of een software-item of een softwarefunctie bij een test slaagt of bij een test mislukt.

passivation *passivering*

Het tegenovergesteld van activering. Zie activering (activation).

patch *noodverband*

1. Een modificatie die direct in een objectprogramma wordt aangebracht zonder de noodzaak het bronprogramma opnieuw te moeten assembleren of compileren.
2. Een modificatie die op het laatste moment in een bronprogramma wordt aangebracht.
3. Elke modificatie die in een bron- of objectprogramma wordt aangebracht.

patch (to) *een noodverband aanleggen*

Een modificatie, zoals beschreven in de punten 1 tot en met 3 van de vorige definitie, aanbrengen.

path 1. *pad*; 2. *route*

- a. Een opeenvolgende reeks opdrachten die bij de verwerking van een programma worden uitgevoerd.
- b. Een hiërarchische reeks namen van indexen (directories) en subindexen die de geheugenlocaties van bestanden specificeren.
- c. Een serie opeenvolgende uitvoerbare uitdrukkingen in een programma vanaf het ingangspunt tot aan het punt van uitgang.

path analysis *padanalyse*

Analyse van een computerprogramma ten behoeve van de identificatie van alle mogelijke paden of routes door dat programma. Dit wordt gedaan om de niet-complete paden te detecteren of programmasegmenten te vinden die geen deel van een pad of route uitmaken.

path coverage *paddekking*

Het percentage paden in een component dat door een test of een testserie wordt onderzocht.

path expression *paduitdrukking*

Een logische uitdrukking waarin is aangegeven aan welke invoercondities moet worden voldaan om een specifiek pad in het programma uit te kunnen voeren.

path name *padnaam*

Instructies waarmee wordt aangegeven hoe een bestand kan worden benaderd (gevonden). Een absolute padnaam geeft aan hoe vanaf de wortelindex en het afzoeken van de boomstructuur van de indexen een bestand kan worden gevonden. Een relatieve padnaam geeft aan hoe het bestand vanaf een gespecificeerd punt kan worden gevonden.

pathological coupling *pathologische koppeling*

Een soort koppeling waarin een bepaald softwaremodule de interne implementatie van een ander softwaremodule beïnvloed of daar afhankelijk van is.

path sensitizing *padsensibilisatie*

Het bepalen van een serie invoerwaarden die zodanig zijn gekozen dat bij de uitvoering van de instructies van het component, het component wordt gedwongen een bepaald pad te volgen.

path testing *testen van een pad*

Een techniek waarmee testen of testseries kunnen worden geproduceerd die zodanig zijn ontworpen dat ze alle of geselecteerde paden in een computerprogramma volgen (uitvoeren).

pause (to) *pauzeren*

De uitvoering van een computerprogramma opschorten.

peer *gelijkwaardige eenheid*

Elke functionele eenheid die zich op hetzelfde hiërarchieke of functionele niveau bevindt als een andere entiteit.

perfective maintenance *perfectief onderhoud*

Software-onderhoud dat wordt uitgevoerd om de prestatie, de onderhoudbaarheid, of andere attributen van een computerprogramma te verbeteren of te perfectioneren.

performance *prestatie*

De mate waarin een systeem of een component de gespecificeerde of toegewezen functies zonder beperkingen kan realiseren. Deze term heeft betrekking op snelheid, nauwkeurigheid, geheugengebruik,

energieverbruik en andere karakteristieke kenmerken en eigenschappen van een systeem of component.

performance and code analysis: PCA *prestatie en code-analyse*

Een dynamisch analyseprogramma van de firma DEC dat de gebruiker informatie verschaft over de prestatie van een programma gedurende de uitvoeringstijd en de wijze waarop de code wordt gebruikt.

performance requirement *prestatie-eis*

Een eis die bepaalde condities oplegt aan een functionele eis; bijvoorbeeld, een eis die de snelheid, de nauwkeurigheid of het geheugengebruik van een functie specificiert.

performance specification *prestatiespecificatie*

Een document waarin de prestatiekenmerken van een systeem of component zijn vastgelegd. Deze karakteristieken omvatten in ieder geval snelheid, nauwkeurigheid en geheugengebruik en soms ook het energieverbruik. Een dergelijke specificatie maakt meestal deel uit van de eisenspecificatie.

performance testing *testen van prestaties*

Testen die worden uitgevoerd om na te gaan in hoeverre een systeem of een component zich conformeert aan gespecificeerde eisen die ten aanzien van de prestaties gelden.

persistent object *hardnekkig object*

Een object dat het proces of de draadmodule (thread) kan overleven waarin, waarmee, of met behulp waarvan het object is ontstaan. Een hardnekkig object blijft bestaan totdat het op expliciete wijze wordt afgevoerd of verwijderd.

Petri net *Petri-net*

Een grafische voorstelling van gelijktijdig werkende systemen. De beschrijving wordt uitgevoerd met behulp van staffels (tokens), plaatsen (places) en overgangen (transition bars). Deze geven de dynamische eigenschappen van het netwerk aan.

PHIGS extension to X: PEX *PHIGS-uitbreiding van X*

Een uitbreiding van het X Window systeem dat drie-dimensionale grafische functies aan het systeem toevoegt.

physical configuration audit: PCA *fysieke configuratie-audit*

Een controle of audit waarin wordt geverifieerd in hoeverre een gerealiseerd of geconstrueerd configuratie-item zich conformeert aan de technische documentatie waarin het is gedefinieerd.

physical requirement *fysieke eis*

Een eis waarin een fysiek kenmerk of attribuut, dan wel een fysieke eigenschap van een systeem of een component is gespecificeerd. Bijvoorbeeld: materiaal, omvang of grootte, vorm, gewicht.

picture specification *beeldspecificatie*

Een teken-voor-teken-beschrijving van de samenstelling en de kenmerken van binaire beeldgegevens, decimale beeldgegevens en tekenrij-beeldgegevens.

pipe *pijp*

Het | teken dat wordt gebruikt om de uitvoer van een bepaalde instructie naar de invoer van een andere instructie te dirigeren.

pipeline *pijplijn*

Een ontwerptechniek voor hardware of software waarin de resultaten van een bepaald proces de invoer voor een ander proces zijn, de resultaten van een tweede proces de invoer voor een derde proces vormen enzovoort. Dit geschiedt vaak met een bepaalde vorm van gelijktijdigheid binnen één enkele cyclustijd.

plan standard *planstandaard*

Een standaard waarin de belangrijkste punten van een plan voor het realiseren van gedefinieerde doelstellingen of werkzaamheden die met behulp van gespecificeerde (hulp)middelen moeten worden uitgevoerd, zijn beschreven.

pointer *1. wijzer; 2. pointer*

- a. Een identificatiesymbool dat de positie van een gegeven aangeeft.
- b. Een gegevensitem dat de locatie van een ander gegevensitem specificeert; bijvoorbeeld een gegevensitem dat het adres van het eerstvolgende personeelsrecord dat moet worden verwerkt, specificeert.
- c. Een functionele eenheid die met de hand kan worden bediend en die wordt gebruikt om een adresseerbaar punt in een grafische toepassing te specificeren.

pointer variable *wijzervariabele*

Een variabele waarvan de waarde de identiteit van een absolute positie in het primaire geheugen aangeeft.

point of invocation *aanroeppunt*

Het punt in het oproepende deel van een programma of blok waar de procedureverwijzing naar de opgeroepen procedure verschijnt.

policy independence *onafhankelijkheidskenmerk*

Een eigenschap van het X Window systeem waarin vensters op de wijze waarop de software-ontwikkelaars dat aangeven, kunnen verschijnen

en de bij het venster behorende functie uit kunnen voeren. Het idee van deze eigenschap is het tegenovergestelde van het idee dat, indien alle vensters op het beeldscherm gelijksoortige verschijningsvormen hebben en gelijksoortige functies uitvoeren, dit voor de gebruiker gemakkelijker zou zijn.

Polish notation *prefixnotatie*

Een methode om wiskundige uitdrukkingen te vormen waarbij elk bewerkingsteken voorafgaat aan de bijbehorende opdrachtelelementen en de bewerking aangeeft die op deze opdrachtelelementen of de tussenresultaten, die er op volgen, moet worden verricht.

polymorphism *polymorfisme*

1. De eigenschap dat een bewerking een afwijkend gedrag voor verschillende objecten kan hebben.
2. In de object-georiënteerde programmering wordt de term gebruikt om variabelen te beschrijven die tijdens de uitvoeringstijd naar objecten van verschillende klassen kunnen verwijzen. Zo kan de variabele 'mijnSysteem' naar de objecten van klassen 'C++' en 'Unix' verwijzen.

portability *portabiliteit*

1. Het gemak waarmee een systeem of component van de ene hardware- of software-omgeving naar een andere kan worden getransporteerd.
2. De mate waarin een module die, dan wel asset of programma dat oorspronkelijk op een bepaalde computer of onder een bepaald besturingssysteem was geïmplementeerd, op een andere computer of onder een ander besturingssysteem kan worden verwerkt. Hoe groter de portabiliteit van een module, asset of programma, hoe groter het potentiële hergebruik daarvan is.

portability testing *testen van de portabiliteit*

Testen die zich richten op de mate en wijze waarop software tussen gespecificeerde apparatuur- en/of softwareplatformen kan worden overgedragen of uitgewisseld.

portable common tool environment: PCTE *omgeving voor overdraagbare gemeenschappelijke gereedschappen*

Een standaard-raamwerk voor softwaregereedschappen dat door de ECMA is opgesteld en in een ESPRIT-programma is ontwikkeld. PCTE is gebaseerd op een object-beheerssysteem (OMS) dat van entiteit-relaties gebruik maakt. Het definieert de manier waarop de gereedschappen toegang tot dit beheerssysteem kunnen krijgen en de wijze waarop ze dit systeem kunnen gebruiken.

portable operating system interface for different computers: POSIX
POSIX

1. POSIX beschrijft een familie definities waarin is vastgelegd hoe delen van computersystemen met elkaar en met gebruikers werken. POSIX lijkt op Unix maar is niet het product van één specifieke leverancier. Er bestaan thans meer dan een dozijn leden van de POSIX-familie; de meest bekende daarvan is 1003.2 (beter bekend als 3 punt 2) en de daaraan verwante gebruikersgerichte portabiliteitsuitbreidingen (UPE of User Portable Extension). Deze laatste twee definiëren de wijze waarop de instructies en de shells werken. Een systeem dat 1003.2 en UPE compileert lijkt zeer sterk op het Unix-besturingssysteem.
2. Een standaard waarin de taalinterface tussen het Unix-besturingssysteem en applicatieprogramma's met behulp van een minimaal aantal ondersteunende functies is gedefinieerd. Een virtuele adresruimte bevat één of meer draadmodules (threads).

portable software *overdraagbare programmatuur*

Programma's en softwaresystemen die op verschillende computers met verschillende besturingssystemen kunnen functioneren. Unix is zelf een portabel (overdraagbaar) besturingssysteem omdat het op een zeer groot aantal computers van verschillende makelij kan worden geïmplementeerd en gebruikt.

port-to-port time *tijdsverloop tussen in- en uitvoerpoorten*

De verstreken tijd tussen het aanbieden van een stimulus aan een invoereenheid en het verschijnen of optreden van de responsie op die stimulus in de uitvoereenheid.

postmortem dump *postmortem dump*

Een dump die bij een abnormale afbreking of beëindiging van een computerprogramma wordt geproduceerd.

postprocessor *1. postprocessor; 2. naverwerkingsprogramma*

Een computerprogramma dat of een routine die een zekere afsluitende verwerkingsstap na beëindiging van het hoofdproces uitvoert; bijvoorbeeld, een routine die gegevens voor de uitvoer in een ander formaat vertaalt.

Postscript: PS *Postscript*

Een taal van de firma Adobe Systems Inc. waarmee de opmaak van pagina's en de verschijningsvorm van tekens, tekst en beelden kan worden gedefinieerd. Een programma dat in Postscript is geschreven kan de documentdefinitie van een pagina-opmaakstelsel communiceren met een afdruksysteem (printstelsel) op een wijze die onafhankelijk is van de gebruikte apparatuur. Er bestaan thans veel printers die Postscript direct interpreteren.

practical extraction and report language: PERL *praktische extractie en rapportage taal*

Een geïnterpreteerde scriptietaal voor het afzoeken van tekstbestanden, het extraheren van informatie, en het afdrukken van rapporten.

practices *praktijken*

Eisen die zijn opgesteld en een beschrijving geven van een gedisciplineerde uniforme benadering in de ontwikkeling van software.

precision *precisie*

De mate van de nauwkeurigheid waarmee een bepaalde grootte wordt gedefinieerd of de mate van het onderscheid dat tussen bepaalde grootheden kan worden gemaakt; bijvoorbeeld, een precisie van 2 decimale plaatsen ten opzichte van een precisie van 5 decimale plaatsen.

precompiler *voorcompiler*

Een computerprogramma dat of routine die broncode verwerkt en een equivalente code genereert die door een compiler kan worden verwerkt. Bijvoorbeeld, een routine die gestructureerd Fortran converteert naar standaard Fortran.

precondition *pre-conditie*

Omgevings- en toestandscondities waaraan moet worden voldaan voordat de functies van een component dat voorzien wordt van één of meer gegeven invoerwaarden, kunnen worden verwerkt.

predicate *predikaat*

1. Een logische uitdrukking die een logische waarde 'waar' of 'onwaar' oplevert indien de uitdrukking wordt geëvalueerd.
2. Een taalelement waaraan een bepaalde betekenis en bepaalde argumenten kunnen worden toegekend. Het vormt een uitbreiding op de taalelementen 'variabele' en 'constante'.
3. Benaming; attribuut.

predicate calculus *predikaten-calculus*

Een notatie voor de representatie van logische uitdrukkingen die buiten het gebied van de propositie-calculus liggen.

predicate logic *predikaten-logica*

Een uitbreiding van de propositie-logica waarin van predikaten gebruik wordt gemaakt.

predicted outcome *voorspeld resultaat*

Het gedrag dat van een gespecificeerd object onder gespecificeerde condities kan worden voorspeld.

prefix *prefix*

1. Een label of een tussen haakjes geplaatste lijst met één of meer conditienamen die door middel van een dubbele punt aan het begin van een opdracht met elkaar zijn verbonden (dit geldt onder andere voor de programmeertaal PL/1).
2. Een code aan het begin van een woord of een bericht.

prefix notation *prefixnotatie*

Een methode om wiskundige uitdrukkingen te vormen waarbij elk bewerkingsteken voorafgaat aan de bijbehorende opdrachtelelementen en de bewerking aangeeft die op deze opdrachtelelementen of de tussenresultaten, die er op volgen, moet worden verricht.

prefix operator *prefixbewerkingsteken*

Een bewerkingsteken dat voorafgaat aan een opdrachtelelement en dat uitsluitend betrekking heeft op dat opdrachtelelement.

preliminary design *voorlopig ontwerp*

1. Het proces waarin ontwerpalternatieven worden geanalyseerd en de architectuur, componenten, interfaces en schattingen van verwerkingstijd en programmagrootte van een systeem of een component worden gedefinieerd.
2. Het resultaat van het proces dat in punt 1 van deze definitie is beschreven.

preliminary design review: PRD *beoordeling van een voorlopig ontwerp*

1. Een beoordeling die wordt uitgevoerd om (a) de voortgang, de technische geschiktheid, en de graad van het risico van een geselecteerde ontwerpbenadering voor één of meer configuratie-items te evalueren; (b) de compatibiliteit van het ontwerp met de eisen die voor het configuratie-item zijn gespecificeerd, te bepalen; (c) het technische risico dat met de gekozen fabricagemethoden en -processen samenhangt, in te schatten; (d) de aanwezigheid en compatibiliteit van de fysieke en functionele interfaces tussen de configuratie-items en andere items betreffende de apparatuur, faciliteiten, hulpmiddelen, software en personeel vast te leggen; en (e) voor zover van toepassing, de voorlopige operationele en ondersteunende documentatie te evalueren.
2. Een beoordeling zoals beschreven in punt 1 van deze definitie die van toepassing is op elk willekeurig hardware- of software-recomponent.

preprocessor *1. preprocessor; 2. voorverwerkingsprogramma*

Een computerprogramma dat of routine die een bepaalde verwerkingsstap voorafgaande aan het hoofdproces uitvoert; bijvoorbeeld, een precompiler of andere routine die de code of de gegevens voor verwerking omzet in een ander formaat.

preset parameter *vooraf ingestelde parameter*

Een parameter die wordt vastgesteld bij het vervaardigen van het programma, bijvoorbeeld wanneer er een stroomschema van wordt gemaakt of wanneer het wordt gecodeerd of gecompileerd.

prestore (to) *vooraf opslaan*

Gegevens die een computerprogramma of routine nodig heeft, opslaan voordat het programma of de routine wordt uitgevoerd.

prettyprinting *fraai afdrukken*

Het gebruik van indentaties, blanco alinea's en andere visuele middelen om de logische structuur van een programma goed tot uitdrukking te laten komen.

preventive maintenance *preventief onderhoud*

Onderhoud dat wordt uitgevoerd met het doel problemen te voorkomen voordat deze optreden.

primary device *primaire eenheid*

1. Het eindstation of de terminal waarmee het programma communiceert.
2. Een eenheid waaraan een andere eenheid is toegewezen.
3. Een terminal die wordt gebruikt voor de communicatie met een andere terminal.

primary program *primair programma*

Het eerste programma dat onder een taak wordt uitgevoerd. Een primair programma heeft over het algemeen één enkel ingangspunt en moet over een programma-aanhef beschikken.

primary task *primaire taak*

1. De eerste taak die wordt geactiveerd wanneer een reeks taken met de uitvoering begint. De primaire taak wordt door een supervisieprogramma gestart.
2. Een taak die van een primaire eenheid gebruik kan maken.

primitive data type *primitief gegevenstype*

Gegevenstype waarvoor geldt dat de bij dat type behorende gegevens uit één enkel niet-opdeelbaar gegevensitem bestaan.

principal interface *principale interface*

De interface die alle verzoeken beschrijft waarin een object van betekenis is.

priority *prioriteit*

Het importantieniveau dat aan een item, taak, proces, programma enzovoort is toegekend.

priority interrupt *prioriteitsonderbreking*

Een onderbreking die wordt geforceerd om een proces dat een hogere prioriteit heeft dan het proces dat op dat moment wordt uitgevoerd, voorrang te geven.

private type *besloten type*

Een gegevenstype waarvoor geldt dat van de bij dat type behorende gegevens waarden zijn gedefinieerd die niet aan de gebruiker ervan bekend zijn of worden gemaakt.

privileged instruction *bevoorrechte opdracht*

Een computerinstructie die slechts door een supervisieprogramma kan worden uitgevoerd.

problem oriented language *probleemgerichte taal*

Een programmeertaal die geconcipieerd en ontwikkeld is voor het oplossen van een bepaalde klasse van problemen. Voorbeelden van dergelijke talen zijn lijstverwerkingstalen, simulatietalen, en hardware-definitietalen.

problem state *probleemtoestand*

De term heeft bij het werken met een computersysteem betrekking op een toestand waarin programma's anders dan een supervisieprogramma de verwerking kunnen besturen.

problem statement language/problem statement analyzer: PSL/PSA
PSL/PSA

Een CASE-systeem waarmee de ontwikkeling en analyse van een pakket eisen van een te ontwikkelen softwaresysteem wordt ondersteund en assistentie gedurende de ontwerpfase van dat systeem wordt geboden.

procedural cohesion *procedurele cohesie*

Een soort cohesie waarbij de taken die door een softwaremodule worden uitgevoerd, allen bijdragen tot het afwerken van een bepaalde procedure van een programma, zoals een iteratie of een beslissingsproces.

procedural language *procedurele taal*

Een programmeertaal waarmee een gebruiker de oplossing van een probleem als een specifieke serie instructies die in een gegeven volgorde moeten worden afgewerkt, of als een expliciet algoritme uit kan drukken. Voorbeelden van procedurele talen zijn Fortran, Algol, Cobol, Pascal, Ada, PL/1, C, en Java.

procedural statement *procedurele opdracht*

In een programmeertaal is dit een opdracht die een specificatie van een bewerking geeft en die de identiteit van de eventueel aanwezige opdrachtelelementen bepaalt.

procedure *procedure*

1. De gedragslijn die of het actieplan dat wordt gevolgd om een probleem op te lossen.
2. Een op schrift gestelde beschrijving van de gedragslijn of het actieplan dat in punt 1 van deze definitie is genoemd.
3. Een opeenvolgende serie opdrachten die kan worden gebruikt op één of meer punten in één of meer programma's en die gewoonlijk één of meer invoerparameters heeft en één of meer uitvoerparameters oplevert.
4. De stappen die worden ondernomen bij het stand brengen van protocols.
5. Een serie opdrachten in een bepaald programma die van een naam is voorzien en vooraf wordt gegaan door een speciale 'procedure'-opdracht en met een END-opdracht wordt afgesloten en die een specifieke actie uitvoert.

procedure call *procedure-oproep*

Bij programmeertalen verwijst de term naar een taalconstructie om een procedure op te roepen en de werking ervan te starten. Een procedure-oproep omvat gewoonlijk een ingangsnaam en mogelijke actuele parameters.

procedure division *procedure-afdeling*

Een bestanddeel van een programma dat in Cobol is geschreven. De procedure-afdeling bevat de instructies die nodig zijn om een probleem op te lossen. Deze afdeling kan onvoorwaardelijke opdrachten, conditionele opdrachten, paragrafen, procedures en secties bevatten.

procedure name *procedurenaam*

Een woord dat voorafgaat aan en de identiteit bepaalt van een procedure die door de software-ontwikkelaar wordt gebruikt om de besturing van het ene punt naar een ander punt van het programma over te dragen.

procedure-oriented language *procedure-georiënteerde taal*

Een programmeertaal waarmee de gebruiker de oplossing van een probleem als een expliciet algoritme uit kan drukken.

procedure reference *procedureverwijzing*

Een ingangsconstante of variabele of een ingebouwde functienaam in een programmeertaal die gevolgd wordt door een lijst met argumenten. Deze verwijzing kan voorkomen in de opdracht die de procedure oproept.

procedure statement *procedure-opdracht*

Een opdracht die wordt gebruikt om een naam aan een procedure toe te kennen.

procedure step *procedurestap*

Een eenheid werk die verbonden is met één verwerkingsprogramma en de bijbehorende gegevens. Deze is in een gecatalogiseerde procedure geplaatst. Een dergelijke procedure bestaat uit één of meer procedurestappen.

process *proces*

1. Een systematische reeks stappen of bewerkingen die gericht zijn op het produceren van een bepaald resultaat.
2. Een unieke en eindige reeks gebeurtenissen in een computersysteem. Deze reeks wordt bepaald door het doel of het resultaat dat onder gegeven omstandigheden moet worden bereikt.
3. Een functie die wordt uitgevoerd of die wacht om te worden uitgevoerd.
4. Een operationele eenheid die door een roosterfunctie van het besturingssysteem wordt bestuurd.

process (to) *verwerken*

Het verrichten van bewerkingen op gegevens in een proces.

processing *verwerking*

Het verrichten van (logische) bewerkingen en berekeningen op gegevens. Dit betekent ook het tijdelijk opslaan van gegevens in het geheugen van de verwerkingseenheid tijdens het proces waarin deze gegevens worden verwerkt of bewerkt.

process management *procesbeheer*

Het geven van richting aan, de besturing, en de coördinatie of de werkzaamheden die noodzakelijk zijn, respectievelijk worden uitgevoerd, in het ontwikkelen van een product of het verlenen van diensten.

processor *1. verwerkend systeem; 2. verwerkingseenheid*

- a. Bij een computer verwijst de term naar een functionele eenheid die instructies leest, interpreteert en uitvoert.
- b. Een functionele eenheid die een deel vormt van een andere eenheid. Een voorbeeld is de intelligente terminal die instructies interpreteert en uitvoert.

process standard *processtandaard*

Een standaard die betrekking heeft op de verzameling acties of bewerkingen die worden gebruikt in het maken of realiseren van een product.

product analysis *productanalyse*

Het proces waarin een product met behulp van handmatige of geautomatiseerde middelen wordt geëvalueerd om te bepalen welke specifieke kenmerken of eigenschappen het product bezit.

product baseline *product-voetlijn*

De initieel goedgekeurde technische documentatie in het configuratiebeheer die een configuratie-item tijdens de productie, de operationele toestand, het onderhoud en de logistieke ondersteuning van de levenscyclus van het item, definieert. (Voor software omvat deze technische documentatie eveneens de afdruk van de broncode).

product configuration identification *productconfiguratie-identificatie*

De goedgekeurde of conditioneel goedgekeurde technische documentatie die een configuratie-item tijdens de productie, de operationele toestand, het onderhoud en de logistieke ondersteuning van de levenscyclus van het item, definieert. Deze documentatie bevat alle voorschriften en beschrijvingen van de noodzakelijk geachte karakteristieke kenmerken en eigenschappen van het configuratie-item betreffende de fysieke aspecten; de aspecten ten aanzien van de vorm, geschiktheid en functie; de geselecteerde functionele kenmerken die voor de productie-acceptatietesten gelden; en de productie-acceptatietesten zelf.

product engineering *productontwikkeling*

Het technische proces waarin een product wordt gedefinieerd, ontworpen, en geconstrueerd of samengesteld.

production library *productiebibliotheek*

Een softwarebibliotheek die software bevat die voor het momentane operationele gebruik is goedgekeurd.

product management *productbeheer*

De definitie, coördinatie, en besturing van de karakteristieken van een product tijdens de ontwikkelingscyclus van dat product. Een voorbeeld van productbeheer is het configuratiebeheer.

product specification *productspecificatie*

1. Een document dat het ontwerp specificiert.
2. Een document dat de karakteristieken van een gepland of bestaand product beschrijft en dat potentiële klanten voor een verdere beoordeling kunnen gebruiken.

product standard *productstandaard*

Een standaard waarin de volledigheid en acceptabiliteit van items die tijdens het ontwikkelingsproces van software worden gebruikt of

geproduceerd, is gedefinieerd. De standaard kan een formele of informele beschrijving van die definitie zijn.

product support *productondersteuning*

Het verstrekken van informatie, assistentie, en training voor het installeren en operationeel maken en houden van software in de daarvoor bestemde omgeving en het distribueren van verbeterde functies of versies van de software aan gebruikers.

professional standard *professionele standaard*

Een standaard die een bepaald beroep als een discipline identificeert en het van andere beroepen onderscheidt.

program *programma*

Een reeks instructies die een computer kan interpreteren en uitvoeren en die vanaf een informatiedrager wordt ingelezen of door een gebruiker of programmeur wordt ingetoetst of ingesproken.

program (to) *programmeren*

Het ontwerpen, schrijven en testen van programma's. Hiertoe behoort soms ook het analyseren van een probleem dat met behulp van een programma kan worden verwerkt.

program block *programmablok*

Bij probleemgerichte programmeertalen is dit een deel van een programma waarin aan elkaar verwante opdrachten worden gegroepeerd, de grenzen van modules worden vastgesteld, de toepasbaarheid van labels worden bepaald, of delen van het programma voor andere doeleinden in segmenten worden ondergebracht.

program data *programmagegevens*

De gegevens die op een programma betrekking hebben.

program design language: PDL *programma-ontwerptaal*

Een specificatietaal met speciale constructies die soms is voorzien van verificatieprotocollen en die wordt gebruikt voor de ontwikkeling, de analyse, en de documentatie van het ontwerp van een programma.

program header *programma-aanhef*

Een blok met gegevens dat aan het geladen programma voorafgaat en dat de bij dat programma behorende functies, kenmerken of attributen beschrijft.

program identifier *programma-identificatiesymbool*

Een getal dat of een naam die door een programmeur aan een programma wordt toegekend en die door een besturingseenheid wordt gebruikt om de identiteit van het programma te bepalen.

program instruction *1. programma-instructie; 2. programma-opdracht*

- a. Een instructiecode die, indien ingelezen, er zorg voor draagt dat één of meer functies automatisch in werking treden.
- b. Besturingscode die in een informatiedrager of in een verwerkingseenheid is opgeslagen en die ervoor zorgt dat de machine op een vooraf bepaalde wijze reageert.
- c. Een computerinstructie of computeropdracht in een bronprogramma. Opmerking: Er bestaat een onderscheid tussen een programma-instructie en een computerinstructie. De laatste is het resultaat van een assembleerproces, een compilatie, of enig ander interpretatieproces.

program listing *programma-uitdraai*

Een afdruk of een andere mens-leesbare afbeelding van de bron- en (soms) objectinstructies die een computerprogramma constitueren.

programmable breakpoint *programmeerbaar breekpunt*

Een breekpunt dat automatisch een eerder gespecificeerd en geïnitieerd foutopsporingsproces activeert.

programmer manual *programmeringshandleiding*

Een document dat informatie bevat die nodig is om software voor een bepaald computersysteem te ontwikkelen of te modificeren. Typische onderwerpen die in een dergelijke handleiding worden beschreven zijn: de configuratie van de apparatuur, operationele karakteristieken, programmeringsmethoden en -technieken, invoer/uitvoerfuncties en/of -mogelijkheden, en compilatie- of assembleerfuncties van het systeem.

programmers hierarchical interactive graphics system: PHIGS
hiërarchisch interactief grafisch systeem voor programmeurs

Een ANSI/ISO-standaard waarin de drie-dimensionale gebruikersinterface voor grafische toepassingen is gedefinieerd.

programming in logic: PROLOG *PROLOG*

Een kunstmatige taal waarmee logische problemen kunnen worden opgelost en waarin de uitdrukkingen voor het merendeel uit logische verklaringen en samengestelde logische uitspraken bestaan. Zie propositionele logica.

programming language *programmeertaal*

1. Een taal die wordt gebruikt voor het schrijven van programma's.

2. De woorden, mnemonische uitdrukkingen en symbolen die, in combinatie met bepaalde regels, bij het samenstellen van een programma worden gebruikt.

programming language one: PL/1 *PL/1*

Een hogere programmeertaal die is ontworpen om te worden gebruikt bij de systeemprogrammering en bij een groot aantal administratieve en technisch-wetenschappelijke toepassingen. PL/1 is zeer geschikt voor het werken met verschillende gegevensstructuren.

programming module *programmamodule*

Een op zich zelf staande en herkenbare reeks instructies die meestal als een eenheid wordt behandeld door een assembleer-, een compileer-, of een montageprogramma.

programming statement *programma-opdracht*

Een opdracht uit een reeks symbolische opdrachten die wordt gebruikt om computerprogramma's te schrijven.

programming support environment: PSE *omgeving voor de programmeringsondersteuning*

1. Een geïntegreerde verzameling softwaregereedschappen die via een commandotaal kunnen worden opgeroepen en de programmeur programma-ondersteuningsfaciliteiten tijdens de gehele levenscyclus van de software verschaft. Een dergelijke omgeving bevat gereedschappen voor de specificatie, het ontwerp, de opmaak, de compilatie, het laden, het testen, het configuratiebeheer, en het projectbeheer. Soms wordt een dergelijke omgeving, de omgeving voor een geïntegreerde programmeringsondersteuning (integrated programming support environment: IPSE) genoemd.
2. Een serie gereedschappen die in een bepaalde omgeving tezamen met een besturingssysteem, wordt gebruikt om projecten te beheren en te besturen.

programming system *programmeringssysteem*

Eén of meer programmeertalen en andere noodzakelijke programmatuur zoals opmaakprogramma's, compilers, laadprogramma's, montageprogramma's enzovoort om deze in combinatie met de computer te kunnen gebruiken.

program loop *programmalus*

Een reeks opdrachten die zolang wordt herhaald tot aan een gestelde voorwaarde of conditie is voldaan.

program mutation *programmamutatie*

1. Een computerprogramma dat opzettelijk ten opzichte van de oorspronkelijke of bedoelde versie is gewijzigd met het doel

de efficiëntie van testgevallen, die die wijzigingen op moeten sporen, te evalueren.

2. Het proces als bedoeld in punt 1 van deze definitie.

program network chart *programmanetwerkschema*

Een schema of diagram dat de relatie tussen twee of meer computer-programma's weergeeft of uitbeeldt.

program parameter *programmaparameter*

Een parameter waaraan een waarde tijdens het oproepen van een programma moet worden toegekend.

program/preliminary design language: PDL *1. programma-ontwikkelingstaal; 2. voorlopig-ontwerptaal*

Een taal waarmee een ontwerp kan worden beschreven en waarin een combinatie van gestructureerde programmaconstructies en natuurlijke Engelse tekst wordt gebruikt.

program sensitive fault *programmagevoelige fout*

Een fout die een storing veroorzaakt indien een specifieke reeks stappen in een programma wordt uitgevoerd.

program status word: PSW *programmastatus-woord*

1. Een computerwoord dat informatie bevat waarmee de huidige status van een computerprogramma wordt gespecificeerd. De informatie kan foutindicatoren, het adres van de volgende instructie die moet worden uitgevoerd, momenteel actief zijnde onderbrekingen enzovoort bevatten.
2. Een register voor algemeen gebruik dat het programmastatuswoord, zoals beschreven in punt 1 van deze definitie, bevat.

program step *programmastap*

Een gedeelte van een programma dat gewoonlijk uit één enkele instructie bestaat.

program synthesis *programmasyntese*

Het gebruik van softwaregereedschappen die behulpzaam kunnen zijn bij de transformatie van de specificatie van een programma naar een programma dat die specificatie kan doen realiseren.

program unit *programma-eenheid*

Een hoofdprogramma of een deelprogramma.

program variable *programmavariabele*

Een veranderbare waarde die alleen binnen programma's kan bestaan. De waarde kan niet worden verkregen of gebruikt indien het

programma dat deze waarde bevat of hanteert, niet meer wordt aangeroepen.

progressive test *progressie-test*

Een test die nieuwe functies op hun juistheid controleert nadat de reeds bestaande functies eerder aan een regressie-test zijn onderworpen.

project assurance *projectgarantie*

Het proces waarin het ondersteuningssysteem (technieken, interne standaarden, metingen, gereedschappen, training, adviezen aan het projectteam, bewaken van de kwaliteit van het project, volgen van de projectvoortgang enzovoort) van een project wordt gespecificeerd.

project file *projectfile*

Een centrale vergaarbak van materiaal dat belangrijk is voor een project. De inhoud ervan bevat memo's, plannen, technische rapporten, en andere daarmee verwante items.

project management *projectbeheer*

Het proces van het plannen, het organiseren, het bemannen, het sturen, en het controleren van de ontwikkeling van een systeem. Deze activiteiten kunnen met behulp van softwaregereedschappen worden ondersteund.

project plan *projectplan*

Een document dat de technische en managementbenadering die door een project moet worden gevolgd, beschrijft. Het plan beschrijft het werk dat moet worden uitgevoerd, de benodigde (hulp)middelen, de te gebruiken methoden, de te volgen procedures, het tijdschema en rooster, de deelprojecten die op het kritieke pad liggen, alsmede de wijze waarop het project zal worden georganiseerd.

prolog breakpoint *proloog-breekpunt*

Een breekpunt dat aan de ingang van een programma of routine wordt geïnitieerd.

prompt 1. *prompt*; 2. *oproepteken*

Een symbool of bericht dat door een computersysteem wordt afgebeeld (op een beeldscherm wordt geprojecteerd) en dat de gebruiker verzoekt het systeem van invoergegevens te voorzien.

prompt (to) *oproepen*

Een symbool of bericht, zoals beschreven in de vorige definitie, afbeelden.

proof of correctness *correctheidsbewijs*

1. Een formele techniek die wordt gebruikt om wiskundig te bewijzen dat een computerprogramma aan de gespecificeerde eisen voldoet.
2. Een bewijs dat het gevolg is van het toepassen van de techniek zoals beschreven in het vorige punt van deze definitie.

property *eigenschap*

Een attribuut waarvan de waarde kan worden gewijzigd.

propositional calculus *propositionele calculus*

Het deel van de formele logica dat met uitspraken rekt ('waar', 'onwaar').

propositional logic *propositionele logica*

Logica waarop de Booleaanse algebra is gebaseerd en die wetten bestudeert die voor samengestelde uitspraken gelden. Die uitspraken zijn verbonden door bewerkingstekens (operatoren) zoals 'En', 'Of', 'Niet-En', 'Niet' enzovoort.

protection exception *beschermingsexceptie*

Een exceptie (uitzonderingstoestand) die optreedt indien een programma in het beschermde (afgeschermd) gebied van het geheugen tracht te schrijven.

protocol *protocol*

1. Een overeenkomst betreffende de wijze waarop gegevens worden overgedragen. Dat geldt speciaal indien gegevens via een netwerk worden getransporteerd. Laag-niveau protocollen definiëren de elektrische en fysieke standaarden die in acht dienen te worden genomen en behandelen de overdracht en de detectie en correctie van fouten in de gegevensstroom (bit-stroom). Hoog-niveau protocollen behandelen de wijze waarop gegevens worden geformatteerd, de vorm van boodschappen en berichten, de dialoog tussen computer en terminal, de dialoog tussen mens en machine, de opmaak en/of definitie van bestanden enzovoort.
2. De verzameling regels en conventies in een (communicatie)systeem die de interactie tussen processen, eenheden en andere systeemcomponenten, alsmede de overdracht en de ontvangst van gegevens regelt.

prototype *prototype*

Een voorlopig type, exemplaar (kopie) of voorlopige vorm van een systeem dat model staat voor latere stadia of voor de uiteindelijke en complete versie van het systeem.

prototyping *vervaardigen van prototypen*

1. Het vervaardigen van een product waarin niet alle details en/of functies zijn verwerkt of in zijn opgenomen. Een prototype wordt over het algemeen gebruikt om de specificaties en de ontwikkeling van het uiteindelijke product (hardware en/of software) respectievelijk te ondersteunen en te bespoedigen.
2. De compositie van een model en de simulatie van alle aspecten van een product. CASE-gereedschappen geven een belangrijke ondersteuning bij het vervaardigen van prototypen. Een software-prototype wordt vooral gebruikt voor de terugkoppeling naar gebruikers, het bepalen van de haalbaarheid of het onderzoeken van tijdsgevoelige aspecten of andere zaken die van belang zijn voor het ontwikkelingsproces.

pseudo code *pseudocode*

Een notatie die op een programmeertaal lijkt maar niet bedoeld is om te worden gecompileerd. In een dergelijke code worden over het algemeen de structuur van een programmeertaal gecombineerd met een informele en in de natuurlijke taal opgestelde beschrijving van de uit te voeren berekeningen of bewerkingen. Een voorbeeld van deze coderingsvorm is de volgende serie instructies:

```
IF de gegevens sneller arriveren dan verwacht
THEN weiger iedere derde serie invoergegevens
ELSE verwerk alle ontvangen gegevens
ENDIF
```

pseudo instruction *pseudo-instructie*

1. Een instructie in de brontaal die informatie of besturingsopdrachten voor het assembleerprogramma of de compiler bevat. Deze instructie wordt niet in een instructie van de doeltaal vertaald. Dit kan, bijvoorbeeld, een instructie zijn die de gewenste opmaak van een afdruk van de broncode specificeert.
2. Een door de gebruiker of programmeur gebruikte of gedefinieerde instructie zoals een macro. Deze behoort niet tot de standaard-instructieset.

pseudo variable *pseudo-variabele*

Eén van de ingebouwde functienamen die kan worden gebruikt om een doelvariabele te specificeren.

push-down stack *omgekeerde lijst*

Een lijst die op een zodanige wijze wordt samengesteld en bijgehouden dat het eerstvolgende gegeven of onderwerp dat zal worden opgezocht (en eventueel verwijderd), het gegeven of onderwerp is dat het laatst aan de lijst werd toegevoegd.

push-down storage *LIFO-geheugen*

Een geheugeneenheid die de gegevens op een zodanige wijze behandelt dat het eerstvolgende gegeven dat moet worden opgezocht, het gegeven is dat het laatst werd opgeslagen en dat zich nog steeds in deze geheugeneenheid bevindt.

push-up stack *directe lijst*

Een lijst die op een zodanige wijze wordt samengesteld en bijgehouden dat het eerstvolgende gegeven of onderwerp dat moet worden opgezocht, het gegeven of onderwerp is dat het eerst werd opgeslagen en zich nog steeds in deze lijst bevindt.

push-up storage *FIFO-geheugen*

Een geheugeneenheid die de gegevens op een zodanige wijze behandelt dat het eerstvolgende gegeven dat moet worden opgezocht, het gegeven is dat het eerst werd opgeslagen en dat zich nog steeds in deze geheugeneenheid bevindt.

Q

qualification testing *kwalificatietesten*

Testen die worden uitgevoerd om te bepalen of een systeem of component geschikt is voor normaal operationeel gebruik.

quality *kwaliteit*

1. De mate waarin een systeem, component of proces voldoet aan de gespecificeerde eisen.
2. De mate waarin een systeem, component, of proces voldoet aan de verwachtingen en/of behoeften van klanten of gebruikers.

quality assurance *kwaliteitsborging*

1. Een gepland en systematisch patroon van alle acties die noodzakelijk zijn voor het kunnen afgeven van de garantie dat een item of product zich conformeert aan de vastgelegde technische eisen.
2. Een reeks activiteiten die zodanig is gespecificeerd of opgesteld dat daarmee het proces, waarmee producten worden ontwikkeld of gefabriceerd, kan worden geëvalueerd.

quality attribute *kwaliteitsattribuut*

Een functie, kenmerk of eigenschap die van invloed is op de kwaliteit van een item.

quality control: QC *kwaliteitsbeheersing*

1. Zie de definities van kwaliteitsborging (quality assurance).
2. Het proces waarin iemands werk of dat van een medewerker wordt geverifieerd.

quality factor *kwaliteitsfactor*

Zie kwaliteitsattribuut. Opmerking: In een hiërarchie van kwaliteitsattributen worden bepaalde hogere-niveau attributen kwaliteitsfactoren genoemd en de lagere-niveau attributen kwaliteitsattributen.

quality metric *kwaliteitsmeting*

1. Een kwantitatieve bepaling van de mate waarin zich in een item een gegeven kwaliteitsattribuut manifesteert.
2. Een functie waarvan de invoer uit softwaregegevens bestaat en de uitvoer één specifieke numerieke waarde is die kan worden geïnterpreteerd als de mate waarin zich in een item een gegeven kwaliteitsattribuut manifesteert.

Quantify *Quantify*

Een gereedschap dat kan worden gebruikt bij de prestatie-analyse van programmatuur. Het systeem is ontwikkeld door de firma Pure Software.

query *gegevensopvraag*

Een activiteit die, onder andere, de selectie van objecten uit impliciet of expliciet geïdentificeerde verzamelingen inhoudt. De opvraag is gebaseerd op een gespecificeerd predikaat.

query language *opvraagtaal*

Een taal die wordt gebruikt om toegang te verkrijgen tot informatie die in een database is opgeslagen.

queue *1. queue; 2. wachtrij*

- a. Een lijst met items waarbij aan de laatste positie van die lijst items worden toegevoegd en van de eerste positie items van de lijst worden verwijderd.
- b. Een wachtrij bestaat uit mensen of objecten die op een dienstverlening of het uitvoeren van een activiteit staan te wachten. De meest voorkomende wachtrij is de printqueue. Hierin wachten de printinstructies op het moment waarop elementen van die printqueue kunnen worden afgedrukt.

queueing *in een wachtrij geplaatst zijn*

Een programmeringstechniek die wordt toegepast om berichten, die wachten om te worden verzonden, af te handelen.

quiescing *afwijzing*

Het proces waarbij een eenheid of een systeem wordt gestopt door nieuwe verzoeken om werk af te wijzen.

R

random failure *willekeurige storing*

1. Een storing waarvan het optreden onvoorspelbaar is.
2. Een storing die slechts in de zin van de waarschijnlijkheidsleer of de statistiek voorspelbaar is.

rapid prototyping *snel vervaardigen van prototypes*

Een specifieke manier voor het vervaardigen van prototypes waarbij nadruk wordt gelegd op het ontwikkelen van prototypes in het begin van het ontwikkelingsproces. Dit geschiedt om een vroegtijdige terugkoppeling en analyse van de functies en eigenschappen van het prototype in dit proces mogelijk te maken.

read (to) *lezen*

Toegang verkrijgen tot de gegevens die in een geheugen- of gegevensmedium zijn opgeslagen.

real address *reëel adres*

1. Het adres van een positie in het reële geheugen of het adres van een reële in- en uitvoereenheid.
2. Het adres van een geheugenlocatie in het hoofdgeheugengedeelte van een virtueel geheugensysteem.

real constant *reële constante*

Een rij decimale cijfers die van een decimaalteken, een decimale exponent of beide kunnen zijn voorzien.

real mode *reële werkwijze*

Een werkwijze waarbij een programma niet mag worden gepagineerd.

real storage *reële opslag*

Het hoofdgeheugengedeelte van een virtueel geheugensysteem.

real time *1. onvertraagd; 2. real time*

- a. De werkelijke tijd waarin een fysiek proces plaats vindt.
- b. Een toepassing waarbij de responsie op de ingevoerde gegevens snel genoeg is om invloed uit te oefenen op de daaropvolgende invoer, zoals dat het geval is bij procesbesturingen.
- c. De term wordt over het algemeen gebruikt om systemen te beschrijven die moeten kunnen garanderen dat een reactie of een responsie op een externe gebeurtenis of een externe actie binnen een bepaalde gespecificeerde tijd of binnen de tijd waarin berekeningen worden uitgevoerd of gegevens worden verwerkt, plaats vindt.

real time processing *1. onvertraagd verwerken; 2. real time verwerking*

- a. Een werkwijze van een computer indien deze onvertraagde bewerkingen verricht.
- b. Het verwerken van gegevens die nodig zijn voor of worden gegenereerd door een proces terwijl dit proces wordt uitgevoerd; gewoonlijk worden de resultaten gebruikt om dat proces en de daarmee verwante processen te beïnvloeden.

real type *reëel type*

Een gegevenstype waarvoor geldt dat de bij dat type behorende gegevens reële getallen als waarden aan kunnen nemen en kunnen worden bewerkt door rekenkundige bewerkingen (optellen, aftrekken, vermenigvuldigen, delen, worteltrekken) die op reële getallen kunnen worden uitgevoerd

reboot (to) *rebooten*

Het opnieuw initialiseren van de uitvoering van een programma door de bewerking die het initieel laden van het programma start, te herhalen.

record *record*

1. Een aantal bij elkaar behorende gegevens(items) of woorden die als een eenheid wordt behandeld.
2. Een eenheid voor de gegevenstransmissie volgens de werkwijze die met records samenhangt. Een record vertegenwoordigt een bepaalde hoeveelheid gegevens die een zendend knooppunt op een gegeven moment verstuurt.

recovery *herstel*

De restauratie of het herstel van een systeem, programma, database, of ander systeemelement in een toestand waarin het de vereiste functies uit kan voeren.

recovery test *hersteltest*

Een test die bedoeld is om na te gaan in hoeverre het systeem is hersteld na eventueel opgetreden storingen.

recursion *recursie*

1. Een proces waarin een softwaremodule zichzelf oproept.
2. Het proces waarin de definitie of de generatie van een proces of gegevensstructuur in termen van zichzelf kan worden uitgevoerd.

recursive *recursief*

1. De term heeft betrekking op een softwaremodule dat zichzelf oproept.
2. De term verwijst naar een proces dat of gegevensstructuur die in termen van zichzelf wordt gedefinieerd of gegenereerd.

3. Een proces waarin bij elke stap gebruik wordt gemaakt van de resultaten van de voorgaande stap of voorgaande stappen.

recursive function *recursieve functie*

Een functie waarvan de waarden getallen zijn die van andere getallen zijn afgeleid met behulp van substitutiefomules. Daarin is de recursieve functie een opdrachtelement.

recursive process *recursief proces*

Een proces waarin bij elke stap gebruik wordt gemaakt van de resultaten van eerder uitgevoerde stappen.

recursive routine *recursieve routine*

Een deel van een programma dat op zich zelf als een programma kan worden opgevat en gebruikt en waarbij het zichzelf rechtstreeks oproept of door een ander deel van het programma wordt opgeroepen. Indien gebruik wordt gemaakt van een recursieve routine, is het gewoonlijk nodig om een administratie bij te houden van de status van het niet gereedgekomen gebruik in, bijvoorbeeld, een omgekeerde lijst.

recursive subroutine *recursieve subroutine*

Zie recursieve routine.

redocumentation *herdocumentatie*

Het genereren of het herzien van een semantisch equivalente representatie binnen hetzelfde relatieve abstractieniveau. De daaruit voortvloeiende representatievormen worden meestal als alternatieve voorstellingen (views) beschouwd die slechts bestemd zijn voor menselijk gebruik.

reduced instruction set computer: RISC *computer met een gereduceerd instructieset*

Een computer die een kleine vereenvoudigde reeks veel toegepaste instructies gebruikt om sneller te kunnen werken. De snelheidstoename wordt vaak voor een deel ongedaan gemaakt door de noodzaak meer gecompliceerde software in een dergelijke computer te moeten gebruiken.

redundancy *redundantie*

De aanwezigheid van fouttolerante hulpcomponenten in een systeem die dezelfde of gelijksoortige functies uit kunnen voeren als andere elementen van dat systeem en tot doel hebben storingen te voorkomen of deze te herstellen.

re-engineering *herontwikkeling*

1. Het onderzoeken en modificeren van een systeem op een zodanige wijze dat het in een nieuwe vorm wordt gedefinieerd en geïmplementeerd.
2. Het proces waarin een bestaand softwaresysteem wordt geanalyseerd, veranderd en opnieuw geïmplementeerd om als zodanig in een nieuwe vorm te worden gepresenteerd en gebruikt.

re-entrant *gelijktijdig bruikbaar*

1. De term heeft betrekking op een softwaremodule waar men toegang toe kan verkrijgen als deel van een bepaald proces, terwijl het eveneens als deel van een ander proces werkzaam is en desondanks de gewenste resultaten oplevert.
2. Een asset, module, routine of programma waarvan geldt dat dezelfde kopie hiervan gelijktijdig door twee of meer taken kan worden gebruikt.

re-entrant program *gelijktijdig bruikbaar programma*

Een programma dat herhaald kan worden ingevoerd voordat de voorgaande bewerking(en) van hetzelfde programma gereed is (zijn). Dit is afhankelijk van de eis dat noch de externe programmamaparameters noch de instructies tijdens de uitvoering veranderingen ondergaan. Een dergelijk programma kan door meer dan één programma tegelijkertijd worden gebruikt.

re-entrant routine *gelijktijdig bruikbare routine*

Een deel van een programma dat herhaald kan worden ingevoerd voordat de voorgaande bewerking(en) van hetzelfde programmadeel gereed is (zijn). Dit is afhankelijk van de eis dat noch de externe programmamaparameters noch de instructies tijdens de uitvoering veranderingen ondergaan. Een dergelijk programmadeel kan door meer dan één programma tegelijkertijd worden gebruikt.

re-entrant subroutine *gelijktijdig bruikbare subroutine*

Een subroutine die herhaald kan worden ingevoerd voordat de voorgaande bewerking(en) van dezelfde subroutine gereed is (zijn). Dit is afhankelijk van de eis dat noch de externe programmamaparameters noch de instructies tijdens de uitvoering veranderingen ondergaan. Een dergelijke subroutine kan door meer dan één programma tegelijkertijd worden gebruikt.

re-entry point *terugkeerpunt*

1. De plaats in een softwaremodule waar men opnieuw toegang tot dit module kan verkrijgen nadat een ander module is opgeroepen.
2. Het adres dat of de label die de plaats aangeeft van de instructie waar het programma, dat een module of een subroutine heeft opgeroepen, dit module of de subroutine opnieuw binnenkomt.

reference 1. verwijzing; 2. referentie

Een gegevenselement waarvan de waarde een adres is.

referential integrity *referentiële integriteit*

De eigenschap die garandeert dat een handvat (handle) dat zich in de toestand bevindt die met een ander object is geassocieerd, één enkel object op betrouwbare wijze identificeert.

regression test *regressietest*

Een test die na modificaties van of op een systeem, programma of asset wordt uitgevoerd en waarin wordt nagegaan of die modificaties fouten hebben geïntroduceerd of fouten ontdekken die in eerdere testen niet waren gevonden.

relation 1. relatie; 2. verwantschap

- a. De toestand die tussen twee uitdrukkingen of variabelen kan bestaan ten aanzien van hun scalaire waarde (groter dan, gelijk aan, kleiner dan) of hun logische waarde (waar, onwaar).
- b. De mate waarin of de regelmaat waarmee twee of meer objecten, entiteiten of toestanden dezelfde kenmerken of eigenschappen vertonen.

relational data base management system: RDBMS *relationeel databasebeheerssysteem*

Een databasebeheerssysteem dat gebaseerd is op een relationeel model voor de opslag en het terugzoeken van gegevens. In een dergelijke database zijn de gegevens en de relaties tussen de gegevens in de vorm van tabellen georganiseerd.

relation character *vergelijkingsbewerkingsteken*

Een teken dat de verwantschap tussen twee opdrachtelelementen uitdrukt. Dit zijn de tekens: groter dan, gelijk aan, en kleiner dan.

relationship *verwantschap*

Een vereniging, afhankelijkheid of verbinding tussen twee entiteiten die, bijvoorbeeld, in een database voorkomen of daarin zijn opgenomen.

relative address *relatief adres*

Een adres dat moet worden aangepast door het optellen van een waarde bij dat adres om het adres van de gewenste geheugenlocatie te verkrijgen.

relative addressing *relatieve adressering*

Een adresseringsmethode waarbij het adresgedeelte van een instructie een relatief adres bevat.

relative instruction *relatieve instructie*

Een instructie als gevolg waarvan een beeldscherm eenheid de gegevens, die hierna volgen, interpreteert als relatieve in plaats van absolute coördinaten.

released version *vrijgegeven versie*

De versie van een programma, asset of object die niet verder wordt gewijzigd.

reliability *betrouwbaarheid*

De mogelijkheid dat een systeem of component de vereiste functies onder gedefinieerde condities gedurende een gespecificeerde tijdsduur uit kan voeren.

reliability growth *betrouwbaarheidstoename*

De verbetering van de betrouwbaarheid als gevolg van de correcties van fouten of de vervanging van foutieve bewerkingen.

relocatable *verplaatsbaar*

De term heeft betrekking op code die in elke deel van het hoofdgeheugen kan worden geladen. Het beginadres wordt door het laadprogramma vastgesteld, waarna dit programma de adressen in de code aanpast. De aanpassing is nodig om de verwijzing naar de adressen in die code in overeenstemming te brengen met de werkelijke geheugenposities.

relocatable address *verplaatsbaar adres*

Een adres dat door het laadprogramma wordt aangepast als het computerprogramma dat dat adres bevat in het geheugen wordt geladen.

relocatable code *verplaatsbare code*

Code die adressen bevat die door het laadprogramma moeten worden aangepast.

relocate (to) *verplaatsen*

Machinecode van een bepaald deel van het geheugen verplaatsen naar een ander deel. Daarbij worden de adressen in de code zodanig aangepast dat deze code in de nieuwe locatie correct kan worden uitgevoerd.

relocating assembler *assembleerprogramma met adresberekening*

Een assembleerprogramma dat verplaatsbare code genereert.

relocating loader *laadprogramma met adresberekening*

Een laadprogramma dat verplaatsbare code in het geheugen plaatst en de adressen in de code aanpast. De aanpassing is nodig om de verwijzing naar de adressen in de code in overeenstemming te brengen met de door het laadprogramma vastgestelde geheugenposities.

relocation dictionary *verplaatsingstabel*

Het deel van een objectmodule of laadmodule dat de adressen van het module identificeert die moeten worden aangepast indien code wordt verplaatst.

remark *opmerking*

1. Een beschrijving, verwijzing of verklaring die wordt toegevoegd aan of zich bevindt tussen de opdrachten van een programmeertaal en die geen uitwerking op de uiteindelijke verwerking (compilatie of montage) van de taal heeft.
2. Een opdracht die wordt gebruikt om een programma te documenteren. Deze opdracht bevat gegevens die van nut kunnen zijn bij het verwerken of het onderhouden van het programma dat dit soort opdrachten bevat.

remote job entry: RJE *karwei-invoer op afstand*

Het aanbieden van karweien via een op zekere afstand van de verwerkingseenheid gesitueerde invoereenheid die met behulp van een communicatieverbinding met die verwerkingseenheid is verbonden.

remote procedure call: RPC *op afstand uitgevoerde oproep*

Een oproep van een routine die als resultaat heeft dat code op een ander systeem wordt geactiveerd en verwerkt dan op het systeem waar het verzoek van de oproep van afkomstig is. In een RPC-systeem kunnen oproepende en opgeroepen procedures op andere systemen worden geactiveerd zonder dat de programmeur dit expliciet behoeft te declareren of coderen.

rendezvous *rendezvous*

In de programmeertaal Ada is een rendezvous de methode waarin de activiteiten in de verschillende taken worden gesynchroniseerd.

repetitive addressing *herhalende adressering*

Een impliciete adresseringsmethode waarin van het bewerkingsveld van een computerinstructie wordt aangenomen dat deze de operanden van de laatst uitgevoerde instructie adresseert.

replication *replicatie*

Het herhalen of het dupliceren van een asset binnen een bepaald softwaresysteem.

report standard *rapportstandaard*

Een standaard die de karakteristieke gegevens bevat van de beschrijvende resultaten van ontwikkelings- en managementactiviteiten.

repository *1. vergaarbak; 2. repository*

De term verwijst naar het hart van een CASE-gereedschap dat over het algemeen uit een databasebeheerssysteem waarin alle documenten worden opgeslagen, bestaat.

representation standard *representatiestandaard*

Een standaard die de karakteristieke gegevens bevat van de beschrijvende aspecten van het product dat het resultaat van een ontwikkeling of van het management is.

request *verzoek*

Een gebeurtenis die uit een bewerking met nul of een bepaald aantal actuele parameters bestaat. Een client lanceert een verzoek met het doel een dienst (service) te laten verrichten of uit te laten voeren.

requirement *1. eis; 2. behoefte*

- a. Een conditie, functie of faciliteit die een gebruiker nodig heeft om een probleem op te lossen of een doel te bereiken.
- b. Een conditie, functie of faciliteit waaraan moet worden voldaan of waarin een systeem, systeemcomponent of systeemelement moet voorzien om aan een contract, een standaard, een specificatie, of een ander formeel document te voldoen. De verzameling van alle eisen vormt de basis van een daaropvolgende ontwikkeling van een systeem, module, component of asset. Indien een asset voor hergebruik in aanmerking komt, moet de catalogusinformatie van dat asset worden geëvalueerd om te bepalen in hoeverre het asset aan de eisen van het systeem, waarvoor het zal worden gebruikt, voldoet.
3. Een gedocumenteerde weergave van een conditie, functie of faciliteit zoals in de twee vorige punten van deze definitie zijn omschreven.

requirements *eisen*

Het eerste stadium van de ontwikkeling van software moet bestaan uit het specificeren of definiëren van de eisen zoals die met potentiële gebruikers worden afgestemd en afgesproken. In moderne methoden dienen deze eisen testbaar te zijn en moeten in latere stadia van de ontwikkeling kunnen worden geïdentificeerd en geëvalueerd.

requirements analysis *eisenanalyse*

1. Het proces waarin de eisen van gebruikers worden bestudeerd met het doel een definitie van systeem-, hardware-, en/of software-eisen te kunnen formuleren.
2. Het proces waarin de eisen met betrekking tot systeem, hardware en software worden bestudeerd en verfijnd.

requirements-based testing *op eisen gebaseerd testen*

Het ontwerpen van testen en testseries die gebaseerd zijn op de doelstellingen die uit de eisen voor een softwarecomponent worden afgeleid. Dit zijn bijvoorbeeld testen die specifieke functies onderzoeken of de niet-functionele beperkingen, zoals de prestatie en de beveiliging van het component, sonderen.

requirements phase *eisenfase*

De fase in de levenscyclus van software waarin de eisen voor een softwareproduct worden gedefinieerd en gedocumenteerd.

requirements review *beoordeling van het eisenpakket*

Een proces of een bijeenkomst waarin de systeemeisen, hardware-items, software-items, modules, assets enzovoort aan de projectmedewerkers, managers, gebruikers, klanten en andere geïnteresseerden voor commentaar of goedkeuring worden gepresenteerd. Er bestaan een aantal verschillende soorten van deze vorm van beoordeling zoals beoordeling van de systeemeisen, beoordeling van de software-eisen enzovoort.

requirements specification *eisenspecificatie*

Een document waarin de eisen voor een systeem of component zijn gespecificeerd. Hierin zijn onder andere de volgende onderwerpen opgenomen: functionele eisen, prestatie-eisen, eisen aangaande interfaces, ontwerpisen, ontwikkelingsstandaarden, en eisen betreffende de omgeving en veiligheid.

requirements specification language *eisenspecificatietaal*

Een specificatietaal met speciale constructies die soms is voorzien van verificatieprotocollen. De taal wordt gebruikt om hardware en/of software-eisen te ontwikkelen, te analyseren en te documenteren.

requirement(s) standard *eisenstandaard*

Een standaard waarin de karakteristieken van een eisenspecificatie zijn beschreven.

reserved word *gereserveerd woord*

Een woord in een programmeertaal waarvan de betekenis door de regels van die taal is vastgelegd en dat in een zekere context van de taal of in alle mogelijke omstandigheden niet door de program-

meur kan worden gebruikt anders dan voor het doel waar dit gereserveerde woord voor mag worden gebruikt.

reset (to) *terugstellen*

Een variabele, register, of andere geheugenlocatie terugstellen naar een voorgeschreven toestand.

resident *resident*

Een programma dat zich permanent in het geheugen bevindt, zoals het kernprogramma in het hoofdgeheugen of een systeembibliotheek in een direct toegankelijk geheugen.

resident segment *resident segment*

Een segment dat in het primaire geheugen aanwezig blijft tijdens de uitvoering van een reeks taken.

resident software *residente programmatuur*

Programmatuur die zich in het interne geheugen van de apparatuur bevindt.

residual control *residuale besturing*

Een techniek in de microprogrammering waarbij de betekenis van een veld in een micro-instructie afhankelijk is van de waarde in een hulpregister.

resource *steemelement*

Elke faciliteit van een computer- of besturingssysteem die voor een bewerking of een taak nodig is. Deze omvat eveneens het (hoofd)geheugen, de in- en uitvoereenheden, de verwerkingseenheid, de gegevensverzamelingen en de besturings-, applicatie- en verwerkingsprogramma's.

resource management *beheer van steemelementen*

De identificatie, schatting, allocatie, en controle van de middelen die worden gebruikt om een product te ontwikkelen of een dienst uit te voeren.

resource manager *steemelementbeheerder*

Een algemene aanduiding voor de functie van een besturingsprogramma dat steemelementen toewijst.

response time *1. responsietijd; 2. reactietijd*

De tijd die verloopt tussen het einde van een vraag aan of commando voor een interactief computersysteem en het begin van de responsie of reactie van dat systeem.

restart (to) *herstarten*

De uitvoering van een computerprogramma na een storing doen vervolgen. Hiervoor worden de toestand op en de vastgelegde resultaten van het programma in een bepaald controlepunt, het herstartpunt, gebruikt.

restart point *herstartpunt*

Een punt in een computerprogramma waarin (of waarop) de werking van het programma na een storing kan worden vervolgd.

restructuring *herstructurering*

De transformatie van een bepaalde voorstellingswijze van een component of een systeem in een andere voorstellingswijze op hetzelfde relatieve abstractieniveau. Daarbij dient het externe gedrag (functionaliteit en semantiek) van het getransformeerde component of systeem ongewijzigd te blijven.

retirement *beëindiging*

1. De permanente verwijdering van een systeem of een component uit de operationele omgeving.
2. De beëindiging van de ondersteuning van een operationeel systeem of component.

retirement phase *beëindigingsfase*

De periode in de levenscyclus van software gedurende welke de ondersteuning van een softwareproduct wordt beëindigd.

retrospective trace *retrospectieve trace*

Een naspeuring (trace) die uit de historische gegevens die gedurende de uitvoering van het programma zijn vastgelegd, wordt afgeleid. Opmerking: Deze verschilt van een normale naspeuring die op cumulatieve wijze tijdens de uitvoering van het programma wordt geproduceerd.

return (to) *terugkeren*

1. De besturing van een softwaremodule overdragen aan een ander module die het heeft opgeroepen.
2. Een waarde door een oproepend module aan een parameter toekennen; bijvoorbeeld, het toekennen van de waarde 66 aan de parameter 'Leeftijd' met het doel deze parameter in het oproepende module te kunnen gebruiken.

return code *terugkeercode*

Een code die wordt gebruikt om de uitvoering van een oproepend module volgend op een terugkeer vanuit een opgeroepen module, te beïnvloeden.

return value *terugkeerwaarde*

Een waarde die wordt of is toegekend aan een parameter door een opgeroepen module en die door het oproepende module kan worden verwerkt of gebruikt.

reusability *herbruikbaarheid*

1. De mate waarin een asset of softwaremodule in meer dan één softwaresysteem kan worden gebruikt. Daarbij dienen de aanpassingen van het asset of het module beperkt te blijven.
2. De karakteristieken van een herbruikbaar software-asset dat in een bibliotheek voor hergebruik is opgeslagen. Deze maken het gemakkelijker om dit asset in verschillende contexten te gebruiken.
3. De mogelijkheid code te gebruiken die voor een bepaalde applicatie in een andere applicatie is ontwikkeld. Dit werd traditioneel bereikt door die code in programmabibliotheken op te slaan. De object-georiënteerde programmering bezit de potentie voor een groter hergebruik van code met behulp van technieken zoals overerving, algemeenheid (genericity) enzovoort. Klassebibliotheken die voorzien zijn van intelligente snuffelprogramma's (browsers) en applicatiegeneratoren kunnen hierbij van groot nut zijn.

reusable *herbruikbaar*

De term heeft betrekking op een softwaremodule die of een asset dat in meer dan één computerprogramma of softwaresysteem kan worden gebruikt.

reusable asset *herbruikbaar asset*

Zie herbruikbaar software-asset (reusable software asset).

reusable component *herbruikbaar component*

Zie herbruikbaar softwarecomponent (reusable software component).

reusable software *herbruikbare software*

Software die ontworpen en geïmplementeerd is met het specifieke doel deze opnieuw te kunnen gebruiken. Herbruikbare software is een veelomvattende term die betrekking heeft op assets, applicaties, of softwaresystemen. De aanbevolen term is herbruikbaar software-asset.

reusable software asset: RSA *herbruikbaar software-asset*

Een asset dat in een catalogus is opgenomen en in een bibliotheek voor herbruikbare software is opgeslagen. Aan de term herbruikbaar software-asset kan een grotere betekenis worden toegevoegd dan aan de alleenstaande term 'asset'. Die toegevoegde waarde kan men vinden in de meta-informatie betreffende het asset in de ingang of de declaratie van het asset in de catalogus en de bibliotheek voor hergebruik of herbruikbare systemen waarin het asset is opgenomen.

Deze toevoegingen maken het gemakkelijker om de assets, die voor eventueel hergebruik zijn geëvalueerd, te lokaliseren en te bepalen in hoeverre ze informatie bevatten die voor eventueel hergebruik in een specifieke context nuttig kan zijn. Daarentegen is een normaal asset een softwareproduct dat al dan niet kan worden gelokaliseerd en/of opnieuw kan worden gebruikt, terwijl een asset al dan niet kan zijn gecertificeerd. Dit is afhankelijk van de eisen die aan de bibliotheek voor hergebruik zijn gesteld en de behoeften waarin een dergelijke bibliotheek kan voorzien. Een asset kan eveneens in een vergaarbak (repository) zijn opgeslagen.

reusable software component: RSC *herbruikbaar softwarecomponent*
De term is synoniem met die van herbruikbaar software-asset (reusable software asset).

reusable software library *herbruikbare softwarebibliotheek*
Zie herbruikbare bibliotheek (reusable library).

reuse (to) *opnieuw gebruiken*

Het proces van de implementatie of de aanpassing van softwaresystemen waarbij gebruik wordt gemaakt van bestaande software-assets. Het herhaalde gebruik of hergebruik bestaat uit het toepassen van herbruikbare software-assets, met of zonder aanpassingen, voor meer dan één softwaresysteem. Hergebruik kan binnen een bepaald softwaresysteem, in gelijksoortige softwaresystemen, of in sterk van elkaar verschillende softwaresystemen worden toegepast.

reuse-based development *op hergebruik gebaseerde ontwikkeling*

Het toepassen van een gedisciplineerde, systematische en kwantificeerbare benadering in de ontwikkeling, de bedrijfsvoering en het onderhoud van software. Hierbij maakt hergebruik een belangrijk deel van de benadering uit. Op hergebruik gebaseerde ontwikkelingen maken gebruik van producten uit de domeinontwikkeling tijdens de ontwikkeling van systemen of software.

reuse library *bibliotheek voor hergebruik*

Een goed beheerbare verzameling van herbruikbare software-assets die voorzien is van procedures en ondersteunende functies die nodig zijn om de herbruikbare software-assets op te vragen. De procedures en ondersteuningsfuncties kunnen met behulp of door middel van een bibliotheekstelsel voor hergebruik zijn geautomatiseerd. Indien dit het geval is, zal de bibliotheek voor hergebruik zowel de herbruikbare software-assets als het bibliotheekstelsel voor hergebruik omvatten.

reuse library interoperability *bibliotheeken voor hergebruik die onderling samenwerken*

De term heeft betrekking op de potentie van twee of meer duidelijk van elkaar verschillende heterogene bibliotheken voor hergebruik die op dynamische wijze toegang tot elkaars assets, beschrijvingen van assets en andere beschikbare informatie hebben.

reuse library system *bibliotheekstelsel voor hergebruik*

Een softwaresysteem dat de procedures en ondersteuningsfuncties van een bibliotheek voor hergebruik automatiseert. Een bibliotheekstelsel voor hergebruik bevat de opslagfuncties en -mogelijkheden van een software-vergaarbak (repository), maar is meer dan een faciliteit voor het opslaan van informatie. Het bezit functies die de gebruiker helpen bij het verkrijgen van toegang tot de inhoud van datgene wat is opgeslagen (snuffelen, hypertext enzovoort). Herbruikbare assets worden over het algemeen in een bibliotheek voor hergebruik die door een bibliotheekstelsel voor hergebruik wordt ondersteund, opgeslagen.

reverse engineering *1. terugwerkend ontwikkelen; 2. herontwikkelen*

- a. Het proces waarin een bestaand systeem wordt geanalyseerd met het doel de componenten en elementen en hun onderlinge relaties te identificeren en representaties van het systeem te creëren die een andere vorm of voorstellingswijze hebben of een hoger abstractieniveau bezitten. Een terugwerkende ontwikkeling vindt meestal plaats om een systeem te creëren dat functionaliteiten heeft die identiek zijn aan die van het oorspronkelijke systeem maar waarvan het onderhoud eenvoudiger is en beter kan worden uitgevoerd.
- b. Zie herontwikkeling (re-engineering).

reversible execution *omgekeerde uitvoering van bewerkingen*

Een foutopsporingstechniek waarin de geschiedenis van de bewerkingen tijdens de uitvoering van een programma is vastgelegd en daarna onder besturing van de gebruiker weer wordt herhaald in of de normale of de tegenovergestelde richting.

review *beoordeling*

Een proces of een bijeenkomst gedurende welke een product, een werkwijze, of een project aan ontwikkelaars, een projectgroep, managers, gebruikers enzovoort wordt gepresenteerd. In een dergelijk proces worden de (tussen)resultaten of de werkmethoden besproken en beoordeeld en de resultaten van het beoordelingsproces vastgelegd. Voorbeelden van beoordelingen zijn de codebeoordeling, de ontwerpbeoordeling, de formele kwalificatiebeoordeling, de beoordeling van het eisenpakket, en de beoordeling van de voorlopige testen.

robustness *robuustheid*

De mate waarin een systeem of component toch nog correct kan functioneren ondanks het feit dat het systeem of component met onjuiste invoergegevens of in slechte omgevingscondities moet werken.

roll in (to) *heropslaan*

Gegevens of segmenten van computerprogramma's vanuit het hulpgeheugen of het secundaire geheugen overbrengen naar het hoofdgeheugen.

roll out (to) *uitlezen*

Gegevens of segmenten van computerprogramma's overhevelen van het hoofdgeheugen naar het hulpgeheugen of secundaire geheugen met het doel ruimte in het hoofdgeheugen voor ander gebruik te creëren.

root *wortel*

In een hiërarchie van items is de wortel het item waarvan alle andere zijn afgeleid of waar alle andere items van afstammen. Het wortelitem (of hoofditem) kent geen items die zich op een hiërarchisch hoger niveau bevinden.

root compiler *hoofdcompiler*

Een compiler waarvan de uitvoer uit een machine-onafhankelijke en eventueel tussentijdse representatie van een programma bestaat. Een hoofdcompiler die met een codegenerator wordt gecombineerd, kan als een complete compiler worden beschouwd.

root directory *1. hoofddirectory; 2. hoofdindex*

De op het hoogste niveau aanwezige directory (index). Alle bestanden maken òf deel uit van die hoofddirectory òf zijn in deeldirectories (subdirectories) van de hoofddirectory (of andere deeldirectories die hiërarchisch op een hoger niveau staan) geplaatst.

routine *routine*

1. Een deelprogramma dat door andere programma's en deelprogramma's wordt opgeroepen.
2. Een op zich zelf staand gedeelte van een programma dat of een serie instructies die door een programma wordt opgeroepen en vaak wordt gebruikt.

rule-based *op regels gebaseerd*

De term heeft betrekking op systemen en software die kunnen redeneren of waarin regels de wijze van uitvoering bepalen.

rule-based language *op regels gebaseerde taal*

Een niet-procedurele taal die de gebruiker in staat stelt een reeks regels te definiëren en vragen of problemen met behulp van deze regels te formuleren.

run *verwerkingsgang*

De ononderbroken uitvoering of verwerking van een programma, een programmadeel of één of meer bewerkingen.

run stream *verwerkingsstroom*

Zie jobstream (job stream).

run time *uitvoeringstijd*

1. De tijd nodig om een instructie uit het geheugen te halen, te decoderen en uit te voeren.
2. De tijd gedurende welke een programma de geassembleerde of gecompileerde programma-instructies uitvoert.
3. Het moment waarop een computer met de verwerking of uitvoering van een programma start.

runtime system *runtime-systeem*

De software-omgeving waarin programma's voor computers of programma's die voor een virtuele Java-machine zijn gecompileerd, kunnen worden verwerkt. Het runtime-systeem omvat alle code die nodig is om programma's te laden, op dynamische wijze oorspronkelijke (native) methoden te koppelen, het geheugen te beheren, en uitzonderingssituaties te behandelen. Het systeem kan verder een implementatie zijn van een virtuele Java-machine (dit kan een Java-vertolkingsprogramma zijn).

S

salvage *berging*

Het proces dat wordt gebruikt om een bestaand component te vinden en met behulp van het herontwikkelingsproces bij te werken (op te knappen) of aan te passen zodat het eventueel kan worden gebruikt in toekomstige of in ontwikkeling zijnde applicaties.

sandbox *zandbak*

De term heeft betrekking op een reeks onderling samenwerkende systeemcomponenten variërend van beveiligingsprogramma's die deel van de applicatie uitmaken tot aan beveiligingsmaatregelen die in een virtuele Java-machine en de taal zelf zijn verwerkt. De zandbak garandeert dat een twijfelachtig en mogelijk niet te vertrouwen applicatie zich geen toegang tot het systeem weet te verschaffen.

save area *opslaggebied*

Het gebied in het hoofdgeheugen waarin de inhoud van registers wordt bewaard.

scaffolding *steunsoftware*

Computerprogramma's en gegevensbestanden die zijn gebouwd met het doel de ontwikkeling en het testen van softwareproducten te ondersteunen, maar die geen deel van het uiteindelijke product uit zullen gaan maken. Bijvoorbeeld, dummy bestanden of routines, generatoren van testseries, software-supervisieprogramma's, stubs enzovoort.

scalar expression *scalaire uitdrukking*

Een uitdrukking die één enkele waarde in plaats van een reeks waarden vertegenwoordigt.

scalar function *scalaire functie*

Routine die scalaire items (waaronder objecten) bevat waarop bewerkingen kunnen worden uitgevoerd.

scalar item *scalair item*

Eén enkel item, gegeven of element.

scalar processor *scalaire verwerkingseenheid*

Een verwerkingseenheid die één berekening tegelijk uit kan voeren.

scalar type object *scalair objecttype*

Een object waarvan de waarde niet uit een component bestaat.

scalar variable *scalaire variabele*

Een variabele die maar één enkel gegeven kan representeren.

scenario *scenario*

Een tabel of lijst met definities die een bepaald proces, een bepaalde bewerking of een communicatie beschrijven.

scheduler *1. planningsprogramma; 2. roosterprogramma*

Een computerprogramma dat meestal deel uitmaakt van een besturings-systeem en dat karweien roostert, initieert en beëindigd.

Schlaer-Mellor *Schlaer-Mellor*

Een modelleringsmethode voor de object-georiënteerde analyse (OOA) die van toepassing is op de integratie van eigenschappen die van structurele aard zijn en het gedrag beïnvloeden.

scope *1. bereik; 2. scoop*

- a. Gebied waarbinnen een verschijnsel op kan treden.
- b. Waardebereik
- c. Een kenmerk van een identificatiesymbool (identifier) dat aangeeft waar dit identificatiesymbool kan worden gebruikt. De meeste identificatiesymbolen in, bijvoorbeeld, Java hebben òf een klasse òf een lokaal bereik. Exemplaar- en klassevariabe-

len en -methoden hebben het bereik van een klasse; ze kunnen slechts buiten de klasse en haar deelklassen worden gebruikt door hen van een prefix van een exemplaar (instance) van de klasse of (voor klassevariabelen en methoden) van de naam van de klasse te voorzien. Alle overige variabelen worden binnen methoden gedeclareerd en hebben een lokaal bereik; ze kunnen slechts binnen een omhullend blok worden gebruikt.

screen editor *schermopmaakprogramma*

Een tekstopmaakprogramma waarmee tekst die een geheel scherm vult, kan worden verwerkt.

scripting language *scriptietaal*

Hoogniveau-taal waarmee diverse componenten tot één geheel kunnen worden geassembleerd. De acties die moeten worden uitgevoerd, worden door de scriptie van de taal bestuurd. Scripties zijn zeer beknopt en worden gebruikt om specifieke problemen sneller op te lossen dan het geval is met procedurele talen zoals C. Scriptietaalen richten zich sterk op bedrijfs-, beheers- en besturingsproblemen van de applicatie (inclusief het beheer van het geheugen) terwijl de routinematige zaken door het taalverwerkingsprogramma (en niet de software-ontwikkelaar) worden opgelost.

search (to) *opzoeken*

Het onderzoeken van een reeks elementen op de aanwezigheid van één of meer elementen die een bepaald kenmerk of een bepaalde eigenschap bezitten.

second generation language: 2GL *tweede-generatietaal*

Zie assembleertaal (assembly language).

section *sectie*

1. De logische opeenvolging van één of meer paragrafen.
2. Een groep opeenvolgende indexingangen in een indexrecord. De indexingangen worden, om het opzoeken van een ingang te versnellen, in ongeveer evenveel secties verdeeld als de wortel uit het aantal ingangen.

section header *sectie-aanhef*

Een combinatie van woorden die voorafgaat aan en de identiteit bepaalt van elke sectie in de verwerkingsomgeving-, de gegevens- en de procedure-afdeling. De term is van toepassing op de programmeertaal Cobol.

section name *sectienaam*

Een woord dat voorafgaat aan en de identiteit bepaalt van een sectie in een procedure-afdeling.

secure socket layer: SSL *SSL*

Een protocol waarmee communicaties tussen een Web-browser en een server kunnen worden versleuteld. Dit wordt gedaan ten behoeve van het kunnen garanderen van de privacy van de verzenders en ontvangers van berichten of gegevens.

security kernel *beveiligde kern(el)*

Een kleine zelfstandige verzameling opdrachten die een grote beveiligingsgraad hebben en die als een geprivilegieerd deel van een besturingssysteem werken. De kern specificeert criteria en past deze toe. Programma's die men wenst te gebruiken en gegevens waartoe men toegang wenst te hebben, moeten aan deze criteria voldoen.

security testing *testen van de beveiliging*

Testen die worden uitgevoerd om na te gaan in hoeverre de specificaties betreffende de beveiliging zijn geïmplementeerd en naar behoren functioneren.

segment *segment*

1. Eén van de deelsystemen of combinaties van deelsystemen die een compleet systeem constitueren; bijvoorbeeld, het grootboek van een financieel systeem.
2. Een zelfstandig deel van een computerprogramma dat bij het toekennen van geheugenlocaties kan worden uitgevoerd zonder dat het gehele programma in het hoofdgeheugen aanwezig behoeft te zijn.
3. Een aantal gegevens dat als een eenheid is opgeslagen of als zodanig wordt getransporteerd.
4. In de padanalyse is een segment een reeks computeropdrachten tussen twee opeenvolgende sprongpunten.

segment (to) *segmenteren*

Een systeem, computerprogramma of gegevensbestand in segmenten opdelen.

selective dump *selectieve dump*

Een dump van een gespecificeerde geheugenruimte.

selective trace *selectieve trace*

Een variabele naspeuring (trace) waarin slechts de gespecificeerde variabelen worden onderzocht.

selector *kiezer*

In Smalltalk of Objective-C heeft de term betrekking op de syntax van een bericht dat of boodschap die een specifieke methode in het doelobject selecteert.

self-descriptive *zelf-beschrijvend*

De mate waarin een systeem of component over voldoende informatie beschikt om de doelstellingen en de eigenschappen van dat systeem of component met die informatie te kunnen verduidelijken.

self-documented *zelf-gedocumenteerd*

De term heeft betrekking op de broncode die commentaar bevat waarmee het doel, de werking, en andere informatie die nuttig is voor het begrijpen en het onderhouden van de code, wordt uitgelegd.

self-reference *zelf-verwijzing*

De mogelijkheid van een methode om het object of de objecten te bepalen die in het verzoek voor het verlenen van een dienst, die door de methode zelf moet worden verricht, is vervat.

self-relative address *autorelatief adres*

Een adres dat moet worden opgeteld bij het adres van een opdracht en waarin dat adres voorkomt om het adres van de gewenste geheugenlocatie te verkrijgen.

semantic error *semantische fout*

Een fout die het gevolg is van het niet begrijpen van de relatie die tussen symbolen of groepen symbolen en hun betekenis(sen) in een gegeven taal bestaat.

semantics *semantiek*

De relatie die tussen symbolen en groepen symbolen en hun betekenis(sen) in een gegeven taal bestaat.

semaphore *semafoor*

1. Een gedeelde variabele die wordt gebruikt om gelijktijdig uitgevoerde processen te synchroniseren en met behulp waarvan kan worden aangegeven dat een actie reeds is voltooid of een gebeurtenis heeft plaatsgevonden.
2. De klassieke methode voor het beperken van de toegang tot gegevens die door verschillende met elkaar samenwerkende processen worden gedeeld.

separator *scheidingstekens*

1. Een identificatieteken dat gegevens van elkaar scheidt of indeelt.
2. Een rij van één of meer tekens die wordt gebruikt om elementen van programma's of gegevens van elkaar te scheiden of in te delen. Hiervoor worden vaak haakjes, het spatieteken, de puntkomma en wiskundige bewerkingstekens, alsmede de sleutelwoorden IF, BEGIN en END gebruikt.
3. Een teken dat woorden en waarden in een invoerregel in groepen indeelt of van elkaar scheidt.

4. Alle bewerkingstekens en commentaren en de volgende tekens: procent, haakjes, komma, punt, puntkomma, toewijzingssymbool en spatieteken; zij bepalen de grenzen van identificatiesymbolen, constanten, beeldspecificaties en sleutelwoorden.

sequence diagram *1. volgordeschema; 2. volgordediagram*

De term wordt geïdentificeerd met een object-interactie of een diagram waarin de volgorde van gebeurtenissen wordt uitgebeeld. De term wordt gebruikt om de logica van het scenario van een gebruikgeval nauwgezet te definiëren. Volgordeschema's worden meestal gebruikt om gebruikgevallen te valideren. Ze beelden het type van de objecten uit die in een gebruikgeval een rol spelen, de berichten die naar elkaar worden verstuurd, alsmede de terugkeerwaarden die met de berichten zijn geassocieerd.

sequence symbol *volgordesymbool*

Een symbool dat bij conditionele instructies als een spronglabel wordt gebruikt. Dit symbool bestaat uit een punt, gevolgd door één tot zeven alfanumerieke tekens waarvan het eerste een letter uit het alfabet dient te zijn.

sequential *1. sequentieel; 2. achtereenvolgens*

De term heeft betrekking op de occurrence van twee of meer gebeurtenissen of activiteiten die op een zodanige wijze optreden of voorkomen dat de ene moet zijn voltooid voordat de volgende kan starten.

sequential cohesion *sequentiële cohesie*

Een soort cohesie waarin de uitvoer van een taak die door een softwaremodule wordt afgewerkt, de invoer voor een andere taak is die door die module moet worden uitgevoerd.

serial *1. serieel; 2. achtereenvolgens*

- a. De term verwijst naar de sequentiële overdracht, occurrence, of verwerking van de afzonderlijke delen van een geheel, zoals de bits van een teken, waarbij dezelfde faciliteiten of functies voor de daaropvolgende delen worden gebruikt.
- b. Zie sequentieel.

serial construct *seriële constructie*

Een programmaconstructie die uit een serie opeenvolgende stappen bestaat waarin geen lussen of beslissingsbewerkingen voorkomen.

server *server*

1. Informatiesysteem dat aan iedere gebruiker van een datanetwerk diensten verleent, met name diensten ten aanzien van het raadplegen van gegevensbestanden.

2. Individu, component van een computersysteem, kanaal enzovoort, die/dat de leden/elementen uit een wachtrij bedient.
3. Een computer die met behulp van netwerkverbindingen delen van een informatieverwerkende taak uitvoert. Dit wordt gedaan op verzoek en ten behoeve van één of meer op afstand opgestelde andere computers.

server object *dienstverlenend object*

Een object dat op een verzoek voor het verlenen van een dienst of diensten reageert.

service *1. service; 2. dienstverlening*

Een berekening die als reactie op een verzoek kan worden uitgevoerd.

servlet *servlet*

Een programma dat zich in een server bevindt of daarmee is geassocieerd en dat Java-georiënteerde servers een toegevoegde functionaliteit verschaft. Servlets zullen in de toekomst in het World Wide Web een grote rol gaan spelen.

set *1. verzameling; 2. reeks*

Een eindig of oneindig aantal willekeurige objecten, entiteiten of begrippen die een bepaalde eigenschap gemeen hebben.

set-up time *insteltijd*

De tijdsduur waarin een systeem of component wordt geprepareerd om een specifieke bewerking uit te voeren.

shell *1. shell; 2. schil*

- a. Een software-interface tussen een gebruiker en het besturingssysteem van een computer. Shellprogramma's interpreteren opdrachten en interacties van de gebruikers op eenheden zoals toetsenborden, aanwijseenheden en aanraakgevoelige schermen en communiceren deze met een besturingssysteem. Shells vereenvoudigen de interacties met de gebruiker door de bemoeienis van de gebruiker met de eisen van het besturingssysteem te elimineren. Een computer kan een aantal lagen met shells hebben.
- b. Programmatuur met behulp waarvan een kernelprogramma onder verschillende omstandigheden die op het besturingssysteem betrekking hebben, kan werken.
- c. Een Unix-programma dat reageert op de ingetoetste commando's die, indien correct, daarna door Unix worden verwerkt. Er bestaan verschillende Unix-shells, zoals de Bourne-shell en de C-shell.

- d. De buitenschil van een besturingssysteem dat de gebruikersinterface bevat. Dit in tegenstelling tot de kern (kernel) waarin de belangrijkste functies en diensten zijn opgenomen.

shell script *shell-script*

1. Een programma dat door de shell (schil) van een besturingssysteem kan worden geïnterpreteerd.
2. Een bestand dat een lijst met Unix-shell commando's bevat. Men kan de verwerking van een shell-script starten, waarna alle instructies in het script worden uitgevoerd.

signature *signatuur*

De term definieert de soorten parameters die voor een bepaalde bewerking van toepassing zijn.

simple buffering *eenvoudige bufferbewerking*

Een buffertechniek waarin een buffer aan een computerprogramma wordt toegewezen. Dit geschiedt gedurende een periode die overeenkomt met de uitvoeringstijd van het programma.

simple subpath *eenvoudige deelroute*

Een (deel)route van de besturingsstroom waarin geen enkel programma of deel van een programma zonder noodzaak wordt uitgevoerd.

simplicity *eenvoud*

De kenmerkende eigenschap van een systeem of component dat een ontwerp en een implementatie heeft die recht-toe recht-aan is en gemakkelijk kan worden begrepen.

simulation *simulatie*

1. Een model dat zich gedraagt zoals of op dezelfde wijze werkt als een gegeven systeem indien het wordt voorzien van speciaal geselecteerde invoergegevens.
2. Het proces waarin een model als beschreven in punt 1 van deze definitie wordt ontwikkeld of gebruikt.
3. De nabootsing en weergave van geselecteerde gedragingen en karakteristieke eigenschappen en kenmerken van een bepaald fysiek of abstract systeem door een ander systeem.

simulator *simulator*

Een eenheid, computerprogramma, of systeem dat bij de verificatie van software wordt gebruikt en dat zich als een bepaald systeem gedraagt of als zodanig werkt indien het van de juiste invoergegevens wordt voorzien.

simultaneous *1. simultaan; 2. gelijktijdig*

De term heeft betrekking op de occurrence van twee of meer gebeurtenissen die op hetzelfde moment plaats vinden.

simultaneous operation *simultane bewerking*

Een werkwijze waarbij twee of meer gebeurtenissen op hetzelfde tijdstip plaatsvinden.

simultaneous processing *simultane verwerking*

Het plaatsvinden van twee of meer verwerkingen op hetzelfde tijdstip.

simultaneous recursion *simultane recursie*

Een situatie die ontstaat of optreedt indien twee softwaremodules elkaar oproepen.

single inheritance *enkelvoudige overerving*

1. De constructie van een definitie die wordt gerealiseerd met behulp van een incrementele wijziging van een bepaalde definitie.
2. De eigenschap van een object-georiënteerde taal waarin het afleiden van een deelklasse uit slechts één ouder aan beperkingen onderhevig is.

single-level encoding *codering op enkelvoudig niveau*

Een techniek in de microprogrammering waarin verschillende microbewerkingen als verschillende waarden in hetzelfde veld van een micro-instructie worden gecodeerd.

single-step operation *1. enkelvoudige bewerking; 2. stapsgewijze bewerking*

Een foutopsporingstechniek waarin één enkele computerinstructie of deel van een dergelijke instructie als een reactie op een extern signaal wordt uitgevoerd.

sizing *1. bemeting; 2. schatting van de omvang*

Het proces waarin de geheugenruimte of het aantal lijnen broncode dat nodig is voor een softwaresysteem of component, wordt geschat.

Smalltalk *Smalltalk*

Een baanbrekend object-georiënteerd programmeringssysteem dat door het Palo Alto Research Centrum van de firma Xerox is ontwikkeld. Het systeem beschikt over een taal (die meestal wordt geïnterpreteerd), een programmeringsomgeving, en een uitgebreide objectbibliotheek.

snapshot dump *snapshot dump*

Een dynamische dump van de inhoud van één of meer gespecificeerde geheugengebieden.

soft failure *zachte storing*

Een storing die een systeem in staat stelt met een beperkte functionaliteit of gedeeltelijke verwerkingsmogelijkheden de werking te vervolgen.

soft link *zachte koppeling*

Een koppeling die de naam van een ander bestand bevat dat al dan niet in een ander bestandssysteem voor kan komen. Een zachte koppeling wekt de indruk dat een bestand, dat in een ander bestandssysteem is opgenomen, in de directory van het eigen bestandssysteem voorkomt. Een zachte koppeling wordt ook wel een symbolische koppeling genoemd.

software *1. programmatuur; 2. software*

- a. Programma's, procedures, regels en alle bijbehorende documentatie en gegevens die betrekking hebben op de werking van een computersysteem.
- b. Een verzameling instructies die door een computer worden geïnterpreteerd en verwerkt. In tegenstelling tot hardware (dat de fysieke componenten van de computer bevat) bestaat software uit informatie die op een schijf, een band, kaarten, of in het geheugen van de computer is opgeslagen.

software application *software-applicatie*

Zie applicatie (application).

software architecture *software-architectuur*

Zie architectuur (architecture).

software asset *software-asset*

Zie asset.

software bus *softwarebus*

Een ondersteuningsomgeving voor heterogene gedistribueerde verwerkingen.

software characteristic *softwarekarakteristiek*

Inherente en mogelijk toevallige karakteristieke kenmerken, kwaliteitsaspecten of eigenschappen van software. Bijvoorbeeld, functionaliteit, prestatie, attributen, ontwerpbeperkingen, aantal toestanden, aantal lijnen of aantal sprongen.

software compatible *softwarecompatibel*

De term wordt gebezigd in een situatie waarbij een systeem dezelfde programmatuur verwerkt of kan verwerken als enig ander systeem.

software design description: SDD *software-ontwerpbeschrijving*

Een voorstellingswijze van software die is geproduceerd ten behoeve van de ondersteuning in de analyse, planning en implementatie van software, en in het maken of nemen van beslissingen. De software-ontwerpbeschrijving wordt als een medium gebruikt voor de verstrekking en uitwisseling van informatie betreffende het software-ontwerp, en kan als een blauwdruk of model van het systeem worden beschouwd.

software development cycle *software-ontwikkelingscyclus*

De tijdsperiode die met de beslissing om een softwareproduct te ontwikkelen aanvangt en eindigt op het moment dat de software is of wordt afgeleverd. Deze cyclus omvat een eisenfase, ontwerpfase, implementatiefase, testfase, en soms een installatie- en nazorgfase. Opmerking 1: De hierboven vermelde fasen kunnen elkaar overlappen of kunnen op iteratieve wijze worden uitgevoerd. Dit hangt van de toegepaste ontwikkelingsbenadering af. Opmerking 2: De term wordt soms gebruikt om een langere tijdsperiode aan te geven. Dat is òf de periode die eindigt op het moment dat de software niet langer door de ontwikkelaar wordt verbeterd of uitgebreid òf de totale levenscyclus van de software.

software development file: SDF *software-ontwikkelingsbestand*

Het materiaal dat voor de ontwikkeling van een gegeven software-eenheid of reeks onderling aan elkaar gerelateerde eenheden van pertinent belang is. De inhoud van dit bestand bevat de eisen, het ontwerp, technische rapporten, lijsten met of afdrukken van de code, testplannen, testresultaten, probleemrapporten, roosters, en notities betreffende de eenheden.

software development library *software-ontwikkelingsbibliotheek*

Een softwarebibliotheek die computer-interpreteerbare en mens-leesbare informatie bevat die relevant is voor een software-ontwikkelingsproject.

software development plan: SDP *software-ontwikkelingsplan*

Een projectplan voor een software-ontwikkelingsproject.

software development process *software-ontwikkelingsproces*

Het proces waarin de wensen en behoeften van gebruikers in een softwareproduct worden vertaald. Het proces omvat de vertaling van de wensen en behoeften van gebruikers in een eisenpakket, de transformatie van de software-eisen in het ontwerp, de implementatie van het ontwerp in code, het testen van de code, en soms de installatie van de code en het controleren van de werking van de software in een operationele omgeving. Opmerking: Deze activiteiten kunnen elkaar overlappen of kunnen op iteratieve wijze worden uitgevoerd.

software diversity *software-diversiteit*

Een software-ontwikkelingstechniek waarin twee of meer functioneel identieke varianten van een programma op basis van dezelfde specificaties door verschillende programmeurs of teams van programmeurs worden ontwikkeld met het doel fouten te ontdekken, de betrouwbaarheid te vergroten, additionele documentatie te produceren, of de waarschijnlijkheid te reduceren dat programmeringsfouten of compiler- of compilatiefouten van invloed op het eindresultaat zullen zijn.

software engineering *1. programmatuurontwikkeling; 2. software-ontwikkeling; 3. software-engineering*

- a. Het gebruik en de toepassing van een systematische, gedisciplineerde, en kwantificeerbare benadering van de ontwikkeling, de bewerkingen, en het onderhoud van software; dat wil zeggen, het gebruik van ontwikkelingsprincipes bij de ontwikkeling van software.
- b. Een systematische benadering van de analyse, het ontwerp, de implementatie en het onderhoud van software. Dit omvat over het algemeen het gebruik van software-ontwikkelingsgereedschappen (CASE-tools) in de levenscyclus van software. Er bestaan verschillende modellen van de levenscyclus van software, alsmede een groot aantal methodologieën die in de diverse fasen van de levenscyclus kunnen worden toegepast.
- c. De studie van de benaderingen die in de punten a) en b) van deze definitie zijn genoemd.

software engineering environment: SEE *software-ontwikkelingsomgeving*

De ondersteunende hardware, software en firmware die gedurende de productie van software tijdens de gehele levenscyclus van die software wordt gebruikt. Typische componenten van een dergelijke omgeving omvatten: computerapparatuur, compilers, assembleerprogramma's, besturings- en/of bedrijfssystemen, foutopsporingstechnieken, simulatoren, emulatoren, testgereedschappen, documentatiesystemen, eisenspecificatie- en ontwerpmethoden, ontwikkelingsgereedschappen, en databasebeheerssystemen. Ook kan een bibliotheek voor hergebruik tot deze omgeving behoren. Andere gereedschappen kunnen het hergebruik vergroten en kunnen worden gebruikt om de kwaliteit van de informatie van, bijvoorbeeld, assets in een daarvoor bestemde catalogus verbeteren.

software feature *1. softwarekenmerk; 2. karakteristieke eigenschap van software*

- a. Een duidelijk herkenbare karakteristieke eigenschap of kenmerk van een software-item. Bijvoorbeeld, prestatie, portabiliteit, of functionaliteit.

- b. Een softwarekarakterstiek die in de softwaredocumentatie is gespecificeerd of daarin wordt geïmpliceerd (functionaliteit, prestatie, attributen, ontwerpbeperkingen enzovoort).

software item *software-item*

Broncode, objectcode, karweibesturingscode, besturingsgegevens, of een verzameling van deze items.

software library *softwarebibliotheek*

Een goed georganiseerde en beheersbare verzameling software met de daarbij behorende documentatie die is opgezet om de software-ontwikkeling, alsmede het gebruik en het onderhoud van software, te ondersteunen.

software life cycle *levenscyclus van software*

De tijdsperiode die met de conceptie van een softwareproduct aanvangt en eindigt op het moment dat de software niet langer wordt gebruikt. De levenscyclus van software omvat de conceptfase, de eisenfase, de ontwerpfase, de implementatiefase, de testfase, de installatiefase, de controlefase, de operationele fase (de fase waarin de software wordt gebruikt) en de onderhoudsfase. Soms wordt hier nog een fase aan toegevoegd namelijk de beëindigingsfase. Opmerking: Deze fasen kunnen elkaar overlappen of iteratief worden uitgevoerd.

software metrics *softwaremetingen*

Metingen van de kwaliteit van software die een aanwijzing zijn voor de complexiteit, de begrijpelijkheid, de testbaarheid, de beschrijving en de ingewikkeldheid van de code.

software monitor *softwaremonitor*

Een softwaregereedschap dat tegelijk met een ander programma wordt uitgevoerd en gedetailleerde informatie verzameld over de werking van dat andere programma.

software product *softwareproduct*

1. De complete reeks computerprogramma's, procedures, en mogelijk daarmee verwante documentatie en gegevens die bestemd is om aan gebruikers te worden afgeleverd.
2. Elk afzonderlijk item dat in punt 1 van deze definitie is vermeld.

software repository *1. softwarerepository; 2. softwarevergaarbak*

- a. Een softwarebibliotheek die als een permanent archief voor software en de daarbij behorende documentatie fungeert.
- b. Een permanente en archief-achtige opslagplaats voor software en de daarmee geassocieerde documentatie. Een dergelijke repository is vaak een eenvoudig opslagmechanisme voor assets.

Dit in tegenstelling tot een bibliotheekstelsel voor hergebruik dat zowel opslagfuncties als gebruikersvriendelijke mechanismen bevat om opgeslagen herbruikbare software-assets te vinden en op te halen.

software requirements review: SRR *beoordeling van software-eisen*

1. Een beoordeling van de eisen die voor één of meer softwareconfiguraties zijn gespecificeerd met het doel de mate waarin deze voldoen aan de systeemeisen en de wijze waarop die eisen in de softwareconfiguraties worden geïnterpreteerd te evalueren, alsmede te bepalen in hoeverre ze een voldoende goede basis vormen voor de start van het voorlopige ontwerp van deze items.
2. Een beoordeling zoals vermeld in punt 1 van deze definitie en waaraan elk willekeurig softwarecomponent kan worden onderworpen.

software requirements specification: SRS *software-eisenspecificatie*

1. De documentatie van de belangrijkste eisen (functies, prestaties, ontwerpbeperkingen, en attributen) van de software en de externe interfaces ervan.
2. Een functie of fase in de ontwikkeling van software die van toepassing is op een systeem of een systeemelement en die resp. moet worden uitgevoerd of doorlopen om te kunnen voldoen aan contractuele verplichtingen, standaarden, specificaties, of andere formeel geaccepteerde, goedgekeurde en vereiste documenten.

software reuse library *bibliotheek voor hergebruik van software*
Zie bibliotheek voor hergebruik (reuse library).

software specification review: SSR *softwarespecificatiebeoordeling*
Zie beoordeling van software-eisen (software requirements review).

software system *1. softwaresysteem; 2. programmatuursysteem*

Een georganiseerde verzameling computerprogramma's, procedures, routines, daarmee samenhangende documentatie en gegevens (zoals assets) die betrekking hebben op de werking van een computersysteem waarmee bepaalde functies of combinaties van functies moeten worden uitgevoerd.

software test incident *softwaretestincident*

Elke gebeurtenis die tijdens de uitvoering van een softwaretest kan optreden of voorkomen en een verder onderzoek vereist.

software tool *softwaregereedschap*

Een computerprogramma dat in de ontwikkeling, het testen, de analyse, of het onderhoud van een programma of de daarbij behorende documentatie wordt gebruikt. Voorbeelden van dergelijke gereedschappen zijn: vergelijkers, kruisverwijzingsgeneratoren, decompileerprogramma's, drivers, opmaakprogramma's, monitors, tijdanalyseprogramma's enzovoort.

Solaris *Solaris*

De Unix-implementatie van de firma Sun voor haar Sun-werkstations en PC's.

source address *bronadres*

Het adres van een eenheid of een geheugenlocatie van waaruit gegevens worden getransporteerd.

source area *brongebied*

1. Een werkgebied in een programma waarin gegevens kunnen worden bewaard of opgeslagen.
2. De geheugenruimte van de verwerkingseenheid die ten behoeve van de gegevensoverdracht aan elke taak wordt toegekend.

source code *broncode*

1. Computerinstructies en gegevens die in een vorm worden uitgedrukt die geschikt is als invoer voor een assembleerprogramma, compiler, of ander vertaalprogramma. Opmerking: Een bronprogramma bestaat uit broncode.
2. De programmeertaal zoals door de programmeur wordt gebruikt. Deze code wordt door een assembleer- of compileerprogramma omgezet in machinetaal.

source code control system: SCCS *broncodebeheerssysteem*

Een populair systeem voor het beheer van code in Unix-systemen.

source code generator *broncodegenerator*

Een gereedschap dat de software-eisen en/of -ontwerpen gebruikt om automatisch broncode te genereren. Een applicatiegenerator genereert complete applicaties, terwijl een broncodegenerator broncodes van een beperkte omvang genereert. De term is synoniem met codegenerator.

source data *1. basisgegevens; 2. brongegevens*

1. De gegevens die door een computergebruiker worden aangeleverd of aan een gebruiker worden verstrekt.
2. De gegevens die in een bronprogramma of een bronmodule zijn opgenomen of die zich op een bepaald moment in het brongebied van een programma bevinden.

source file *bronbestand*

Een bestand dat bronopdrachten bevat voor programma's in hogere programmeertalen en gegevensomschrijvingsspecificaties.

source language *brontaal*

De taal waarin de invoer voor een computer-ondersteund vertaalproces wordt uitgedrukt.

source library *bronbibliotheek*

Een bibliotheek waarvan alle onderdelen in een brontaal zijn vastgelegd.

source listing *1. uitdraai van bronopdrachten; 2. afdruk van bronopdrachten*

Een gedeelte van een afdruk of uitdraai van een compileerprogramma dat bronopdrachten bevat. Soms bevat een dergelijke afdruk eveneens diagnostische informatie.

source module *bronmodule*

De bronopdrachten die de invoer vormen voor een bepaalde uitvoering van een vertaalprogramma.

source program *bronprogramma*

1. Een computerprogramma dat moet worden gecompileerd, geassembleerd, of op andere wijze moet worden vertaald om door een computer te kunnen worden uitgevoerd.
2. Een programma dat in een brontaal is geschreven.

source statement *bronopdracht*

1. Een opdracht die in een bepaalde programmeertaal is geschreven.
2. Zie opdracht (statement).

special character *speciaal teken*

1. Een grafisch teken in een tekenverzameling dat geen letter, cijfer of spatieteken is.
2. Een teken dat niet numeriek of alfabetisch is.

specialization *specialisatie*

Een klasse x is een specialisatie van een klasse y indien x als een directe of indirecte overerving van y is gedefinieerd.

specification *specificatie*

1. Bij het ontwikkelen van systemen verwijst de term naar een beschrijving van hoe het ontwerp van een systeem, apparaat of programma moet worden geïmplementeerd.
2. Een beschrijving van de functie van een component in termen van de uitvoerwaarden die bij bepaalde invoerwaarden behoren,

- terwijl de verwerking van die invoerwaarden onder zekere vooraf bekende en/of gedefinieerde condities plaats vindt.
3. Een document dat volledig en nauwkeurig de eisen, het ontwerp, het gedrag, of andere kenmerken en eigenschappen van een systeem of component specificeert. Het bevat vaak eveneens de procedures die nodig zijn voor het kunnen bepalen indien aan de eisen die in de specificatie zijn vervat, wordt voldaan of kan worden voldaan.

specification and design language: SDL *specificatie- en ontwerp-taal*

Een taal die door de vroegere CCITT is gedefinieerd. De taal moet als een gereedschap voor ondubbelzinnige specificaties en beschrijvingen van het gedrag van telecommunicatiesystemen worden beschouwd. Het toepassingsgebied omvat verder de procesbesturing en real-time applicaties. SDL heeft naast de puur tekstuele verschijningsvorm eveneens een grafisch equivalent. Een systeem wordt in SDL gedefinieerd als een verzameling onderling met elkaar verbonden abstracte machines. Dit is een uitbreiding van het model van de eindige-toestand machine (finite state machine of FSM).

specification change notice: SCN *specificatiewijzigingsnotitie*

Een document dat in het configuratiebeheer wordt gebruikt om wijzigingen in de specificatie voor te stellen, door te geven, te distribueren, en vast te leggen.

specification language *specificatietaal*

Een taal die meestal een machine-verwerkbare combinatie van een natuurlijke en een formele taal is en die wordt gebruikt om daar de eisen, het ontwerp, het gedrag of andere kenmerken en eigenschappen van een systeem of component in uit te drukken.

specification statement *specificatie-opdracht*

Een opdracht die het compileerprogramma van informatie voorziet over de gegevens die in het bronprogramma worden gebruikt. Bovendien verschaft deze opdracht het compileerprogramma de informatie die nodig is om geheugenruimte voor deze gegevens te reserveren.

specification tree *specificatieboom*

Een schema of diagram dat alle specificaties en relaties tussen die specificaties voor een bepaald systeem weergeeft of uitbeeldt.

specific markup *specifieke opmaak*

Bij de vervaardiging van gecomputeriseerde documenten verwijst de term naar een methode waarbij opmaakopdrachten aan de tekst van het document worden toegevoegd. Die opmaakopdrachten zorgen voor de

besturing van de opmaak van het document zoals nieuwe regels, nieuwe bladzijden, het centreren van de tekst enzovoort.

specified input *gespecificeerde invoer*

Een invoer waarvoor de specificatie een bepaald resultaat voorspelt.

spiral model *spiraalvormig model*

Een model van het software-ontwikkelingsproces waarin de constituerende activiteiten, zoals de eisenanalyse, het voorlopige en gedetailleerde ontwerp, de codering, de integratie en het testen op iteratieve wijze worden uitgevoerd totdat de software compleet is.

spool (to) *spoolen*

Invoergegevens inlezen of uitvoergegevens wegschrijven naar hulpgeheugens of het hoofdgeheugen ten behoeve van een latere verwerking of latere uitvoer. Dit stelt invoer/uitvoereenheden in staat tegelijkertijd de verwerking van andere karweien uit te kunnen voeren. De term is afgeleid van het acroniem SPOOL, dat Simultaneous Peripheral Output On Line betekent.

spooler *spooler*

Een programma dat het spool-proces initieert en bestuurt.

stack *stapel*

Een lijst die op een zodanige wijze wordt samengesteld en bijgehouden dat het eerstvolgende gegeven, dat zal worden opgezocht en verwijderd, het gegeven is dat het laatst aan die lijst (of stapel) is toegevoegd.

stand-alone *alleenstaand*

De term heeft betrekking op hardware of software die hun functies uit kunnen voeren zonder met andere componenten of systemen te zijn gekoppeld of daarmee te zijn verbonden. Een voorbeeld van een dergelijk systeem is een tekstverwerkend systeem.

standard data format *standaard gegevensformaat*

De fysieke of logische grootte van een record in het geheugen. De lengte in het standaard-gegevensformaat wordt uitgedrukt in het aantal bytes dat door een record wordt ingenomen en niet noodzakelijkerwijs het aantal tekens. Dit omdat sommige tekens een gehele byte aan geheugenruimte innemen en andere tekens minder ruimte behoeven.

standard generalized markup language: SGML *SGML*

Een standaard van de ISO, ANSI en ECMA waarin de wijze waarop tekstuele documenten van annotaties en markeringen worden voorzien, is gedefinieerd. De markeringen en annotaties specificeren de

informatie betreffende de representatie of verschijningsvorm van een document.

standards *1. standaarden; 2. normen*

- a. Verplichte eisen die moeten worden opgevolgd en die een gedisciplineerde en uniforme benadering van de software-ontwikkeling afdwingen. In feite kunnen verplichte conventies en praktijken standaarden of normen worden genoemd.
- b. De software-ontwikkelaars worden met een grote verscheidenheid aan standaarden en/of normen geconfronteerd. Die standaarden kunnen worden opgedeeld in nationale, internationale, industriële, en de-facto standaarden, alsmede bedrijfsgebonden standaarden. Een groot aantal lichamen verzorgt en distribueert die standaarden zoals: ISO, ANSI, ECMA, IEEE, CEN, NNI, OSF enzovoort.

standard system action *standaard systeemactie*

Een actie die door de programmeertaal wordt gespecificeerd en die moet worden genomen bij afwezigheid van bepaalde condities.

standby redundancy *standby redundantie*

Het gebruik van redundante elementen in fout-tolerante technieken of praktijken. Deze elementen zijn niet-operationeel zolang er geen storing in een primair element optreedt.

starting address *beginadres*

Het adres van de eerste instructie van een computerprogramma in het hoofdgeheugen. Opmerking: Dit adres kan al dan niet hetzelfde adres zijn als dat van de oorsprong van het programma. Dit hangt af van het feit of het programma gegevens bevat die aan de eerste instructie voorafgaan.

state *1. status; 2. conditie; 3. toestand*

- 1a. De waarden die door variabelen op een gegeven moment worden aangenomen en die de karakteristieke kenmerken of eigenschappen van een systeem, component of simulatie definiëren.
- 1b. Informatie betreffende de geschiedenis van eerder gedeponeerde verzoeken die nodig is om het gedrag van toekomstige verzoeken te kunnen bepalen.
- 1c. Eén van de mogelijke waarden die een gegevenselement aan kan nemen.
2. Voorwaarde waaraan moet worden voldaan.
- 3a. Een toestand waarin een systeem, component of proces zich kan bevinden.
- 3b. Toestand waarin iets (zoals een systeemcomponent of gegevenselement) zich bevindt.

state chart diagram *toestandsschemadiagram*

Objecten bezitten zowel eigenschappen als toestanden; dat wil zeggen, objecten voeren acties uit en bevatten kennis. Bepaalde objecten kunnen meer acties uitvoeren en hebben meer kennis van (gecompliceerde) zaken dan andere objecten. Bepaalde objecten zijn erg gecompliceerd en toestandsschemadiagrammen helpen de werking van dit soort objecten verduidelijken.

state consistency *statusconsistentie*

Conditie die moet garanderen dat de status die met een object is geassocieerd overeenkomt met het gegevensmodel van het object.

state data *toestandsgegevens*

Gegevens die een interne toestand van de testeenheid definiëren en die worden gebruikt om die toestand vast te stellen of die toestand te vergelijken met andere toestanden.

state diagram *toestandsschema*

1. Een schema of diagram dat de toestanden die een systeem of component aan kan nemen, uitbeeldt. Het toont verder de gebeurtenissen of omstandigheden die het gevolg zijn van de overgang van de ene toestand in een andere.
2. Zie toestandsovergangsdigram (state transition diagram).

state integrity *statusintegriteit*

De term verwijst naar de eis dat de status die met een object is geassocieerd niet door externe gebeurtenissen wordt gecorrumpeerd.

statement *opdracht*

1. In een programmeertaal verwijst de term naar een zinvolle reeks woorden die bewerkingen kan omschrijven of specificeren.
2. Bij het programmeren is dit een rangschikking van symbolen.
3. Een syntactisch geldige combinatie van woorden en symbolen die in de procedure-afdeling is geschreven. Dit is een combinatie van gereserveerde woorden en door de programmeur omschreven opdrachtelelementen.
4. De basiseenheid van een programma die uit één of meer regels bestaat en die een combinatie bevat van namen, opdrachtelelementen, constanten of woorden, waarvan de betekenis in het compileerprogramma is vastgelegd.
5. Een basiselement dat wordt gebruikt om een gedeelte van het programma te begrenzen, om de in het programma gebruikte namen te omschrijven, of om een te verrichten handeling te specificeren.

statement body *opdrachtinhoud*

Het gedeelte van een opdracht dat volgt op het opdrachtidentificatiesymbool en dat door een puntkomma wordt afgesloten.

statement coverage *opdrachtdekking*

Het percentage uitvoerbare opdrachten in een component dat door een test of een testserie wordt uitgevoerd en gecontroleerd.

statement function *opdrachtfunctie*

Een opdrachtfunctie die wordt gedefinieerd door een naam, gevolgd door een lijst met dummy argumenten, gevolgd door een is-gelijk-teken (=), gevolgd door een logische of rekenkundige uitdrukking.

statement identifier *opdrachtidentificatiesymbool*

Het sleutelwoord dat het doel en eventueel de functie van de opdracht aangeeft.

statement testing *testen van opdrachten*

Een techniek voor het vervaardigen van testen of testseries die de afzonderlijke opdrachten van een computerprogramma uitvoeren.

state modifying request *status-modificatieverzoek*

Een verzoek dat bij de uitvoering van een dienst het resultaat van toekomstige verzoeken doet wijzigen.

state transition *toestandsovergang*

Een overgang tussen twee toegestane toestanden van een systeem of een component.

state transition diagram *toestandsovergangsdigram*

Een diagram dat uit cirkels bestaat waarmee toestanden worden voorgesteld en lijnstukken die die cirkels verbinden en die de overgangen tussen de toestanden representeren. Iedere overgang kan met één of meer acties worden geassocieerd. Het diagram stelt een eindige-toestandsmachine (FSM) voor.

state transition testing *testen van toestandsovergangen*

Een techniek waarmee testen of testseries kunnen worden vervaardigd die toestandsovergangen uitvoeren en op hun juistheid controleren.

state variable *toestandsvariabele*

Deel van de status of toestand van een object.

static *statisch*

De term heeft betrekking op een gebeurtenis die of een proces dat respectievelijk optreedt of plaats vindt zonder dat er van de uitvoering van een computerprogramma sprake is; bijvoorbeeld statische analyse of statische binding.

static analysis *statische analyse*

De analyse van een systeem, component of programma zonder dat bij die analyse het systeem of component actief zijn of het programma wordt uitgevoerd. De analyse is gebaseerd op de vorm, structuur, inhoud of documentatie van het systeem, component of programma.

static analyzer *statisch analysegereedschap*

Een gereedschap dat een statische analyse uitvoert.

static binding *statische binding*

Een binding die plaats vindt voordat een computerprogramma wordt uitgevoerd en niet onderworpen is aan enige wijziging gedurende de uitvoering van dat programma.

static breakpoint *statisch breekpunt*

Een breekpunt dat tijdens de compilatie kan worden gespecificeerd zoals de ingang van een bepaalde routine.

static class *statische klasse*

Dit is een andere naam voor klassevariabele.

static dump *statische dump*

Een dump die voor of na de uitvoering van een computerprogramma wordt geproduceerd.

static error *statische fout*

Een fout die onafhankelijk is van de tijdsgebonden aard van een invoergegeven.

static method *statische methode*

Andere benaming voor klassemethode.

statistical testing *statistisch testen*

Een ontwerptechniek voor testen en testseries waarin een model van de statistische verdeling van de invoer wordt gebruikt om representatieve testen en testseries te construeren.

status code *statuscode*

Een code die wordt gebruikt om de resultaten van de werking van een computerprogramma aan te geven. Bijvoorbeeld, een code waarmee een overdracht (carry), een overloop (overflow) of een pariteitsfout wordt aangegeven.

step-by-step operation *stapsgewijze bewerking*

Zie enkelvoudige bewerking (single-step operation).

stepwise refinement *stapsgewijze verfijning*

Een software-ontwikkelingstechniek waarin gegevens en bewerkingstappen eerst vrij ruw en daarna met een toenemende mate van gedetailleerdheid worden gedefinieerd.

stop (to) *stoppen*

De verwerking of uitvoering van een computerprogramma beëindigen.

storage *geheugen*

Elk medium waarin of waarop gegevens kunnen worden opgeslagen en bewaard. Dit geschiedt over het algemeen op een zodanige wijze dat men op een later tijdstip weer over deze gegevens kan beschikken.

storage allocation *geheugenallocatie*

Een aspect van de toekenning of allocatie van computerelementen aan gedefinieerde bewerkingen of verwerkingen. Dit bestaat uit het toewijzen van geheugengebieden aan specifieke karweien en het uitvoeren van aan die toewijzing verwante procedures zoals het transport van gegevens tussen hulpgeheugen(s) en hoofdgeheugen en vice versa.

storage capacity *geheugencapaciteit*

Het maximale aantal items dat in een gegeven geheugeneenheid kan worden opgeslagen. Dit wordt meestal uitgedrukt in bytes of veelvouden van bytes (kilobytes, megabytes enzovoort).

storage efficiency *geheugenefficiëntie*

De mate waarin een systeem of component de daaraan toegekende functies uit kan voeren met een minimaal gebruik (verbruik) van de beschikbare geheugenruimte.

storage location *geheugenpositie*

Een gebied in een geheugeneenheid.

storage testing *testen van opslagmogelijkheden*

Het uitvoeren van testen die ten doel hebben om na te gaan in hoeverre een systeem aan de gespecificeerde eisen ten aanzien van de opslag van gegevens en code voldoet.

store (to) *opslaan*

1. Gegevens in een geheugenruimte of opslagmedium plaatsen.
2. Computerinstructies of gegevens vanuit een register naar een intern geheugen of vanuit een intern geheugen naar een extern geheugen kopiëren.

straight-line code *lineaire code*

Een aantal opeenvolgende computerinstructies waarin geen lussen voorkomen.

straight-line coding *lineaire codering*

Een programmeringstechniek waarin lussen worden vermeden door expliciet alle instructies die in elke lus voor kunnen komen, volledig (opdracht na opdracht) te coderen.

stratified language *gelaagde taal*

Een taal die niet als haar eigen metataal kan worden gebruikt. Voorbeelden hiervan zijn onder andere Fortran en Cobol.

streams *stromen*

Routines die bewerkingen op (continue) stromen gegevens van een bepaald type uit kunnen voeren.

stress testing *testen van grenzen*

Testen die worden uitgevoerd om te controleren in hoeverre een systeem of een component de grenzen van de gespecificeerde eisen heeft bereikt en/of heeft overschreden.

string *rij*

Een lineaire rangschikking van elementen, zoals tekens of bits.

string oriented symbolic language: SNOBOL *Snobol*

Een programmeertaal die is ontworpen om rijen te verwerken en patronen met elkaar te vergelijken.

string variable *rijvariabele*

Een variabele die van een bepaald kenmerk (zoals een bit-kenmerk of een teken-kenmerk) is voorzien en waarvan de waarde uit bitrijen of tekenrijen kan bestaan.

strong typing *sterke gegevenstypering*

Een kenmerk van bepaalde programmeertalen waarin het type van elk gegevensitem moet worden gespecificeerd (gedecclareerd). Dit sluit het gebruik van bewerkingen op onjuiste of niet-geldige gegevenstypen uit en voorkomt de interactie tussen niet-compatibele gegevenstypen.

structural coverage *structurele dekking*

Maatregelen die gebaseerd zijn op de interne structuur van het component en die betrekking hebben op de dekking (van bijvoorbeeld fouten) van dat component.

structural test case design *test(serie)ontwerp voor structureel testen*

De keuze van testen en testseries die gebaseerd is op een analyse van de interne structuur van het component.

structural testing *structureel testen*

1. Testen van systemen of componenten waarbij rekening wordt gehouden met het interne mechanisme van die systemen of componenten.
2. Zie ook test(serie)ontwerp voor structureel testen (structural test case design).

structure *structuur*

Een hiërarchische reeks namen die verwijst naar een combinatie van gegevensbestanddelen die eventueel verschillende kenmerken en eigenschappen kunnen bezitten.

structure chart *structuurkaart*

Een schema of diagram dat modules, activiteiten, of andere entiteiten in een systeem of computerprogramma uitbeeldt en dat laat zien hoe grotere en meer algemene entiteiten worden opgedeeld in kleinere, en meer specifieke entiteiten. Opmerking: Het resultaat is niet noodzakelijkerwijs hetzelfde als dat wat in een oproepschema wordt uitgebeeld.

structure clash *structuurconflict*

Een situatie die in een software-ontwerp op kan treden en waarin een module twee of meer gegevensverzamelingen moet verwerken of waarin twee of meer gegevensverzamelingen voorkomen waarvan de gegevensstructuren onderling niet-compatibel zijn.

structured analysis *gestructureerde analyse*

Eén van de vele analysemethoden die in de software-ontwikkeling wordt gebruikt om de eisen van de software te bepalen. In deze fase van de software-ontwikkeling wordt gebruik gemaakt van gegevensstroomdiagrammen, gegevensdictionaries en procesbeschrijvingen voor de analyse en de representatie van een eis die aan de software wordt gesteld of die daaraan ten grondslag ligt.

structured and open environment: STONE *gestructureerde en open omgeving*

Een project dat wordt gesteund door het Duitse Ministerie van Onderzoek en Technologie (BMFT) en dat de ontwikkeling, de implementatie en de distributie van een software-ontwikkelingsomgeving voor onderzoek en onderwijs tot doel heeft.

structured basic testing *fundamenteel structureel testen*

Het ontwerpen van testen en testseries waarin deze worden afgeleid van de logica in de code met het doel een 100% dekking van de mogelijke fouten in de code te verkrijgen.

structured design *gestructureerd ontwerp*

1. Elke gedisciplineerde benadering van het ontwerpen van software die voldoet of zich conformeert aan de gespecificeerde regels die gebaseerd zijn op de principes van modulariteit, top-down ontwerp, en de stapsgewijze verfijning van gegevens, systeemstructuren en verwerkings- of bewerkingsstappen.
2. Het resultaat van het toepassen van de benadering die in het voorgaande punt van deze definitie is omschreven.
3. Eén van de vele systematische technieken in de software-ontwikkeling die van boven naar beneden (top-down) wordt uitgevoerd. Het gestructureerde ontwerp volgt (onmiddellijk) op de gestructureerde analyse.

structured program *gestructureerd programma*

1. Een computerprogramma dat met behulp van een aantal fundamentele besturingsstructuren is geconstrueerd en waarin elke structuur één ingang en één uitgang heeft. De besturingsstructuren zijn: twee of meer opeenvolgende opdrachten, een conditionele keuze uit één of meer opdrachtenreeksen, en de herhaling van een opdrachtenreeks.
2. Een hiërarchie van modules die allen één ingangspunt en één uitgangspunt hebben; de besturing wordt door de structuur naar beneden geleid zonder onvoorwaardelijke sprongen naar de hogere niveaus van de structuur.

structured programming *gestructureerde programmering*

1. Elke software-ontwikkelingstechniek waarin het gestructureerde ontwerp wordt toegepast en die resulteert in de ontwikkeling van gestructureerde programma's.
2. Een techniek om programma's te organiseren en te coderen waardoor de complexiteit wordt gereduceerd en de duidelijkheid wordt verbeterd. Op deze wijze kunnen fouten gemakkelijker worden opgespoord en wijzigingen eenvoudiger aangebracht. Ook creëert het gebruik van deze techniek de mogelijkheid de programma's op hun correctheid te bewijzen.

structured programming language *gestructureerde programmeertaal*

Een programmeertaal die de programmeur voorziet van gestructureerde programmaconstructies zoals sprongen, lussen (loops) en program-masegmenten met enkelvoudige ingangen en enkelvoudige uitgangen. Het verschaft de programmeur verder een groot aantal andere faciliteiten voor de ontwikkeling van gestructureerde programma's.

structured query language: SQL *gestructureerde ondervraagtaal*

Een ISO- en ANSI-standaard die als gebruikersinterface voor relationele databasebeheerssystemen fungeert.

structured walkthrough *gestructureerde programma-inspectie*

Zie programma-inspectie (walkthrough).

structure member *structuurdeel*

Een willekeurige secundaire structuur van elementaire namen binnen een structuur.

structure of arrays *structuur van reeksen*

Een structuur die reeksen bevat. Deze worden gespecificeerd door het benoemen van de afzonderlijke samenstellende delen.

stub *stub*

1. Een skeletachtige of zeer specifieke implementatie van een programmamodule die wordt gebruikt om een component te ontwikkelen of te testen.
2. Een opdracht in een computerprogramma die een softwaremodule die op een andere plaats is of wordt gedefinieerd, doet vervangen.

subarray *1. sublijst; 2. deeltabel*

Een lijst of tabel die binnen een andere lijst of tabel is gedefinieerd.

subclass *deelklasse*

Een klasse die van een andere specifieke klasse is afgeleid. Tussen de deelklasse en de klasse waarvan deze is afgeleid, kunnen andere deelklassen bestaan.

subdomain *deeldomein*

Een domein dat als een deel van een groter domein kan worden beschouwd.

sublanguage *deeltaal*

Eén van de talen die met een databasebeheerssysteem is geassocieerd zoals een gegevensdefinitietaal, een ondervraagtaal of een gegevensmanipulatietaal.

subparameter *deelparameter*

Eén van de variabele informatie-eenheden die op een sleutelwoordparameter volgt. De betekenis van een deelparameter wordt door de positie of aan de hand van een sleutelwoord bepaald.

subpath *deelpad*

Een serie opeenvolgende uitvoerbare opdrachten in een component.

subprogram *deelprogramma*

1. Een programma dat door een ander programma wordt opgeroepen.
2. Een programma-eenheid die deel uitmaakt van een functie, een subroutine, een procedure enzovoort.

3. Een programma dat afzonderlijk kan worden gecompileerd en uitgevoerd. Opmerking: De termen 'routine', 'deelprogramma' en 'subroutine' worden in verschillende programmeertalen afwijkend gedefinieerd.

subroutine *subroutine*

1. Een opeenvolgende reeks opdrachten die in één of meer programma's en op één of meer punten binnen een programma kan worden gebruikt. Door het gebruik van subroutines bespaart men geheugenruimte; dit gaat wel ten koste van de uitvoeringssnelheid.
2. Een programmasegment dat deel uit kan maken van een ander programmasegment.
3. Een procedure die met behulp van een bepaalde opdracht kan worden opgeroepen.
4. Een routine die de besturing aan het programma of deelprogramma dat het heeft opgeroepen, teruggeeft.

subroutine call *subroutine-oproep*

De constructie die als onderdeel van een programma, de verbinding met een gesloten subroutine verzorgt.

subroutine statement *subroutine-opdracht*

Een opdracht die wordt gebruikt om een naam aan een subroutine toe te kennen.

subroutine trace *subroutinetrace*

Een verslaggeving van alle of van geselecteerde subroutine- of functie-oproepen die tijdens de uitvoering van een computerprogramma zijn uitgevoerd, en eventueel de waarden van de parameters die aan de subroutine of functie zijn doorgegeven en door elke subroutine of functie zijn teruggegeven aan het oproepende programma.

subscript *onder-index*

1. Een symbool dat met de naam van een verzameling is verbonden om de identiteit van een bepaalde deelverzameling of een element te kunnen bepalen.
2. Een integer getal of een variabele waarvan de waarde naar een bepaald element in een tabel verwijst.
3. Een grootheid of een reeks grootheden die tussen haakjes staat en die wordt gebruikt in combinatie met een naam van een reeks om de identiteit van een element uit die reeks te kunnen bepalen.
4. Een elementaire uitdrukking die een positie binnen de dimensie van een reeks specificiert. Dit kan ook een sterretje zijn, in welk geval het de totale omvang van de dimensie van de reeks specificiert.

subset *deelverzameling*

1. Een verzameling binnen een grotere verzameling waarvan elk element een element van die grotere verzameling is.
2. Een afwijkende vorm van een programmeertaal die minder mogelijkheden of meer beperkingen heeft dan de oorspronkelijke programmeertaal.

subsystem *deelsysteem*

Een secundair of ondergeschikt systeem dat meestal onafhankelijk van en asynchroon met een besturend systeem werkt.

subtask *deeltaak*

Een taak waarvan de werking wordt gestart en beëindigd door een taak van een hoger niveau.

subtype *deeltipe*

1. Een deelverzameling (subset) van een gegevenstype die wordt verkregen door de verzameling van de mogelijke waarden van het gegevenstype te beperken.
2. Indien type x type y uit doet breiden of type y implementeert, dan is type x een deeltipe van y .

suffix notation *suffixnotatie*

Een methode om wiskundige uitdrukkingen te vormen waarbij elk bewerkingsteken wordt voorafgegaan door de bijbehorende opdracht-elementen en de bewerking aangeeft die op deze opdrachtelelementen of de tussenresultaten die er aan voorafgaan, moet worden verricht.

suite *1. serie; 2. reeks*

De term verwijst naar een verzameling programma's, functies, bewerkingen of bewerkingstappen die met een verzamelnaam kunnen worden geïdentificeerd, bijvoorbeeld testserie of testreeks (testsuite).

superclass *superklasse*

Een klasse waarvan een bepaalde klasse is afgeleid. Er kunnen één of meer klassen tussen de superklasse en de afgeleide klasse bestaan.

superscript *boven-index*

Eén of meer tekens die een halve regel boven de normale (afdruk)-regel zijn geplaatst en eveneens worden afgedrukt.

supertype *supertipe*

De supertypes van een type zijn alle interfaces en klassen die door dat type zijn uitgebreid of daardoor zijn geïmplementeerd.

superuser *supergebruiker*

De gebruikersnaam (die ook bekend staat als wortel) waarmee men allerlei gevaarlijke manipulaties binnen en met het systeem uit kan voeren zoals het creëren van nieuwe gebruikersnamen en het installeren van hardware en software. De systeemadministrateur is eigenlijk de enige die als supergebruiker in zou moeten kunnen loggen.

supervisor *supervisieprogramma*

Zie supervisieprogramma (supervisory program).

supervisor state *toestand van het supervisieprogramma*

De toestand waarbij het supervisieprogramma tijdens de werking van een computersysteem wordt uitgevoerd. Deze toestand heeft meestal een grotere prioriteit dan de applicatieprogramma's en schort de werking daarvan op.

supervisory program *supervisieprogramma*

Een computerprogramma dat meestal deel uitmaakt van het besturingssysteem en dat de uitvoering of werking van andere computerprogramma's regelt of bestuurt en het werkschema in een computersysteem regelt.

support *ondersteuning*

De activiteiten die nodig zijn voor de correcte werking van een operationeel systeem of component en dat er zorg voor draagt dat het systeem of component voldoet aan de oorspronkelijk opgestelde eisen en alle daarna gespecificeerde modificaties op die eisen.

support domain *ondersteuningsdomein*

De term is synoniem met die van horizontaal domein (horizontal domain).

support manual *ondersteuningshandleiding*

Een document dat de informatie bevat die nodig is om een operationeel systeem of component gedurende de gehele levenscyclus van dat systeem of component te ondersteunen en te onderhouden. In deze handleiding worden de hardware en software beschreven die een systeem of component constitueren, alsmede de procedures voor het onderhoud, de reparatie, of een noodzakelijk geachte herprogrammering.

support software *ondersteuningssoftware*

Software die kan worden gebruikt bij de ontwikkeling of het onderhoud van andere software. Bijvoorbeeld, compilers, laadprogramma's, en andere utiliteitsprogramma's.

swap *verwisseling*

De verwisseling van de inhoud van twee geheugengebieden. Dit is meestal een verwisseling van een geheugengebied binnen het hoofdgeheugen met een geheugengebied in een hulpgeheugen.

swap (to) *verwisselen*

Een verwisseling, zoals vermeld in de vorige definitie, uitvoeren.

swing set *schommelserie*

Een verzameling van alle componenten van grafische gebruikersinterfaces in en voor Java. Dit functioneert op uniforme wijze op elk platform dat een virtuele Java-machine ondersteunt.

switch *keuzeschakelaar*

1. De term heeft betrekking op het argument van een opdracht in, bijvoorbeeld, het Unix-besturingssysteem.
2. Zie optie (option).

syllable *woorddeel*

Een tekenrij of een rij binaire elementen in een woord.

symbol *symbool*

1. Een willekeurige groep alfanumerieke tekens.
2. Een teken dat of tekenrij die adressen of willekeurige waarden resp. voor kan of voor kunnen stellen. Een symbool zal vaak uit een maximaal aantal tekens bestaan.

symbolic address *symbolisch adres*

Een adres dat in symbolen (namen en labels) is uitgedrukt. De symbolen zijn gemakkelijk door de programmeur te hanteren. Een symbolisch adres wordt tijdens het vertalen van een programma omgezet in een door de computer te gebruiken absoluut adres.

symbolic addressing *symbolische adressering*

Een adresseringsmethode waarbij het adresgedeelte van een instructie een symbolisch adres bevat.

symbolic coding *symbolische codering*

Het vervaardigen van programma's en programmasegmenten door gebruik te maken van machine-instructies die symbolische adressen bevatten.

symbolic execution *symbolische uitvoering*

1. Een software-analysetechniek waarin de uitvoering van een programma met behulp van symbolen (variabele namen in plaats van actuele waarden van de invoergegevens) wordt gesimuleerd en de uitvoergegevens van het programma als logische of

wiskundige uitdrukkingen, waarin deze symbolen voorkomen, worden uitgedrukt.

2. Een techniek die in de statische analyse wordt gebruikt en die een symbolische uitdrukking voor de paden (takken) in een programma genereert.

symbolic file *symbolisch bestand*

Een bestand in een toepassingsprogramma dat, bij het afzonderlijk uitvoeren van het programma, naar verschillende bestanden kan verwijzen. Deze bestanden zijn als fysieke bestanden bekend. Een symbolisch bestand wordt door middel van een bestandsopdracht met een fysiek bestand geassocieerd.

symbolic language *symbolische taal*

Een programmeertaal waarvan de instructies in gemakkelijk te hanteren symbolen in plaats van in de machinetaal zijn uitgedrukt. Een voorbeeld van een symbolische taal is de assembleertaal.

symbolic link *symbolische koppeling*

Zie zachte koppeling (soft link).

symbolic name *symbolische naam*

In een programmeertaal is dit een unieke naam die wordt gebruikt om een entiteit, zoals een veld, record, bestand, gegevensstructuur of label voor te stellen.

symbolic parameter *symbolische parameter*

1. Een variabel symbool dat in een macrodefinitie wordt gedefinieerd. Aan een symbolische parameter wordt gewoonlijk de waarde toegekend van het overeenkomstige opdrachtelemeent in de macro-instructie die de macrodefinitie oproept.
2. Een symbool dat vooraf wordt gegaan door een en-teken en dat een parameter of de waarde voorstelt die aan een parameter in een gecatalogiseerde of een zich in de invoerstream bevindende procedure wordt toegekend. De waarden worden toegekend aan symbolische parameters wanneer de procedure, waarin ze voorkomen, wordt opgeroepen.

symbolic trace *symbolische naspeuring*

Een vastlegging van de resultaten van de bronopdrachten en sprongen tijdens de uitvoering van een computerprogramma waarbij symbolische waarden, in plaats van de actuele waarden van de invoergegevens, worden gebruikt.

symbol substitution *symboolsubstitutie*

Het vervangen van een symbool door een tekenrij.

symbol table *symbolentabel*

Een tabel waarin de in een programma gebruikte symbolen, hun adressen, waarden, attributen en betekenissen zijn vastgelegd.

syntactic error *syntactische fout*

Een foutief gebruik van de structurele of grammaticale regels die voor een taal zijn gedefinieerd.

syntax *syntax(is)*

1. De structurele en grammaticale regels die de definities bevatten van de wijze waarop tekens en symbolen in een taal tot woorden, zinnen en uitdrukkingen en andere toegestane constructies worden gecombineerd, alsmede de wijze waarop deze worden geïnterpreteerd en gebruikt.
2. De grammaticale regels die aan de structuur en de grammatica van een programmeertaal ten grondslag liggen.

syntax checker *syntax-controleprogramma*

Een programma dat bronopdrachten in een bepaalde programmeertaal op fouten in de syntax van die taal controleert.

syntax language *syntaxtaal*

Een metataal die wordt gebruikt om de syntax van een andere taal te specificeren of te omschrijven.

syntax testing *testen van de syntax*

Een techniek voor de ontwikkeling van testen en testseries voor een component of een systeem. De testen zijn bestemd om de syntax van de ingevoerde gegevens op hun juistheid te controleren.

system *systeem*

1. Een combinatie van mensen, machines en methoden waarmee bepaalde functies kunnen worden verricht.
2. Een combinatie van methoden, procedures en/of technieken die door een geordende wisselwerking een zinvol geheel vormt.
3. Componenten, modules of assets die zodanig zijn georganiseerd dat daarmee een bepaalde functie of serie functies kan worden gerealiseerd.
4. Een serie onderling verwante taken waarin mensen, methoden, hardware, software en/of firmware een rol spelen en die in combinatie voorkomen en functioneren voor de realisatie van een gespecificeerd doel.

system design review: SDR *systeemontwerpbeoordeling*

Een beoordeling die wordt uitgevoerd om (a) de wijze waarop de systeemeisen aan configuratie-items zijn toegewezen, (b) het systeem-ontwikkelingsproces dat die toewijzingen heeft geproduceerd, (c) de ontwikkelingsplanning voor de eerstvolgende fase van het ontwikkelingsproces, (d) de consideraties betreffende de

fabricage, en (e) de planning voor de fabricagevoorbereiding te evalueren.

system development cycle *stysteemontwikkelingscyclus*

De tijdsperiode die aanvangt met het besluit een systeem te ontwikkelen en eindigt op het moment dat het systeem wordt opgeleverd. Opmerking: Deze term wordt soms gebruikt om een langere tijdsperiode aan te geven namelijk òf de periode waarop het systeem niet langer wordt onderhouden en/of uitgebreid òf de gehele duur van de levenscyclus.

system library *steeembibliotheek*

Een softwarebibliotheek die software bevat die steeds deel van het systeem uitmaakt en die voor normaal gebruik kan worden geraadpleegd of via een verwijzing in andere programma's kan worden geïncorporeerd, bijvoorbeeld een macro-bibliotheek.

system life cycle *levenscyclus van een systeem*

De tijdsperiode die aanvangt met de conceptie van een systeem en eindigt op het moment dat het systeem niet langer beschikbaar is of niet langer wordt gebruikt.

system model *stysteemmodel*

Een representatiewijze van een systeem bij de evaluatie van de prestaties van een computer waarin de relaties tussen de werkbelastingen en de prestatiemetingen in het systeem worden weergegeven.

system object model: SOM *stysteem-objectmodel*

Een implementatie van CORBA van IBM.

system profile *steeemprofiel*

Een reeks metingen die bij de evaluatie van de prestaties van een computer worden gebruikt en waarin het deel van de tijd waarin de belangrijkste systeemelementen werkzaam of operationeel zijn, wordt gedeeld door de tijd dat die systeemelementen beschikbaar zijn.

system programmer *steeemprogrammeur*

1. Een programmeur die programmasystemen en andere toepassingen ontwerpt.
2. Een programmeur die een automatisatiesysteem ontwerpt, vervaardigt, onderhoudt, uitbreidt en het gebruik ervan beheert of regelt met het doel de totale productiviteit van het systeem te vergroten.

system requirements review: SRR *beoordeling van de systeemeisen*

Een beoordeling die wordt uitgevoerd om: de volledigheid en geschiktheid van de eisen die voor een systeem zijn gedefinieerd te evalueren; het systeemontwikkelingsproces dat het gevolg is van die

eisen te evalueren; de resultaten van systeemontwikkelingsstudies te beoordelen; en de systeemontwikkelingsplannen te evalueren.

system reuse *hergebruik van een systeem*

Het hergebruik van assets met behulp waarvan softwaresystemen tijdens een software-ontwikkelingsproces worden ontwikkeld. Dit proces is specifiek voor hergebruik gestructureerd. Een systematisch hergebruik betekent dat de software-ontwikkeling wordt aangestuurd door een georganiseerd gebruik van producten uit de domeinontwikkeling (zoals een domeinmodel, domeinarchitectuur, en andere assets) tijdens de opeenvolgende fasen van een software-ontwikkelingsproces.

systems analysis *systeemanalyse*

De analyse van een activiteit om na te kunnen gaan wat er tot stand moet worden gebracht en hoe dat dient te geschieden.

systems application architecture: SAA *SAA*

De familie van standaard-interfaces van de firma IBM met behulp waarvan software kan worden ontwikkeld die onafhankelijk is van de apparatuur of het besturingssysteem.

system software *systeemsoftware*

1. Software die ontwikkeld is om het operationele gebruik en het onderhoud van een computersysteem en de daarmee verwante programma's te ondersteunen; bijvoorbeeld besturingssystemen, assembleerprogramma's, utiliteitsprogramma's.
2. Programmatuur die deel uitmaakt van een computersysteem en die de wijze waarop toepassingsprogramma's worden verwerkt, specificeert en/of aanstuurt.

system testing *testen van een systeem*

Het proces waarin een (geïntegreerd) systeem wordt getest en waarin wordt nagegaan in hoeverre aan de gespecificeerde eisen wordt voldaan.

T

target language *doeltaal*

De taal waarin de opdrachten van de brontaal met behulp van een computer-ondersteund vertaalproces worden vertaald.

target machine *doelmachine*

1. De computer waarop men een programma wil gaan draaien.
2. Een computer die door een andere computer is geëmuleerd.

target program *doelprogramma*

Een programma is een doeltaal dat met behulp van een assembleertaal of een hogere orde programmeertaal is vertaald.

target variable *doelvariabele*

Een variabele waaraan een waarde wordt toegekend.

task *taak*

1. Een basis eenheid van werk die door een machine moet worden verricht.
2. Een basis eenheid van werk die door een computer moet worden verricht. Deze werkeenheid wordt gewoonlijk voor een besturingsprogramma in een bepaalde omgeving gespecificeerd.
3. Een eenheid van werk die door het supervisieprogramma wordt gebruikt om het gelijktijdig uitvoeren van het programma binnen het systeem vast te stellen en te volgen. Elke taak vertegenwoordigt een deel van de opdrachten dat door een programma of een serie programma's wordt verwerkt. Het eerste programma dat onder elke taak wordt uitgevoerd is een primair programma. Alle andere zijn secundaire programma's.
4. Een eenheid van werk voor de verwerkingseenheid die wordt gebruikt om een programma of een systeemfunctie uit te voeren.
5. De basiseenheid die zorg draagt voor het synchroon uitvoeren van het programma. Een taak bestaat uit instructies die synchroon op de programmeergegevens werken. Hoewel een taak synchroon ten opzichte van de eigen instructies wordt uitgevoerd, worden deze instructies asynchroon ten opzichte van alle andere taken in het systeem verwerkt.
6. In het ontwerp van software verwijst deze term naar een softwarecomponent dat gelijktijdig met andere softwarecomponenten kan werken.

task name *taaknaam*

Een identificatiesymbool dat wordt gebruikt om naar een taakvariabele te verwijzen.

task state *taakstadium*

In een (netwerk)besturingsprogramma van een verwerkingseenheid bestaan de taakstadia uit het actieve stadium, het gereedstadium, het onderbroken stadium en het wachtende stadium.

task states *taakstadia*

De stadia in de toestand van de uitvoering van een taak ten opzichte van de verwerkingseenheid: actief, gereed, onderbroken en in afwachting.

task variable *taakvariabele*

Een variabele die een kenmerk bezit dat die variabele met een taak associeert. De waarde van de variabele geeft de relatieve prioriteit van een taak aan.

taxonomy *taxonomie*

Een schema dat een bepaalde hoeveelheid kennis opdeelt en de relaties tussen de aldus ontstane delen definieert. Het wordt gebruikt om die kennis te classificeren en te begrijpen.

technical management *technisch management*

Het toepassen van technische en administratieve middelen voor het plannen, organiseren, beheren en besturen van de ontwikkelingsfuncties.

technical standard *technische standaard*

Technische en bestuurlijke procedures die kunnen helpen bij de evaluatie en het verbeteren van het software-ontwikkelingsproces.

template *1. mal; 2. model*

Het patroon, formaat, lay-out, of vorm van een standaard module, programma, proces, object, component dat of de (gegevens)structuur die als een hulp bij de ontwikkeling kan dienen. Een dergelijke mal kan worden gekopieerd en van details worden voorzien.

template code *malcode*

Pseudocode die door een automatisch werkend software-ontwikkelingssysteem wordt gegenereerd en de nodige handmatige aanpassingen behoeft voordat het kan worden gecompileerd.

temporal cohesion *tijdsafhankelijke cohesie*

Een soort cohesie waarin de taken die door een softwaremodule worden uitgevoerd tijdens een specifieke fase in de uitvoering van een programma, worden afgewerkt; bijvoorbeeld, een module die alle initialisatietaken van een programma bevat en verwerkt.

termination construct *beëindigingsconstructie*

Een programmaconstructie die in een pauze of een beëindiging van de werking van het programma resulteert.

test *test*

1. Een activiteit waarin een systeem of component onder gespecificeerde condities wordt uitgevoerd, waarvan de resultaten worden geobserveerd of vastgelegd, en waarin een evaluatie van bepaalde aspecten van het systeem of het component worden gemaakt.
2. Eén testproces of een serie testprocessen.
3. Eén testprocedure of een serie testprocedures.

4. Eén testproces en -procedure of een reeks testprocessen en testprocedures.

test (to) *testen*

1. Een activiteit zoals beschreven in het eerste punt van de vorige definitie uitvoeren.
2. De werking van een functionele eenheid (zoals een programma of een apparaat) evalueren en het bereikte resultaat met het gespecificeerde resultaat vergelijken teneinde de correctheid van de geëvalueerde werking vast te kunnen stellen.
3. Uitvoering van het proces waarin voor een product de verschillen tussen de verwachte en werkelijke resultaten en prestaties worden geïdentificeerd.

testability *testbaarheid*

1. De mate waarin een systeem of component het opstellen van testscriteria en de uitvoering van testen, die ten doel hebben om na te gaan of aan de criteria wordt voldaan, vergemakkelijkt.
2. De mate van het compleet zijn van de formulering van een eis. Met die formulering wordt het mogelijk testscriteria op te stellen en testen die ten doel hebben om na te gaan of aan die criteria wordt voldaan, uit te voeren.

test automation *automatisering van testen*

Het gebruik van software voor het besturen van de werking van de testen, het vergelijken van de werkelijke resultaten met de voorspelde resultaten, het instellen van bepaalde aan de testen voorafgaande condities, en andere regelende en rapporterende functies.

test case *testproces*

1. Een serie invoergegevens die voorafgaande aan de test zijn bepaald of geformuleerd, alsmede de verwachte of voorspelde resultaten van de test(en). Testprocessen worden ontwikkeld voor een speciaal doel, zoals het controleren en/of onderzoeken van een bepaald pad dat het programma volgt of om na te gaan of het programma conform opgestelde specificaties werkt.
2. Documentatie waarin invoergegevens, voorspelde resultaten en een reeks bewerkingscondities voor een testitem zijn gespecificeerd.

test case design technique *ontwikkelingstechniek voor testprocessen*

Een methode die wordt gebruikt om testprocessen op te stellen, af te leiden, of te selecteren.

test case generator *testprocesgenerator*

Een softwaregereedschap dat als invoer broncode, testscriteria, specificaties, of definities van gegevensstructuren accepteert; deze ingevoerde gegevens gebruikt om testinvoergegevens te genereren; en soms verwachte resultaten bepaalt.

test case specification *testproces-specificatie*

Een document dat de invoergegevens voor een test, de verwerkingscondities, en de voorspelde resultaten voor een te testen item specificieert.

test case suite *testreeks*

Een test of een aantal testen waarmee software kan worden onderzocht.

test comparator *testvergelijker*

Een testgereedschap dat de werkelijke resultaten die door geteste software wordt geproduceerd, met de verwachte of voorspelde resultaten van de in uitvoering zijnde test(en) vergelijkt.

test completion criterion *criterium voor de completering van testen*

Criterium voor het bepalen op welk tijdstip of onder welke condities de geplande testen zijn afgewerkt. Het completeren van de test wordt als functie van een test-meettechniek gedefinieerd.

test coverage *testdekking*

1. De mate waarin een bepaalde test of testserie alle gespecificeerde eisen voor een gegeven systeem of component afdekt.
2. Zie dekking (coverage).

test criteria *testscriteria*

De criteria waaraan een systeem of component moet voldoen om met succes aan een bepaalde test te kunnen worden onderworpen.

test design *testontwerp*

Documentatie waarin de details van de testbenadering voor een softwarefunctie of combinatie van softwarefuncties wordt gespecificeerd en de daarmee geassocieerde testen worden geïdentificeerd.

test documentation *testdocumentatie*

Documentatie waarin de plannen voor of de resultaten van het testen van een systeem of component worden beschreven.

test driver *testdriver*

1. Een software-eenheid die wordt gebruikt om een te testen module te initiëren. De eenheid voorziet het module van

testinvoergegevens, bestuurt en volgt de uitvoering van de test, en doet verslag van de testresultaten.

2. Een programma of een testgereedschap dat wordt gebruikt om de werking van de software te initiëren die voor een bepaalde test, testserie, testreeks of testproces bestemd is.

test environment *testomgeving*

Een beschrijving van de hardware- en software-omgeving waarin de test of testen worden uitgevoerd, alsmede alle andere software en hardware waarmee deze testsoftware tijdens het uitvoeren van de test(en) communiceert. Onder die andere software en hardware vallen bijvoorbeeld stubs en test-besturingseenheden.

test execution *testuitvoering*

Het verwerken van een test, testserie, testreeks of testproces door de software die wordt getest. Deze testuitvoering levert een testresultaat op.

test execution technique *testuitvoeringstechniek*

De methode die wordt gebruikt om de reële test uit te voeren, bijvoorbeeld handmatig, automatisch, half-automatisch, door middel van een signatuur enzovoort.

test generator *testgeneratieprogramma*

Een programma dat testen, testseries, testreeksen en testprocessen uitvoert die overeenkomstig een gespecificeerde strategie of heuristische methode worden gegenereerd.

test harness *testharnas*

Een testgereedschap dat uit een test-besturingseenheid en een testvergelijker bestaat.

test incident report *testincidentverslag*

Een document dat een voorval beschrijft dat tijdens het testen is opgetreden en waarvoor een nader onderzoek nodig is.

testing *toetsing*

1. Het proces waarin een systeem of een component onder gespecificeerde condities werkt, waarin de resultaten worden geobserveerd of vastgelegd, en waarin een evaluatie van een bepaald aspect of bepaalde aspecten van het systeem of component wordt gemaakt.
2. Het proces waarin een software-item wordt geanalyseerd met het doel de verschillen tussen bestaande en vereiste condities te bepalen (dat wil zeggen fouten te bepalen) en de functies, kenmerken en eigenschappen van de software-items te evalueren.

test item *testitem*

Een software-item dat aan een test wordt onderworpen.

test item transmittal report *testitem-verslag*

Een document dat één of meer items identificeert die aan een test of testproces moeten worden onderworpen. Het verslag bevat de momentele status van het testitem en informatie betreffende de locatie.

test log *testlog*

Een chronologische verslaggeving van alle relevante details betreffende de uitvoering van een test.

test measurement technique *testmeettechniek*

Een methode die wordt gebruikt om items die betrekking hebben op de dekking van een test of testen, te meten.

test objective *testdoelstelling*

Een geïdentificeerde reeks softwarefuncties of -eigenschappen die onder gespecificeerde condities zullen worden gemeten door het werkelijke gedrag te vergelijken met het vereiste gedrag zoals dat in de softwaredocumentatie is beschreven.

test phase *testfase*

De tijdsperiode in de levenscyclus van software waarin de componenten van een softwareproduct worden geëvalueerd en geïntegreerd en waarbij het softwareproduct intensief wordt onderzocht om te bepalen of al dan niet aan de gestelde eisen is voldaan.

test plan *testplan*

1. Een document dat de scope, benadering, middelen, en het rooster van geplande testactiviteiten beschrijft. Het plan identificeert testitems, de eigenschappen en functies die getest moeten worden, de testactiviteiten en -taken, wie de testen uit zal voeren, en de risico's die een onvoorziene planning noodzakelijk zouden kunnen maken.
2. Een document dat de technische en bestuurlijke benadering beschrijft die bij het testen van een systeem of component moet worden gevolgd. Typische onderwerpen in het plan bestaan uit beschrijvingen van de items die moeten worden getest, taken die moeten worden uitgevoerd, verantwoordelijkheden, roosters en middelen die voor de testactiviteiten nodig zijn.
3. Een verslaggeving van het testproces waarin de kwaliteit van de tester, de testomgeving, de testontwikkeltechniek(en) en de testmeettechnieken, alsmede de redenen van de gemaakte keuzes, worden gedetailleerd.

test procedure *testprocedure*

1. Een document waarin de gedetailleerde instructies betreffende de opstelling, uitvoering en evaluatie van de resultaten van een bepaald testproces zijn opgenomen.
2. Een document dat een reeks instructies bevat zoals in punt 1 van deze definitie is aangegeven.
3. Documentatie die een serie acties bevat waarmee een test kan worden uitgevoerd.

test readiness review: TRR *beoordeling van testvoorbereidingen*

1. Een beoordeling die wordt uitgevoerd om: de voorlopige testresultaten van één of meer configuratie-items te evalueren; te verifiëren in hoeverre de testprocedures voor elk afzonderlijk configuratie-item compleet zijn, in overeenstemming zijn met testplannen en testbeschrijvingen, en voldoen aan de test Eisen; na te gaan in hoeverre een project gereed is om de formele testen van de configuratie-items te vervolgen.
2. Een beoordeling zoals vermeld in punt 1 van deze definitie ten aanzien van hardware- of softwarecomponenten.

test record *testverslag*

Een ondubbelzinnige vastlegging van de identiteiten en versies van het te testen component, de testspecificatie en het resultaat van de test.

test repeatability *herhaalbaarheid van een test*

Een attribuut van een test dat aangeeft dat iedere keer dat een test wordt uitgevoerd hetzelfde resultaat wordt bereikt.

test report *testverslag*

Een document dat de uitvoering en resultaten beschrijft van de testen van en/of testactiviteiten die van toepassing zijn op een systeem of een component.

test script *testscript*

1. De term verwijst meestal naar de geautomatiseerde testprocedure die in een testharnas wordt toegepast.
2. Zie testprocedure.

test set *testserie*

Een aantal testen of testprocessen.

test set architecture *testarchitectuur*

De geneste relaties tussen verzamelingen testseries die de hiërarchische decompositie van de doelstellingen van de test weerspiegelen.

test specification *testspecificatie*

De dekking, de initiële toestand van de te testen software, de invoer, en de voorspelde uitkomst(en) van elke afzonderlijke test, testserie, testreeks of testactiviteit.

test summary report *samenvattend testverslag*

Een document waarin de testactiviteiten en -resultaten zijn samengevat. Het bevat eveneens een evaluatie van de daarmee corresponderende testitems.

test unit *testeenheid*

Een verzameling bestaande uit één of meer modules van een computerprogramma en daarmee geassocieerde besturingsgegevens (zoals tabellen), gebruiksvorschriften, en operationele procedures die aan de volgende voorwaarden voldoen: (a) dat alle modules deel uitmaken van één computerprogramma, en (b) dat de verzameling modules en de daarmee geassocieerde gegevens en procedures het enige doel van de testactiviteiten vormt.

text editor *1. teksteditor; 2. tekstopmaakprogramma*

Een programma waarmee tekst of tekstbestanden kunnen worden aangepast, gewijzigd, en/of verbeterd.

text file *tekstbestand*

Een bestand dat tekst bevat. Een dergelijk bestand zal over het algemeen geen besturingscodes, programma's enzovoort bevatten en zal uitsluitend uit ASCII-tekens bestaan.

text formatter *tekstopmaakprogramma*

Een programma dat tekstbestanden leest en een fraai ogend geformatteerd resultaat oplevert.

THEN-clause *THEN-clausule*

Het gedeelte van een IF-bewering in een IF-THEN-ELSE bewerking dat wordt gebruikt om de actie te specificeren die moet worden verricht indien de vergelijking van de opdrachtelelementen van de IF-opdracht waar is.

thin client *slanke client*

Een systeem dat onder een zeer licht besturingssysteem zonder lokale administratieve procedures draait en Java-applicaties verwerkt die via een netwerk worden aangeleverd.

think time *denktijd*

De tijd die verloopt tussen het einde van een bericht of prompt die door een interactief systeem wordt gegenereerd en het begin van de reactie of het antwoord van een persoon.

third generation language: 3GL *derde-generatietaal*

Zie hogere-orde taal (high order language).

thrashing *operationele ondergang*

Een situatie waarin een computersysteem praktisch alle tijd van de systeemelementen gebruikt voor het uitvoeren van overheadfuncties, zoals het constant transporteren van gegevens tussen hoofdgeheugen en hulpgeheugens, in plaats van het uitvoeren van de beoogde rekenfuncties.

thread *1. thread; 2. draadmodule*

- a. Een verzameling processen of programma's waarvan de volgorde bepaalt welk proces of programma in aanmerking komt om te worden uitgevoerd.
- b. De basiseenheid van de programma-uitvoering. Een proces kan verschillende draadmodules bezitten die tegelijkertijd werken waarbij elk van die draadmodules een verschillende taak uitvoert, zoals wachten op gebeurtenissen, of het verwerken van een langdurig karwei dat niet onmiddellijk in haar geheel afgewerkt behoeft te worden. Indien een draadmodule de taak heeft beëindigd, wordt de taak beëindigd, opgeschort of vernietigd.
- c. Een proces binnen een programma. Een meervoudig gekoppeld programma kan twee of meer draadmodules (threads) uitvoeren door tijdsegmenten van de centrale verwerkingseenheid te verdelen. Het lijkt dan alsof een aantal bewerkingen tegelijkertijd worden uitgevoerd.

thread testing *testen van draden*

Een variatie op het testen dat van boven naar beneden (top-down) plaats vindt en waar de opeenvolgende integratie van componenten de implementatie van deelverzamelingen van de eisen volgt. Dit in tegenstelling tot het testen van de integratie van componenten die op lager gelegen niveaus wordt uitgevoerd.

three-address instruction *instructie met drie adressen*

Een computerinstructie die drie adresvelden bevat. Bijvoorbeeld, een instructie voor het optellen van de inhoud van locaties A en B, en het plaatsen van het resultaat van deze optelling in locatie C.

three-plus-one address instruction *instructie met drie adressen en een extra adres*

Een computerinstructies die vier adresvelden bevat. Het vierde veld bevat het adres van de instructie die vervolgens moet worden uitgevoerd. Bijvoorbeeld, een instructie voor het optellen van de inhoud van de locaties A en B, een instructie die het resultaat van de optelling in locatie C plaatst, waarna de instructie in locatie D wordt uitgevoerd.

throughput *doorvoer*

De hoeveelheid werk die door een computersysteem of component in een bepaalde tijdsperiode kan worden uitgevoerd. Bijvoorbeeld het aantal karweien per dag.

time out *blokkeertijd*

Een situatie die optreedt indien een vooraf vastgestelde tijd verloopt zonder dat er van de occurrence van een verwachte gebeurtenis sprake is. Bijvoorbeeld, de situatie die een on line proces doet beëindigen indien binnen een bepaalde tijdsperiode geen invoergegevens van een gebruiker worden ontvangen.

time-sharing *1. time-sharing; 2. tijdscharing*

- a. Het afwisselend gebruiken van de verwerkingstijd van een computersysteem waardoor twee of meer taken gelijktijdig kunnen worden afgewerkt.
- b. Een methode van werken met een computersysteem waarbij door middel van het afwisselend gebruik van de verwerkingseenheid, twee of meer processen gelijktijdig kunnen worden afgehandeld. Deze processen kunnen eventueel op interactieve wijze met de gebruikers worden uitgevoerd. Opmerking: time-sharing kan met behulp van tijdsegmentering, op prioriteiten gebaseerde onderbrekingen of andere roostermethodieken worden uitgevoerd.

timing *tijdmeting*

Het proces waarin de uitvoerings- of verwerkingstijd van een softwaresysteem of -component wordt gemeten (of geschat).

timing analyzer *tijdmetingsanalyseprogramma*

Een softwaregereedschap dat de uitvoerings- of verwerkingstijd van een computerprogramma of segment van een computerprogramma meet (of schat). Dit gebeurt door òf de verwerkingstijden van de instructies die bepaalde paden aflopen te sommeren òf door meetpunten in het programma op te nemen en de uitvoerings- of verwerkingstijden tussen deze punten te meten.

token *token*

Een fundamentele en grammatisch ondeelbare eenheid in een taal.

tool builder kit: TBK *gereedschapsontwikkelingskist*

Een gereedschapskist waarmee software-ontwikkelingsgereedschappen (CASE tools) kunnen worden ontwikkeld die geschikt zijn voor elke mogelijke software-ontwikkelingsmethodologie.

tool command language: Tcl *gereedschap-commandotaal*

Een commandotaal en het daarbij behorende bibliotheekprogramma dat op een groot aantal platforms kan worden geïmplementeerd en gebruikt.

top-down *top-down*

De term heeft betrekking op een activiteit die bij of met het hoogste punt van een hiërarchie aanvangt en achtereenvolgens alle lagere niveaus doorloopt.

top-down testing *1. van boven naar beneden testen; 2. top-down testen*

Een benadering van het integratietesten waarbij het component dat zich aan de top van de hiërarchie bevindt eerst wordt getest en waarbij de in de hiërarchie lager gelegen componenten met behulp van stubs worden gesimuleerd. Geteste componenten worden daarna gebruikt om de lager gelegen componenten te testen. Het proces wordt herhaald totdat de componenten die zich op het laagste niveau van de hiërarchie bevinden, zijn onderzocht.

total correctness *algehele correctheid*

In correctheidsbewijzen verwijst de term naar een designatie die aangeeft dat de uitvoerasserties van een programma een logisch gevolg zijn van de invoerasserties en bewerkingstappen en dat het programma tevens onder alle gespecificeerde invoercondities de verwerking normaal kan beëindigen.

trace *1. opsporing; 2. naspeuring; 3. trace*

Een verslaggeving van de uitvoering van een computerprogramma die de volgorde van de instructies, de namen en waarden van variabelen of beide toont. Er bestaan een aantal verschillende naspeuringen zoals uitvoeringstrace, retrospectieve trace, subroutine-trace, symbolische trace en de trace van variabelen.

trace (to) *1. nagaan; 2. naspeuren*

- a. Een verslaggeving, zoals genoemd in de vorige definitie, produceren.
- b. Een relatie tussen twee of meer producten van het ontwikkelingsproces vaststellen; bijvoorbeeld, de relatie tussen een bepaalde eis en het ontwerp-element dat die eis implementeert, vaststellen.

traceability *naspeurbaarheid*

1. De mate waarin een relatie kan worden vastgesteld tussen twee of meer producten die een voorganger-opvolger of chef-ondergeschikte relatie met elkaar hebben.
2. De term heeft in de software-ontwikkeling betrekking op de succesrijke kruisverwijzing tussen de modules, routines of assets van een softwaresysteem. De kruisverwijzingen gelden voor het gehele traject vanaf de operationele modules tot aan de oorspronkelijke eisen en behoeften, met daartussen in alle fasen van de ontwikkeling en het testen, alsmede de daaraan verwante documentatie.

3. In de domeinontwikkeling verwijst de term naar de karakteristieken van domeinmodellen, domeinarchitecturen, ontwerpen, en softwaresystemen die, voor elk specifiek asset, het derivatiepad (het pad vanaf de voorgaande fase in het software-ontwikkelingsproces) en het allocatiepad (het pad naar de volgende fase) van eisen, behoeften en beperkingen identificeren en documenteren.

traceability matrix *naspeurbaarheidsmatrix*

Een matrix waarin de relaties tussen twee of meer producten uit het ontwikkelingsproces zijn vastgelegd; bijvoorbeeld, een matrix die de relatie tussen de eisen en het ontwerp van een bepaald software-component toont.

trace program *volg- en controleprogramma*

Een programma dat een ander programma controleert door de volgorde, waarin de instructies worden uitgevoerd, alsmede de resultaten van de uitvoering van die instructies weer te geven.

trace routine *weergaveroutine*

Een programmadeel dat zorgt voor de weergave van bepaalde gebeurtenissen die tijdens de uitvoering van het programma plaats vinden.

trailer *uitloper*

Identificatie- of besturingsinformatie die aan het einde van een bestand of boodschap is geplaatst.

trailing decision *nalopende verwerking*

Een lusbesturing die na de lus-body wordt uitgevoerd.

transaction *transactie*

De term verwijst in de software-ontwikkeling naar een gegevenselement, besturingselement, signaal, gebeurtenis, of een verandering van de toestand die een actie of een serie acties veroorzaakt of initieert.

transaction analysis *transactieanalyse*

Een software-ontwikkelingstechniek waarin de structuur van een systeem wordt afgeleid met behulp van de analyse van de transacties die het systeem moet verwerken.

transaction matrix *transactiematrix*

Een matrix die de mogelijke verzoeken voor het raadplegen van een database identificeert en elk verzoek doet corresponderen met informatiecategorieën in of elementen van de database.

transfer (to) *1. transporteren; 2. overdragen*

1. Informatie van een bepaalde plaats versturen en dit op een andere plaats ontvangen.
2. Besturing door een bepaald proces overdragen aan een ander proces. Dit geschiedt in de verwachting dat die besturing òf terug wordt gegeven aan het proces dat de besturing heeft afgedragen òf dat dit nimmer plaats zal vinden.

transient error *vluchtige fout*

Een fout die slechts eenmaal of op onvoorspelbare momenten optreedt.

transient object *vluchtig object*

Een object waarvan het bestaan beperkt is (wordt) door de levensduur van het proces waarin het werd gecreëerd.

translate (to) *1. vertalen; 2. verplaatsen*

1. Het omzetten van gegevens van de ene taal of vorm in een andere.
2. Het verplaatsen van het gehele beeld of een gedeelte van een beeld op een beeldscherm.

translation *vertaling*

De conversie van de ene taal naar een andere zonder daarbij de inhoudelijke betekenis van de taal te wijzigen.

translator *vertaler*

Een computerprogramma dat een reeks instructies die in een bepaalde taal zijn uitgedrukt, vertaalt of omzet in een equivalente reeks instructies van een andere taal.

trap *val*

Een voorwaardelijke sprong naar een routine die een uitzonderingstoestand of onderbreking (exceptie of interrupt) afhandelt. Dit gebeurt meestal automatisch en wordt door de hardware geactiveerd, terwijl de locatie waarin de sprong voorkwam, wordt geregistreerd.

tree *boom*

Een verzameling knooppunten die een willekeurig aantal verwijzingen naar andere knooppunten heeft. Elke twee onderling verbonden knooppunten hebben één uniek eenvoudig pad of bezitten één unieke eenvoudige onderlinge verbinding.

Trellis *Trellis*

Een object-georiënteerd applicatieontwikkelingssysteem van de firma DEC.

turnaround time *omlooptijd*

De tijd die tussen het moment van het aanbieden van een karwei aan een partij-verwerkend systeem en het moment van de ontvangst van de totale gecompleteerde uitvoer verstrijkt.

turnkey *turnkey*

De term heeft betrekking op een hardware- of softwaresysteem dat in een complete en operationele toestand wordt afgeleverd.

two-address system *instructie met twee adressen*

Een computeropdracht die twee adresvelden bevat. Bijvoorbeeld, een instructie die de inhoud van locatie A optelt bij de inhoud van locatie B.

two-level address *adres met twee niveaus*

Een indirect adres dat de geheugenlocatie bevat van het adres met de gewenste operand.

two-level encoding *codering in twee niveaus*

Een techniek in de microprogrammering waarin twee microbewerkingen op identieke wijze in hetzelfde veld van een micro-instructie worden gecodeerd waarbij de bewerking die wordt uitgevoerd afhankelijk is van de waarde in een ander veld in dezelfde of een andere micro-instructie.

two-plus-one address instruction *instructie met twee adressen en een extra adres*

Een computerinstructie die drie adresvelden bevat. Het derde veld bevat het adres van de instructie die vervolgens moet worden uitgevoerd. Bijvoorbeeld, een instructie die de inhoud van de locatie A bij die van B optelt, waarna de instructie in locatie C wordt uitgevoerd.

type *type*

1. Een klasse of een interface.
2. Een predikaat dat aan waarden is/wordt toegekend en dat in een signatuur kan worden gebruikt om een mogelijke parameter te beperken of een resultaat te karakteriseren.
3. Zie gegevenstype.

type interface *type-interface*

De term verwijst naar de definitie van verzoeken waarin exemplaren (instances) van dit type op zinvolle wijze als parameters dienst kunnen doen.

type object *type-object*

Een object dat als type fungeert.

U

Ultrix *Ultrix*

De Unix-versie van de firma DEC die gebaseerd is op die van de Universiteit van Californië in Berkeley.

unconditional branch *onvoorwaardelijke sprong*

Een sprong die ondanks bepaalde bewerkingscondities toch wordt uitgevoerd.

underflow exception *onderloopexceptie*

Een uitzonderingstoestand (exceptie) die optreedt indien het resultaat van twee rekenkundige bewerkingen te klein is om door de geheugenlocatie waarin dit resultaat moet worden geplaatst, te worden gerepresenteerd.

undirected graph *niet-gerichte graaf*

Een graaf waarin de richtingen tussen de knooppunten niet zijn gedefinieerd of niet worden geïmpliceerd.

unit *eenheid*

1. Een afzonderlijk te testen element dat als zodanig in het ontwerp van een softwarecomponent is gespecificeerd.
2. Een logisch af te zonderen deel van een computerprogramma.
3. Een softwarecomponent dat niet in andere componenten is opgedeeld.
4. Zie testeenheid.

Opmerking: De termen 'module', 'component', en 'eenheid' zijn vaak onderling verwisselbaar, zijn afhankelijk van de context waarin ze worden gebruikt, of worden als deelelementen van elkaar gedefinieerd. De onderlinge relatie tussen deze drie termen is niet gestandaardiseerd.

unit development folder: UDF *eenheidsontwikkelingsfolder*

Zie software-ontwikkelingsbestand.

unit requirements documentation *documentatie van de eisen van een (test)eenheid*

Documentatie waarin de eisen van functionele, interface-, prestatie- en ontwerpbeperkingen voor een testeenheid zijn gespecificeerd.

unit testing *testen van een eenheid*

Het testen van de individuele hardware- of software-eenheden of groepen die uit onderling verwante eenheden bestaan.

universal modelling language: UML *universele modelleringstaal*

Een hoofdzakelijk grafische beschrijvings- en definitiemethode of notatie voor de specificatie van grote object-georiënteerde analyse- en ontwerpideeën. UML maakt gebruik van eerder gedefinieerde grafische beschrijvingstalen. Een methode of proces in UML bestaat uit aanbevelingen of adviezen van de wijze waarop de object-georiënteerde analyse en het object-georiënteerde ontwerp kunnen worden uitgevoerd. UML kent diagrammen of grafische notaties voor gebruiksgevallen, klassen, volgorden, componenten, toestands-schema's, samenwerkingen (collaborations) en opstellingen (deployments). UML is evenals CORBA ontwikkeld door een industrieel consortium, de OMG (Object Management Group).

Unix *Unix*

Een computer-onafhankelijk besturingssysteem dat in 1972 in de Bell Laboratoria van AT&T is ontwikkeld. Ook werd daar de computer-onafhankelijke C-taal geconcipieerd. Unix werd in de C-taal geschreven en kon op die wijze redelijk eenvoudig tussen verschillende computersystemen worden overgedragen. Sinds 1972 zijn er echter een aantal varianten van Unix gepubliceerd, die allen onderlinge verschillen vertonen en de oorspronkelijke standaardisatiepogingen van dit besturingssysteem ernstig hebben bemoeilijkt.

unpack (to) *uitpakken*

Het herstellen van de oorspronkelijke vorm van één of meer gegevensitems vanuit gegevens die zijn ingepakt.

unstratified language *niet-gelaagde taal*

Een taal die als de eigen metataal kan worden gebruikt.

UNTIL-clause *UNTIL-clausule*

Definitie van een lus met één enkele ingang en één enkele uitgang waarin de lusbesturing na de lus-body wordt uitgevoerd.

unwind (to) *afwickelen*

Expliciet en volledig alle programma-instructies coderen (uitschrijven) die in de meervoudige bewerkingen van een lus voorkomen. Zie lineaire codering (straightline coding).

up *in bedrijf*

De term verwijst naar een systeem of component dat operationeel is en waarvan tevens geldt dat onderhoud op het systeem van toepassing is. Een dergelijk systeem is òf bezig opdrachten te verwerken òf is inactief.

up time *bedrijfstijd*

De tijdsperiode gedurende welke een systeem of component operationeel is en waarvoor tevens geldt dat onderhoud op het systeem van toepassing is; dat wil zeggen, de bedrijfstijd is de som van de

tijd waarin het systeem bezig is opdrachten te verwerken en de tijd waarin het systeem inactief is.

upward compatible *bovenwaarts-compatibel*

De term heeft betrekking op hardware of software die compatibel is met een latere of meer ingewikkelde versie van zichzelf; bijvoorbeeld, een programma dat bestanden verwerkt die door een latere versie van datzelfde programma zijn gecreëerd.

upward compression *bovenwaartse compressie*

De term verwijst in het ontwerp van software naar een demodularisatie-vorm waarin een ondergeschikt module tussen de programma-opdrachten van een superieur module wordt gekopieerd; dat wil zeggen, het module wordt in-line in het superieure module gekopieerd.

usability *bruikbaar*

Het gemak waarmee een gebruiker een systeem of een component kan leren gebruiken, bedienen, interpreteren of invoergegevens voor kan leren prepareren.

use case *gebruiksgeval*

De term heeft betrekking op de wijze waarop gebruikers met elkaar communiceren of samenwerken. Gebruikers kunnen in dit geval fysieke of abstracte representaties hebben zoals personen, organisaties, systemen enzovoort.

use case diagram *diagram van een gebruiksgeval*

Een diagram dat het scenario van een gebruiksgeval in de universele modelleringstaal (UML) afbeeldt.

use case scenario *scenario van een gebruiksgeval*

Een beschrijving die in gestructureerd Engels of in een zeer beknopte notatie is geschreven. De beschrijving geeft mogelijke situaties aan die al dan niet door een applicatie kunnen worden verwerkt of afgehandeld. Het scenario beschrijft de wijze waarop een organisatie met personen of andere organisaties en systemen communiceert.

user documentation *gebruikersdocumentatie*

Documentatie waarin de wijze waarop een systeem of component moet worden gebruikt om de gewenste resultaten te verkrijgen, wordt beschreven.

user friendly *gebruik(er)svriendelijk*

De term heeft betrekking op een eenheid, computersysteem, programma of document dat zodanig is ontworpen, ontwikkeld of opgemaakt dat het gebruik(er)sgemak daarin doorslaggevend is.

user interface *gebruikersinterface*

1. Een interface met behulp waarvan informatie tussen mensen en de hardware- of softwarecomponenten van een computersysteem kan worden uitgewisseld.
2. Apparatuur, programmatuur of een combinatie van beide met behulp waarvan een gebruiker een interactie of interacties kan plegen met en/of bewerkingen kan verrichten op een systeem, programma of apparaat.

user interface management system: UIMS *beheerssysteem voor gebruikersinterfaces*

Een systeem dat de ontwikkeling en het werken met gebruikersinterfaces ondersteunt.

user manual *gebruikershandleiding*

Een document dat de informatie bevat die nodig is om met een systeem te kunnen werken voor het verkrijgen van de gewenste resultaten. Het document beschrijft met name: de functionele mogelijkheden, de beperkingen, de opties, de toegestane invoer, de verwachte uitvoer, mogelijke foutboodschappen en speciale instructies van of voor het systeem of het component. Opmerking: een gebruikershandleiding verschilt van de handleiding van de bedieningsfunctionaris indien een onderscheid wordt gemaakt tussen hen die de normale werking van een computersysteem besturen of regelen (zoals het opzetten van magneetbanden) en hen die het systeem gebruiken voor het doel waarvoor het is ontwikkeld en geïnstalleerd.

utility *utiliteitsprogramma*

Een functie, routine of programma voor de ondersteuning van bepaalde geautomatiseerde systemen, programma's, of taken die onder de besturing van een besturingssysteem worden uitgevoerd. De utiliteitsprogramma's zijn over het algemeen een onderdeel van het besturingsprogramma.

utility program *utiliteitsprogramma*

1. Een programma dat een algehele ondersteuning vormt van de processen binnen een computer, bijvoorbeeld een diagnostisch, een controle- en een sorteerprogramma.
2. Een programma dat is ontworpen om routinewerkzaamheden te verrichten, zoals het kopiëren van gegevens van een bepaald geheugengebied naar een ander.

utility routine *utiliteitsroutine*

Een module die een algehele ondersteuning vormt van de processen binnen een computer, zoals een invoerroutine.

utilization *benutting*

In de evaluatie van de prestaties van een computer verwijst de term naar de verhouding van de tijd waarin een systeem of component in bedrijf is en de totaal beschikbare tijd.

V

validation *validatie*

Het proces waarin een systeem of component tijdens of aan het einde van het ontwikkelingsproces wordt geëvalueerd met het doel te controleren in hoeverre aan de gespecificeerde eisen is of wordt voldaan.

value *waarde*

1. Een grootheid die aan een constante, een variabele, een parameter of een symbool wordt toegekend.
2. Een specifieke hoedanigheid van een kenmerk, bijvoorbeeld 'geel' voor de eigenschap 'KLEUR'.
3. Elke entiteit die als een mogelijke actuele parameter in een verzoek kan fungeren.

value-dependent operation *waarde-afhankelijke bewerking*

Een bewerking waarin het gedrag van het met die bewerking geassocieerde verzoek afhankelijk is van de namen waarmee parameters van objecten kunnen worden geïdentificeerd.

variable *variabele*

1. Een teken dat of een groep tekens die in een programma naar een waarde verwijst; bij het uitvoeren van het programma komt deze groep tekens overeen met een adres.
2. Een grootheid die een bepaalde waarde uit een reeks waarden aan kan nemen.
3. Een gegeven waarvan de waarde tijdens het uitvoeren van een programma kan worden gewijzigd.
4. Een gegeven dat geen reeks of reekselement is en dat door een symbolische naam wordt geïdentificeerd.
5. Een element dat wordt gebruikt om naar gegevens te verwijzen en waaraan waarden kunnen worden toegekend.
6. Een gegevensitem dat een naam bezit waarmee het kan worden geïdentificeerd. Elke variabele heeft een type (zoals INTEGER of OBJECT) en een scope.

variable trace *trace van variabelen*

Een opsomming van de naam en de waarden van de variabelen die tijdens de uitvoering van een computerprogramma zijn gebruikt of gewijzigd.

variant *variant*

Een versie van een programma dat in de fouttolerantie het resultaat is van het toepassen van de techniek die betrekking heeft op het creëren van de software-diversiteit.

verification *verificatie*

1. Het nagaan en/of controleren of een bewerking of een proces op de juiste wijze is uitgevoerd.
2. Het proces waarin van een systeem of een component wordt bepaald in hoeverre de producten van een bepaalde fase in de levenscyclus aan de opgestelde condities en specificaties voldoen.
3. Het proces waarin wordt geëvalueerd of een systeem of component in een bepaalde fase van de ontwikkeling aan de voorwaarden voldoet die aan het begin van die fase zijn vastgesteld.
4. Het formele bewijs van de correctheid van een programma.

verification and validation: V&V *verificatie en validatie*

Het proces waarin wordt bepaald of de eisen die aan een systeem of component zijn of worden gesteld, volledig en juist zijn; of de producten van elke afzonderlijke ontwikkelingsfase voldoen aan de eisen of voorwaarden die aan het begin van elke fase en elke eerdere fase aan de producten zijn gesteld; en of het uiteindelijke systeem of component zich conformeert aan de gespecificeerde eisen.

version *versie*

1. Een initiële vrijgave of vroegtijdige en voorlopige vrijgave van een configuratie-item van computersoftware en de daarmee geassocieerde volledige compilatie of hercompilatie van het desbetreffende configuratie-item.
2. Een initiële vrijgave of vroegtijdige en voorlopige vrijgave van een document. Dit in tegenstelling tot een revisie die het gevolg is van de uitgifte van wijzigingsbladen die van toepassing zijn op een eerdere vrijgave.
3. Een programmaproduct dat op een bestaand programmaproduct is gebaseerd en waarin gewoonlijk nieuwe of verbeterde functies zijn aangebracht.

version description document: VDD *versie-beschrijvingsdocument*

Een document dat een bepaalde versie van een systeem of component begeleidt en identificeert. De inhoud van een dergelijk document bestaat uit de (onder)delen van het systeem of component, de identificatie van wijzigingen die in deze versie zijn aangebracht, en de installatie en operationele informatie die specifiek van toepassing zijn op datgene wat in deze versie is beschreven.

vertical domain *verticaal domein*

Een domein waarin aspecten van een functie of applicatiegebied aan de orde worden gesteld. Voorbeelden zijn een salarissysteem, een

numeriek besturingssysteem, of een voorraadsysteem. Een verticaal systeem maakt gebruik van mogelijkheden die door horizontale systemen worden geboden.

vertical microinstruction *verticale micro-instructie*

Een micro-instructie die één bewerking in een reeks van bewerkingen specificiert waarmee een opdracht van de machinetaal kan worden uitgevoerd. Opmerking: Verticale micro-instructies zijn relatief kort, 12 tot 24 bits, en worden verticaal genoemd omdat deze reeks instructies, die normaal verticaal op een pagina of bladzijde worden geschreven of afgedrukt, nodig is om één enkele machinetaal-instructie uit te kunnen voeren.

vertical reuse *verticaal hergebruik*

Het hergebruik van assets in een verticaal domein.

virtual address *virtueel adres*

Het adres dat naar het virtuele geheugen verwijst en dat, indien het wordt gebruikt, door het besturingsprogramma in een reëel geheugenadres moet worden vertaald.

virtual machine *virtuele machine*

Een abstracte specificatie van een rekeneenheid die op verschillende manieren kan worden geïmplementeerd: in hardware, in software of beide. Men kan een programma naar het instructieset van een virtuele machine compileren op dezelfde wijze als men dat naar de instructiereeks van een microprocessor doet. Zo heeft een virtuele Java-machine een instructieset dat uit bytecodes, een aantal registers, een stapel, een saneerder, en een gebied dat bestemd is voor het opslaan van methoden, bestaat.

virtual memory *virtueel geheugen*

Adresseerbare geheugenruimte die zich aan de ontwikkelaar of gebruiker als reëel geheugen manifesteert en van waaruit de instructies en gegevens naar reële geheugenplaatsen worden overgebracht.

virtual software factory: VSF *virtuele softwarefabriek*

Een product van de firma Systematica waarmee gebruikers CASE-gereedschappen kunnen ontwikkelen die in elke mogelijke software-ontwikkelingsmethode kunnen worden toegepast.

virtual storage *virtuele opslag*

Een geheugenallocatietechniek waarin het hulpgeheugen kan worden geadresseerd als ware het een deel van het hoofdgeheugen. Delen van een programma en gegevens worden in het hulpgeheugen geplaatst en het besturingssysteem plaatst deze automatisch in het hoofdgeheugen

en weer terug in het hulpgeheugen al naar gelang de noodzaak daartoe aanwezig is of daarom vraagt.

Visual Basic *Visual Basic*

Een programmeertaal en -omgeving die door Microsoft is ontwikkeld. Visual Basic is gebaseerd op de programmeertaal Basic en was één van de eerste producten die was voorzien van een grafische programma-ontwikkelingsomgeving, alsmede grafische software die op de software lijkt waarmee beelden kunnen worden verwerkt en gemanipuleerd. De programmeur kan een aanzienlijke hoeveelheid code aan het programma toevoegen door gebruik te maken van toetsen en dialoogfuncties, waarna vervolgens de verschijningsvorm en het gedrag van die code kunnen worden gedefinieerd. Dit ontlast de programmeur van de zorg grote aandacht aan de details ten aanzien van de syntax te moeten besteden. Hoewel Visual Basic geen object-georiënteerde programmeertaal in de strikte betekenis van deze term is, bezit het toch een object-georiënteerde filosofie. Het wordt soms ook wel een gebeurtenis-gestuurde taal genoemd omdat elk object op verschillende gebeurtenissen, zoals het indrukken van een toets, reageert. Sinds de lancering van Visual Basic in 1990 is deze benadering de norm geworden voor andere programmeertalen, met inbegrip van Pascal, C, C++, Pascal, en Java.

visualization *visualisatie*

Een methode waarmee een computersysteem gegevens aan de gebruiker presenteert of deze gegevens op een beeldscherm projecteert.

W

waiver *verklaring van afstand*

Een geschreven document waarin wordt verklaard dat een configuratie-item of enig ander geselecteerd item tijdens het ontstaan of de productie van dat item is gecontroleerd en dat daarbij is gebleken dat dit item afwijkt van de gespecificeerde eisen, maar dat het desalniettemin geschikt wordt bevonden om als zodanig te worden gebruikt of dat het na een herziening met behulp van een goedgekeurde methode kan worden gebruikt.

walkthrough *programma-inspectie*

1. Een statische analysetechniek waarin een ontwerper of een programmeur met de leden van het ontwikkelingsteam en andere geïnteresseerden door een segment van de documentatie of de code loopt en de deelnemers aan deze inspectie vragen stelt en opmerkingen maakt ten aanzien van mogelijke fouten, het niet naleven van ontwikkelingsnormen, en andere problemen.
2. Een beoordeling van eisen, behoeften, ontwerp en/of code van een object, component of systeem. De voortgang van bepaalde

ontwikkelingsactiviteiten van objecten, componenten en systemen kan met een dergelijke beoordeling worden geëvalueerd.

waterfall model *watervalmodel*

Een model van het ontwikkelingsproces van software waarin de constituerende activiteiten, bestaande uit een conceptfase, een eisenfase, een ontwerpfase, een implementatiefase, en een installatie- en controlefase in die volgorde worden uitgevoerd. Er kunnen overlappingen tussen deze fasen voorkomen; er mogen echter geen of slechts zeer weinig iteraties tussen de fasen plaats vinden.

wearout-failure period *slijtage- en storingsperiode*

De periode in de levenscyclus van een systeem of een component gedurende welke in toenemende mate hardwarestoringsen optreden die het gevolg zijn van de degeneratie van het systeem of het component.

Whetstone *Whetstone*

Whetstone is de naam van een veel gebruikt benchmarkprogramma.

WHILE-clause *WHILE-clausule*

Een lus met één enkele ingang en één enkele uitgang waarin de lusbesturing vóór de lus-body wordt uitgevoerd.

WHILE-DO operation *WHILE-DO bewerking*

Een bewerking (opdracht) in een programma om een programmalus uit te voeren zolang aan een bepaalde conditie wordt voldaan.

white-box testing *witte-box test*

Zie structurele test (structural test).

widget *widget*

Een venster (window) in het X Window systeem waarin de in- en uitvoerfuncties zijn afgebeeld. Deze widgets (dingetjes) zijn opgenomen in een bibliotheek en worden gebruikt als bouwblokken voor de constructie van een grote variëteit aan applicatieomgevingen.

window *1. window; 2. venster*

Een rechthoekig gebied van een beeldscherm waarbinnen symbolen, iconen en andere informatie is/wordt afgebeeld. Eén beeldscherm kan verschillende vensters tegelijkertijd bevatten. Deze vensters kunnen elkaar al dan niet geheel of gedeeltelijk overlappen.

window manager *venstermanager*

Een programma (zoals Open Look of Motif) dat het beeldscherm de verschijningsvorm van een grafische gebruikersinterface geeft. Deze venstermanager verzorgt de verplaatsing, de vergroting of verkleining van vensters en andere praktische functies.

window system *venstersysteem*

Software die het gebruik en de toepassing van vensters mogelijk maakt. Voorbeelden van dit type software zijn het X Window systeem, Windows van Microsoft, en venstersystemen van andere leveranciers zoals Apple (Macintosh), NeXT en Sun Microsystems.

word *woord*

1. Een serie bits of tekens die als een eenheid in een bepaalde computer wordt opgeslagen, geadresseerd, getransporteerd en waarop bewerkingen kunnen worden uitgevoerd.
2. Een element van een computergeheugen waarin een reeks bits of tekens, zoals vermeld in punt 1 van deze definitie, kan worden opgeslagen.
3. Een reeks bits of tekens die een betekenis heeft en in bepaalde talen als een entiteit wordt beschouwd; bijvoorbeeld, een gereserveerd woord in een programmeertaal.

working set *werkset*

In de pagineringsmethode die bij het toekennen van geheugenruimte wordt toegepast, verwijst deze term naar het aantal pagina's dat het meest waarschijnlijk permanent tijdens de uitvoering van een programma in het hoofdgeheugen aanwezig zal zijn.

working space *werkruimte*

Dat deel van het hoofdgeheugen dat voor de tijdelijke opslag van gegevens aan een computerprogramma is toegewezen.

workload *werkbelasting*

De verscheidenheid aan taken die door een computersysteem worden verwerkt. De belangrijkste karakteristieken daarvan zijn: de eisen die aan de in- en uitvoer worden gesteld, het aantal en de soort berekeningen en bewerkingen die in die taken worden uitgevoerd, en de vereiste computerelementen of -(hulp)middelen.

workload model *model van de werkbelasting*

Een model dat in de evaluatie van computerprestaties wordt gebruikt en waarin het gebruik van de elementen en het meten van de prestaties van verwachte en werkelijke werkbelastingen in een computersysteem worden uitgebeeld of weergegeven.

write (to) *schrijven*

Gegevens in een geheugenruimte of gegevensmedium vastleggen.

X

X Athena widget: XAW *X-Athena widget*

Een serie widgets die door het X Window systeem worden gedistribueerd.

X client *X-client*

Een applicatie in het X Window systeem; deze applicatie krijgt toegang tot de functies van het X Window systeem via de X-bibliotheek. Deze worden door het systeem vertaald in berichten en/of boodschappen die bestemd zijn voor een X-server.

X image extension: XIE *afbeeldingsuitbreiding van X-protocol*

Uitbreidingen van het X-protocol waarmee het manipuleren van beelden in het X Window systeem mogelijk is.

X library: Xlib *X-bibliotheek*

Een programma-interface van het X Window systeem.

X/Open *X/Open*

Een internationaal consortium van leveranciers. Dit consortium spant zich in om de in dit consortium opgestelde standaarden, die ontwikkeld zijn om de portabiliteit tussen applicaties mogelijk te maken, te propageren en te distribueren.

X/Open common applications environment *X/Open omgeving voor gemeenschappelijke applicaties*

Een door het X/Open consortium gespecificeerde omgeving waarin applicaties die binnen het X Window systeem werken, gemeenschappelijk kunnen worden gebruikt.

X/Open system interface: XSI *systeeminterface van X/Open*

Een systeeminterface die deel uitmaakt van de X/Open omgeving voor het werken met en het gebruik van gemeenschappelijke applicaties.

X protocol *X-protocol*

Een standaard die door cliënten (applicaties) en servers in het X Window systeem wordt gebruikt ten behoeve van de uitwisseling van verzoeken die betrekking hebben op venstermanipulaties.

X server *X-server*

Een programma dat afbeeldingen op beeldschermen projecteert bij het gebruik van het X Window systeem. De server voert bewerkingen op aanvraag van client-applicaties uit.

X terminal *X-terminal*

1. Een terminal of eindstation dat als X-server kan fungeren. Met een dergelijke terminal kan van het X Window systeem, Motif of Open Look gebruik worden gemaakt.
2. Een intelligent eindstation dat een ingebouwde implementatie van een X-server heeft.

X user interface: XUI *X-gebruikersinterface*

Een programma-interface van de firma DEC voor het X Window systeem.

XView *XView*

Een gereedschapskist van de firma Sun Microsystems. XView verschaft de gebruiker een Open Look interface voor X Window applicaties.

X Window system *X Window systeem*

Een venstersysteem met een specificatie voor hard- en software-onafhankelijke vensterapplicaties waarvan de resultaten op bitmap-schermen zichtbaar kunnen worden gemaakt. Het systeem is ontwikkeld op het MIT en is thans een de facto standaard die wordt ondersteund door het X/Open consortium.

Y

yet another compiler-compiler: YACC *nog een compiler-compiler*

Een generator van ontledingsprogramma's (parsers) die voor het Unix-systeem is ontwikkeld.

Z

zero-address instruction *instructie zonder adressen*

Een computerinstructie die geen adresvelden bevat.