

ORDLISTA

Version 2.3

**Baserad på "Standard Glossary of Terms used in Software
Testing, version 2.3"**

Meddelande om upphovsrätt

Detta dokument får kopieras i sin helhet eller delar av, så länge som referens till källan uppges. Den senaste versionen av SSTB Ordlista finns publicerad på <http://www.sstb.se/>.

Medverkande

Sigrid Eldh (SSTB)
Anders Pettersson (SSTB)
Beata Karpinska Säter (SSTB)
Maria Jönsson (SSTB)
Susanne Lieberg (SSTB)
Ingegerd Bynert (Combitech)
Kennet Osbjør (Addq)
Johan Klintin (SSTB)
Ninna Morin (SSTB)

Ingvar Nordström (SSTB)
Ingela Skytte (SSTB)
Klaus Zeuge (SSTB)
Patrik Norrby (SSTB)
Daniel Sundmark (Mälardalens Högskola)
Mattias Nordin (Addq)
Mats Grindal (SSTB)
Robert Bornelind (SSTB)
Tobias Ahlgren (SSTB)

Innehållsförteckning

Förord.....	4
1 Introduktion.....	4
2 Omfattning.....	5
3 Disposition	5
4 Normgivande referenser	5
5 Varumärken	6
6 Definitioner	6
A.....	6
B.....	10
C.....	11
D.....	12
E.....	14
F.....	15
H.....	19
I.....	20
J.....	22
K.....	22
L.....	27
M.....	28
N.....	30
O.....	31
P.....	32
Q.....	34
R.....	34
S.....	36
T.....	39
U.....	47
V.....	48
W.....	50
Å.....	50
Ä.....	50
Ö.....	51
Bilaga A	52
Bilaga B	54

Förord

Det är en svår uppgift att dokumentera, beskriva och definiera en begreppsvärld som används inom ett speciellt område. Det finns alltid olika uppfattningar om vilka termer som verkligen används eller vilka termer som bör användas. Ordlistor för terminologi inom programvarutestning som finns tillgängliga i dag är ofta på annat språk än svenska eller begränsar sig inte till området programvarutestning.

Arbetet med att framställa denna ordlista har fokuserat på att ordlistan ska innehålla vanligt förekommande ord som används av testare inom Sverige såväl som i övriga världen, men att orden i störst möjliga mån ska vara på svenska. Dock är det inte alltid som en svensk översättning är lämplig; i dessa fall bör det engelska ordet även i fortsättningen användas. Vid diskussioner om val av ord och eventuella oklarheter i definition har hänsyn tagits till synpunkter från olika områden där programvarutestning är en del av eller är den huvudsakliga utövningen. Exempel på företag och områden är konsultföretag inom programvarutestning, bankvärlden, utbildningsföretag och forskning inom den akademiska världen. Vid val av ord och definition har dock störst vikt lagts vid de standarder som finns i dag.

En utav de mest använda standarder inom programvarutestning är och har varit "BS 7925-2:1998, *Software Component Testing*". Det finns två nackdelar: standarden är skriven utifrån ett komponentbaserat perspektiv och under de år som standarden har funnits har det getts kommentarer på förbättringar i befintliga definitioner och förslag på nya definitioner. "Standard Glossary of Terms used in Software Testing, Version 2.0", är den ordlista som internationellt ersätter BS 7925-2. Denna ordlista utgår ifrån och är en översättning av terminologi i "Standard Glossary of Terms used in Software Testing, Version 2.1."

1 Introduktion

Att det är viktigt att bli förstådd när man kommunicerar med kolleger inom sitt yrke är självklart för alla. Lika viktigt är det att bli förstådd vid kommunikation utanför sitt yrkesområde. Exempel på sådan kommunikation är kontakt med kunder och andra yrkesgrupper eller kommunikation mellan intressenter inom industri och den akademiska världen.

Om missförstånd uppstår kan det bero på att deltagarna har olika uppfattningar om vad ord betyder och vilken definition som bör användas för ordet. Sådana missförstånd kan leda till ökade kostnader och tvetydigheter i skrivna dokument. Därför är det viktigt att kommunikationen utgår från en gemensam terminologi.

Syftet med denna ordlista är att den ska kunna användas som underlag för en gemensam svensk terminologi vid kommunikation där det annars är risk för missförstånd mellan intressenter.

2 Omfattning

Terminologin i denna ordlista är begränsad till:

- terminologi som används i *ISTQB/SSTB kursplan för certifierad testare grundnivå version 2007* utgiven av Swedish Software Testing Board, SSTB, en del av ISTQB
- terminologi som används vid kommunikation mellan intressenter inom området programvarutestning och relaterade discipliner

3 Disposition

Ordlistan är uppbyggd enligt följande mall:

Term: Förklaring.

Kommentar: Förtydligande.

Term: det ord som förklaras.

Förklaring: en beskrivande text (definition) som förklarar ordet. En förklaring kan vara en hänvisning till ett annat ord. Definitionen gäller då för alla de ord som kan följas via hänvisningen och orden kan användas som synonymer. Exempel: **acceptans:** Se *acceptanstestning*.

I början på en förklaring kan ett eller flera alternativa ord finnas angivna. Exempel på alternativt ord är:

- ett ord som är mindre förekommande men ändå tillräckligt viktigt för att nämnas i ordlistan:
 - **avvikelse:** även incident, ...
- ett ord där det finns ett svenskt ord men en översättning är olämplig (även motsatsen kan gälla här, en översättning till svenskt ord kan göras men ett engelskt kan också användas):
 - **finess:** även feature, ...

I slutet av en förklaring kan ett eller flera relaterade ord finnas angivna. Ett relaterat ord är ett ord som sätter det förklarade ordet i ett sammanhang för ökad tydlighet. Exempel:

feltolerans: ... Se även tillförlitlighet.

Kommentar: ett förtydligande där:

- det allmänt förekommer missuppfattningar,
- det finns tvetydigheter i hur terminologin används,
- det finns skillnader i hur terminologin används i Sverige jämfört med övriga länder.

4 Normgivande referenser

Nedanstående referenser har använts som utgångsmaterial för denna ordlista. Dessa referenser var vid publicering av denna ordlista de senast gällande. Vid användning av denna ordlista bör hänsyn tas till om det har utkommit nyare versioner av nedanstående standarder.

- BS 7925-2:1998. *Software Component Testing*

- DO-178B:1992. *Software Considerations in Airborne Systems and Equipment Certification, Requirements and Technical Concepts for Aviation* (RTCA SC167)
- IEEE 610.12:1990. *Standard Glossary of Software Engineering Terminology*
- IEEE 829:1998. *Standard for Software Test Documentation*
- IEEE 1008:1993. *Standard for Software Unit Testing*
- IEEE 1012:1986. *Standard for Verification and Validation Plans*
- IEEE 1028:1997. *Standard for Software Reviews and Audits*
- IEEE 1044:1993. *Standard Classification for Software Anomalies*
- IEEE 1219:1998. *Software Maintenance*
- ISO/IEC 2382-1:1993. *Data processing - Vocabulary - Part 1: Fundamental terms*
- ISO 9000:2000. *Quality Management Systems – Fundamentals and Vocabulary*
- ISO/IEC 9126-1:2001. *Software Engineering – Software Product Quality – Part 1: Quality characteristics and sub-characteristics*
- ISO/IEC 12207:1995. *Information Technology – Software Life Cycle Processes*
- ISO/IEC 14598-1:1996. *Information Technology – Software Product Evaluation - Part 1: General Overview*

5 Varumärken

I detta dokument används följande registrerade varumärken:

- *CMM* och *CMMI* är registrerade varumärken av Carnegie Mellon University
- *TMap*, *TPA*, *TPI* och *TPI Next* är registrerade varumärken av Sogeti Nederland BV
- *TMM* är registrerat varumärke av Illinois Institute of Technology
- *TMMi* är registrerat varumärke av TMMi Foundation

6 Definitioner

A

abstrakt testfall: Se högnivåtestfall.

acceptans: Se acceptanstestning.

acceptanskriterier: De sluttillstånd eller krav som en komponent eller ett system ska uppnå för att accepteras av en användare, kund eller annan auktoriserad person eller organisation. [IEEE 610]

acceptanstestning i driftsmiljö: Acceptanstestning i driftsmiljö utförs av användare/kund i deras miljö. Användare/kund avgör om komponenten eller system fyller deras behov och passar in i deras affärsmodell. Normalt tar man hänsyn till både hårdvara och programvara.

acceptanstestning: Formell testning med inriktning på användarbehov, användarkrav och användarens verksamhet. Avsikten är att bedöma om ett system uppfyller acceptanskriterier och för att ge användare, kunder och annan auktoriserad person eller organisation möjlighet att acceptera systemet. [Efter IEEE 610]

ad hoc-granskning: Se informell granskning.

ad hoc-testning: även slumpmässig testning. Testning där man inte använder någon speciell testteknik, metod, förberedelse eller förväntat resultat, utan slumpen avgör vad som testas.

administrationsgranskning: En systematisk granskning av programinköp, orderhantering, utveckling, drift och underhåll som övervakar vidareutvecklingen och konstaterar krav och tillhörande systembehov. Utvecklar effektiviteten hos administrativa rutiner och hanteringsprocesser för att de ska uppnå bästa resultat för sitt syfte. [Efter IEEE 610, IEEE 1028]

agerande (IDEAL): Den fas inom IDEAL modellen i vilken förbättringar identifieras, utvecklas och genomförs inom organisationen. Agerande-fasen består av aktiviteterna: skapa lösning, utvärdera lösning, förfina lösning och implementera lösning. Se även IDEAL.

agil programvaruutveckling: En grupp av programvaruutvecklingsmetoder baserade på iterativ, inkrementell utveckling, där krav och lösningar växer fram genom samarbete mellan självorganiserande tvärfunktionella team.

agil testning: Ett tillvägagångssätt för test i ett projekt som använder agila programutvecklingsmetodiker, inklusive tekniker och metoder, som exempelvis extreme programming (XP), där utveckling ses som avnämning till test och betonar paradigmet testa först. Se även testdriven utveckling.

agilt manifest: En beskrivning av de gemensamma värden som all agil programvaruutveckling vilar på:

- Individer och samspel framför metoder, processer och verktyg
- Körbar programvara framför omfattande dokumentation
- Kundsamarbete framför kontraktsförhandlingar
- Anpassning till förändring framför följandet av en plan.

aktionsord-driven testning: Se nyckelordsdriven testning.

aktör: Användare, annan person eller system som samspelar med systemet som testas på ett specificerat sätt.

alfatestning: Simulerad eller verklig driftstestning av potentiella användare, kunder eller ett oberoende testteam hos leverantören, men utanför utvecklingsorganisationen. Alfatestning används ofta för standardprogramvara som en form av intern acceptanctestning hos leverantören.

algoritmstyrd testning: [TMap] Se bågtestning.

allvarlighet: Den grad av påverkan som ett fel har på utveckling eller drift av en komponent eller system. [Efter IEEE 610]

analys av grundorsak: En analysmetod som fokuserar på att identifiera grundorsaken för ett fel. Genom att koncentrera mätningar för korrigeringar hoppas man på att minimera sannolikheten för återkomst av felet.

analys av risker som kan leda till skada eller fara för liv: En teknik som används för att utmärka riskbenägna delar. Resultatet av analysen pådriver använda metoder för utveckling och testning av system. Se även riskanalys.

analys baserat på orsak-verkangrafer: Se metod för att skriva testfall genom orsak-verkangrafer.

Kommentar: Analys baserat på orsak-verkangrafer resulterar i beslutstabeller och som kan användas som testteknik.

analyserbarhet: Möjligheten att avgöra felorsak i ett system genom att diagnosticera av avvikelser eller orsaker till felsymptom, d.v.s. lokalisera fel i programvaran, och därmed göra en specifik rättning av felet utan att införa sidoeffekter. [ISO 9126] Se även underhållbarhet.

analysverktyg: Se verktyg för statistisk analys.

analytisk testning: Testning baserat på en systematisk analys av t.ex. produktrisker eller krav.

angreppssätt för test: Implementering av teststrategi för ett specifikt projekt. Vanligen baseras detta angreppssätt på projektmålen och systemets identifierade risker, där man anpassar testprocess, väljer testområden och testtekniker samt testfall och beslutar om vilka slutkriterier som gäller.

anomali: Något tillstånd som avviker från det förväntade, vilket baseras på exempelvis kravspecifikationer, designspecifikationer, användardokumentation, standarder alternativt personliga uppfattningar eller erfarenheter.

Anomalier kan hittas vid t.ex. granskning, testning, analys, kompilering, eller användning av programvaruprodukt eller tillhörande dokumentation. [IEEE 1044] Se även defekt, avvikelse, fel, felsymptom, incident, problem.

Kommentar: Se mer information om felsymptom. Speciellt används termen av IEEE om feltyper inom många yrkesgrenar. Det är ett bra samlingsbegrepp för åsikter, avvikelser och felsymptom, som ännu inte är klassificerade. Se IEEE Std. 1012 och IEEE Std. 1044.

anpassad programvara: Se kundanpassad programvara.

anpassat verktyg: Ett programverktyg utvecklat speciellt för en grupp användare eller kunder.

anpassningsbarhet: Programvaruproduktens kapacitet att anpassas till olika omgivningar och miljöer utan att programvaran behöver ändras nämnvärt på programvaran. [ISO 9126] Se även flyttbarhet.

anropsgraf: En abstrakt representation av anropsstrukturerna hos subrutinerna i ett program.

anti-mönster: Upprepad handling, process, struktur eller återanvändbar lösning som från början verkar vara till nytta och används ofta, men är ineffektiv och/eller kontraproduktiv i praktiken.

användaracceptanstestning: Se acceptanstestning.

användarbaserad kvalitet: Ett sätt att se på kvalitet som innebär att kvaliteten hos en produkt definieras av produktens förmåga att uppfylla användarens behov och önskningar. En produkt som inte uppfyller användarnas behov kommer troligtvis inte att ha några användare. Denna vy av kvalitet är kontext och tillfällighetsberoende eftersom olika affärssituationer kräver olika egenskaper hos en produkt. [Efter Garvin] Se även produktbaserad kvalitet, tillverkningsbaserad kvalitet, upplevd kvalitet och värdebaserad kvalitet.

användarberättelse : Ett användar- eller verksamhetskrav på hög nivå. Vanligt förekommande i agil mjukvaruutveckling, vanligen bestående av en eller flera meningar i vardagligt eller affärsspråk som fångar den funktionalitet en användare behöver, icke-funktionella kriterier och som även innehåller acceptanskriterier. Se även agil mjukvaruutveckling, krav.

användarberättelse-testning: En black-box-teknik där testfallen är designade baserade på användarberättelser för att verifiera att de implementerats korrekt. Se även användarberättelse.

användartest: Tester där användare från aktuell verksamhet är inblandade för att utvärdera användbarheten hos komponenten eller systemet.

användbarhet: Programvarans kapacitet att förstås, läras in, användas av och vara attraktiv för användaren när den används under specificerade förhållanden [ISO 9126]

användbarhetstestning: Testning av hur svår en produkt är att lära sig använda och om produkten löser den uppgift den är avsedd att lösa. [Efter ISO 9126]

användningsfall: En sekvens av transaktioner i en dialog mellan en aktör och en komponent eller ett system med ett verkligt resultat, där en aktör kan vara en användare eller en godtycklig entitet som kan utbyta information med systemet.

Kommentar: Varje användningsfall har förutsättningar, som skall vara uppfyllda för att användningsfallet skall fungera korrekt. Dessutom avslutas varje användningsscenario med sluttillstånd, vilka är de observerbara resultaten och det slutgiltiga tillståndet hos systemet då scenariot är exekverat.

användningsfallsbaserad testning: En black-box-testdesignsteknik där testfall designas för att exekvera användningsfall.

API-testning: Testa den kod som möjliggör kommunikation mellan olika processer, program och/eller system. API-testning innebär ofta negativa tester, t.ex. för att validera stabiliteten i felhantering. Se även gränssnittstestning.

aptestning: Testning med slumpvis urval av ett stort omfång invärden och med slumpvisa knapptryckningar utan hänsyn till hur produkten normalt används.

attack: Ett försök, inriktat på att utvärdera kvaliteten hos ett testobjekt genom att framtvinga specifika felsymptom. Se även negativ testning.

attackbaserad testning: En erfarenhetsbaserad testteknik där man använder sig av programvarubaserade attacker för att framkalla fel, i synnerhet säkerhetsrelaterade fel. Se även attack.

automatiserad testexekvering: Att kontrollera testexekvering samt jämföra testutfall med förväntat resultat, sätta upp förutsättning, annan teststyrning och rapportfunktioner med hjälp av en programvara.

automatiserad testvara: Testvara som används i automatiserad testning, t.ex. testskript.

avbrottskriterier: Kriterier som används för att (tillfälligt) stoppa alla eller delar var testaktiviteterna för testobjektet.

avlusare: även debugger. Se avlusningsverktyg.

avlusning: även debuggning. Processen att hitta, analysera och avlägsna orsaker till felsymptom i programvara.

avlusningsverktyg: även debuggningsverktyg. Ett verktyg som används av programmerare för att återskapa felsymptom och undersöka program i syfte att hitta var det finns fel som orsakar felsymptom. Dessa verktyg gör det möjligt för utvecklare att undersöka program rad för rad, att stanna på en viss kodrad i programmet och att undersöka programvariabler.

avslutskriterier: En uppsättning generiska och specifika villkor som har överenskommit med intressenter och som måste vara uppfyllda för att en process officiellt ska anses vara klar. Syftet med avslutskriterier är att förhindra att ett arbete anses avslutat när det fortfarande finns delar som inte är avslutade. Avslutskriterier används inom test för att planera och rapportera när testningen skall avslutas. [Efter Gilb and Graham]

avslutspunkt: En exekverbar kodsats eller processteg som utgör slutpunkt för en given process.

avvikelse: Händelser som sker vid testning och som kräver vidare utredning. [Efter IEEE 1008]

avvikelsehantering: Se felhantering (eng. defect management).

avvikelsehanteringsverktyg: Se felhanteringsverktyg (eng. defect management tool).

avvikelseloggning: Att skriva logg över händelser eller avvikelser, ibland även händelseloggning.

avvikelserapport: Se eng. "incident report".

avvikelse rapport: Även felrapport. Ett dokument som rapporterar om en avvikelse eller händelse som inträffat under t.ex. testning och som kräver en utredning. [Efter IEEE 829]. Se även felrapport.

B

balanserat styrkort: Ett strategiskt prestandamätningssverktyg som används för att undersöka huruvida ett företags aktiviteter är i linje med dess mål i termer av affärsvision och strategi. Se även corporate dashboard, dashboard och styrkort.

basblock: En nod i en kontrollflödesgraf representerar ett basblock.

begränsad tillståndsmaskin: En datamodell som består av ett begränsat antal tillstånd och tillståndsövergångar mellan dessa tillstånd, möjligen med medföljande handlingar (action). [IEEE 610]

beslut-/villkorstestning: En white-box-testdesigntechnik där testfall är utvecklade för att exekvera utfall från villkor och utfall från kodvillkor.

beslutstabell för orsak-verkan relationer: Se beslutstabell.

beslutstabell: En tabell som visar invärden och/eller stimuli med motsvarande utvärden och/eller funktion (verkan). Används för att ta fram testfall.

beslutstestning: En white-box-testdesigntechnik där testfall tas fram för att exekvera beslutsvägar i programkoden.

beslutstäckning: Den procentuella del av kodvillkor som har exekverats av en testsvit. 100 % beslutstäckning innebär både 100 % bågäckning och 100 % kodsatstäckning.

best practice: En överlägsen metod eller ett uppfinningsrikt arbetssätt som bidrar till en förbättrad prestation inom en organisation. Bygger på en erfarenhetsbaserad bedömning gällande vilket av flera möjliga arbetssätt/angreppssätt som leder fram till avsett resultat.

betatestning: även fälttestning. Drifttestning som utförs på plats hos slutkund eller slutanvändare där man låter kunden vara en del av "testorganisationen". Syftet är att ta reda på om en komponent eller ett system uppfyller användarens/kundens behov och passar in i den verksamhet som det ska användas i.

Kommentar: Notera att det kan förekomma att betatestning används synonymt med acceptanstestning, men syftet med ett betatest behöver inte vara acceptanstestning.

beteende: Komponentens eller systemets svar på en uppsättning av invärden och förutsättning.

big-bangtestning: En integrationstestningsstrategi där alla komponenter av ett system testas tillsammans på en gång utan att ha integrationstestats i mindre delar innan, t.ex. som delsystem i förväg. [Efter IEEE 610] Se även integrationstestning.

black-box-teknik: Se black-box-testdesigntechnik.

black-box-testdesigntechnik: Procedur för att skapa och välja testfall för en komponent eller ett system utan referens till dess interna struktur. En black-box-testdesigntechnik bygger på analys av specifikationen som kan vara funktionell eller icke-funktionell.

Kommentar: Kallas även specifikationsbaserad teknik, vilket bland annat omfattar ekvivalensklassindelning, testning med hjälp av beslutstabeller, tillståndsbaserad testning och användningsfallsbaserad testning.

black-box-testning: Testning, funktionellt eller icke-funktionellt, utan referens till den interna strukturen hos en komponent eller ett system.

Kommentar: Test som utförs UTAN vetskap om hur ett system eller en komponent är uppbyggd eller vad det består av. Det engelska ordet är ett inarbetat begrepp.

blockerat testfall: Ett testfall som inte kan exekveras, därför att förutsättningarna för testfallet inte är uppfyllda.

buffert: Utrustning eller ett lagringsutrymme för tillfällig lagring av data. Lagringen sparas i form av skillnader i frekvens av dataflöde, skillnad tid för en händelse eller skillnad i mängden av data som kan hanteras av enheten eller den process som flyttar eller använder data. [IEEE 610]

buffertöverskridande: En minnesåtkomstavvikelse som orsakas av att en process försöker lagra data utanför gränserna av en begränsad buffert som resulterar i en överskrivning av intilliggande minnesareor eller en undantagshantering. Se även buffert.

bugg: Se fel.

bugghanteringsverktyg: Se felhanteringsverktyg.

buggrapport: Se felrapport.

buggtaxonomi: Se defekttaxonomi.

burndown chart: Ett publikt diagram som visar den återstående insatsen kontra tiden i en sprint (iteration). Det visar status och trend för färdigställande av uppgifterna i sprinten. X-axeln representerar vanligen dagarna i sprinten, medan Y-axeln visar återstående insats (vanligen antingen i idealarbetsstimmar eller poäng).

BVT: Build verification test. Se verifiering av bygge.

bågtestning: En strukturell (white-box) testdesigntechnik där testfall utvecklas för att exekvera kodgrenar.

bågtestning: Se eng. "branch testing".

bågtäckning: De procentuella delarna av alla kodgrenar som har exekverats av en testsvit. 100% bågtäckning omfattar både 100% beslutsstäckning och 100% kodsatstäckning.

C

Capability Maturity Model Integration: Ett ramverk som beskriver de viktigaste delarna av en effektiv produktutveckling och underhållsprocess. Capability Maturity Model Integration täcker bästa praxis för planering, konstruktion och hantering av produktutveckling och underhåll. [CMMI]

CASE: Datorstödd programutveckling. Förkortning från engelska termen Computer Aided Software Engineering.

CAST: Datorstödd programvarutestning. Förkortning från engelska Computer-Aided Software Testing. Se även testautomatisering.

CCB (Configuration Control Board): även ändringsråd. En grupp människor som är ansvariga för att utvärdera och godkänna eller förkasta föreslagna ändringar av konfigurationsobjekt och för att säkra införandet av godkända ändringar. [IEEE 610]

certifiering: Processen att bekräfta att en komponent, ett system eller person uppfyller specificerade krav, t.ex. genom att klara en examen.

charter: Se test charter.

checklistebaserad testning: En erfarenhetsbaserad testdesigntechnik i vilken en erfaren testare använder en lista som innehåller punkter som ska noteras, kontrolleras eller kommas ihåg. Listan kan även innehålla regler eller kriterier mot vilka en produkt ska verifieras.

clear-box-testning: Se white-box testning.

CMMI: Se Capability Maturity Model Integration.

corporate dashboard: En presentation av företagets status med hjälp av ett instrumentpanelliknande verktyg. Se även balanserat styrkort, styrkort och dashboard.

COTS: Förkortning från engelska namnet Commercial Off-The-Shelf software. Se även kommersiellt från hyllan.

Critical Testing Processes: En innehållsbaserad modell för testprocessförbättring som bygger på tolv tydliga delprocesser. Bland dessa ingår de mest kritiska processerna för företagets vinst och rykte och används av ledning och medarbetare för att bedöma företagets kompetens och prestanda. Se även innehållsbaserad modell.

CTP: Se Critical Testing Processes.

cyklomatisk komplexitet: Det totala antalet oberoende exekveringsvägar genom ett program.

Cyklomatisk komplexitet definieras som: $L - N + 2P$, där L = antalet bågar i en graf, N = antalet noder i en graf och P = antalet separerade delar i en graf (t.ex. en anropad graf och en subrutin). [Efter McCabe]

cyklomatiskt mått: Se cyklomatisk komplexitet.

D

dagligt bygge: En utvecklingsaktivitet där ett komplett system kompileras och länkas varje dag (oftast nattetid). Ett dagligt bygge möjliggör tillgång till ett konsekvent system där alla de senaste ändringarna ingår.

dashboard: En representation av dynamiska mätningar av operativ prestanda för en organisations aktiviteter, via mätetal representerade av metaforer som t.ex. "visare", "räknare" och "indikatorer" liknande de som kan ses på instrumentbrädan hos en bil. Detta gör att effekterna av händelser och aktiviteter enkelt kan förstås och relateras till operativa mål. Se även corporate dashboard, styrkort och balanserat styrkort.

datadefinition: En exekverbar sats där en variabel tilldelas ett värde.

datadriven testning: En testteknik där testdata, dvs. invärden till testning, separeras från testfall. Ett vanligt tillvägagångssätt är att lägga de separerade invärdena i en tabell eller ett kalkylblad, och sedan låta mer generellt skrivna testfall läsa och använda dem. Datadriven testning används ofta för att stödja användning av testexekveringsverktyg t.ex. in- och uppspelningsverktyg. [Fewster and Graham] Se även nyckelordsdriven testning.

dataflöde: En abstrakt representation av en sekvens och möjliga förändringar av tillståndet hos ett dataobjekt där tillståndet är ett av följande: skapa (eng. creation), använda (eng. usage) eller destruktion (eng. destruction). [Beizer]

dataflödesanalys: En form av statistisk analys som grundar sig på tilldelning och användning av variabler.

dataflödestäckning: Den procentuella andel av alla definitions- och användningspar som har exekverats i en testsvit.

dd-path: En väg mellan två beslutpunkter i en algoritm, eller två beslutsnoder i en motsvarande graf, som inte innehåller några andra beslut. Se även väg.

Defect Detection Percentage (DDP): Antalet fel funna i en testfas dividerat med summan av dessa fel och fel funna i senare testfaser eller under andra situationer, t.ex. drift.

defekt: Se fel.

defektbaserad teknik: Se defektbaserad testdesigntechnik.

defektbaserad testdesigntechnik: En procedur för att basera och/eller välja testfall för en eller flera defekttyper, där test utvecklas utifrån vad som är känt för en specifik defekttyp. Se även defekttaxonomi.

defektkategori: Se defekttyp.

defektmaskering: Se felmaskering.

defekttaxonomi: Ett system av (hierarkiskt ordnade) kategorier designade att vara användbara i arbetet med att (reproducerbart) klassificera defekter.

defekttäthet: Antal fel funna i en komponent eller ett system dividerat med storleken på komponenten eller systemet (uttryckt i standardmåttetermer, t.ex. kodrader, antal klasser eller funktionsenheter).

definitions- och användningspar: Relationen mellan en definition och en användning av en variabel. Användning av en variabel ingår antingen i en beräkning eller i ett villkor.

delväg: En sekvens av exekverbara satser inom en komponent.

Deming-cykel: En iterativ problemlösningssprocess i fyra steg (planera-utföra-kontrollera-agera) som ofta används vid processförbättring. [Efter Deming]

designbaserad testning: Ett angreppssätt för testning där utvecklingen av testfall baseras på arkitekturen och/eller detaljerad design av komponenten eller systemet (t.ex. gränssnitt mellan komponenter eller system).

destruktiv testning: Se negativ testning.

detaljerad integrationstestning: Se komponentintegrationstestning.

diagnostisering (IDEAL): Den fas inom IDEAL modellen i vilken man fastställer var man är i förhållande till var man vill vara. Diagnostiseringsfasen består av aktiviteterna: karakterisera nuvarande och önskade tillstånd samt utveckla rekommendationer. Se även IDEAL.

domän: Den uppsättning data från vilken giltiga in- och/eller utdata kan väljas. Domän betyder också område/gebit, t.ex. inom bank, finans, inbyggda system.

domänanalys: En black-box-testdesigntechnik som används för att identifiera ändamålsenliga och effektiva testfall när flera variabler kan eller bör provas tillsammans. Den bygger på och generaliserar ekvivalensklassindelning och gränsvärdesanalys. Se även ekvivalensklassindelning, gränsvärdesanalys.

driftacceptanstestning: Drifttestning i acceptanstestfasen, vanligtvis utförd i en (simulerad) driftsmiljö av drifts- eller administratörspersonal, och fokuserad på driftsaspekter som t.ex., återhämtningsförmåga, beteende hos resurser, installationsbarhet och teknisk överensstämmelse. Se även drifttestning.

driftsmiljö: En miljö bestående av hårdvaru- och programvaruprodukter, regler och rutiner som är installerade hos kund, eller en miljö där komponenten eller systemet som testas ska användas senare. Programvaran kan också omfatta operativsystem, databashanteringssystem och andra applikationer.

drifttestning: Testning utförd i syfte att utvärdera en komponent eller ett system i dess driftsmiljö. [IEEE 610]

drivrutin: även driver. En programvarukomponent eller ett testverktyg som ersätter en komponent och tar kontroll och/eller anrop till en komponent eller system. [Efter TMap]

dynamisk analys: Processen att utvärdera ett system eller en komponent baserat på dess beteende under exekvering. [Efter IEEE 610] Motsats: statisk analys.

dynamisk jämförelse: Jämförelse av aktuellt och förväntat resultat som görs medan programmet exekveras, t.ex. med hjälp av ett testexekveringsverktyg.

dynamisk testning: Testning som innefattar exekvering av programvaran till en komponent eller ett system.

dynamiskt analysverktyg: Ett verktyg som tillhandahåller realtidsinformation om tillståndet för programvaran. Dessa verktyg används oftast för att identifiera ej tilldelade pekare, visa minnesallokering och flagga för minnesläckor.

död kod: Se onåbar kod.

E

effektivitet: (1) Programvarans förmåga att tillhandahålla tillräckliga prestanda i relation till använda resurser och under definierade villkor. [ISO 9126] (2) Förmågan hos en process att producera avsett resultat, i förhållande till den mängd resurser som används.

effektivitetstestning: Testning som utförs för att avgöra hur tidseffektiv en programvaruprodukt är.

EFQM (European Foundation for Quality Management) excellensmodell: Ett icke-normativt ramverk för en organisations kvalitetsledningssystem, framtaget och ägt av European Foundation for Quality Management, baserat på fem "aktiveringskriterier" (vilka täcker vad organisationen gör) och fyra "resultatkriterier" (vilka täcker vad organisationen uppnår).

ekvivalensklass: Se ekvivalensuppdelning.

ekvivalensklassindelning: En black-box-testdesignsteknik där testfall har utvecklats för att exekvera representativa värden i ekvivalensklasser. I princip tas testfall fram för att täcka varje ekvivalensklass åtminstone en gång.

ekvivalensklassäckning: Den procentuella delen av alla ekvivalensklasser som har blivit täckta av en testsvit.

ekvivalensuppdelning: även ekvivalensindelning. Ett område i en in- eller utmatning där ett testobjekt kan antas ha samma beteende inom området, där antagandet kan vara baserat på en specifikation eller vara underförstått av sammanhanget.

emotionell intelligens (EQ): Förmåga eller skicklighet att identifiera och hantera sinnesstämningar hos sig själv, hos andra individer och hos grupper.

emulator: Utrustning, program eller system som accepterar samma indata och producerar samma utdata som ett bestämt system. [IEEE 610] Se även simulator.

Engagemangsmatris (RACI-matris): Utförandeansvarig (Responsible), Medansvarig (Accountable), Konsulterad (Consulted) och Informerad (Informed)

enhet: Se komponent.

enhetstestning: Se komponenttestning.

enhetstestramverk: Ett verktyg som erbjuder en miljö för en komponent eller enhet, så att den kan testas enskilt eller med lämpliga stubbar eller drivrutiner. Ramverket erbjuder också stöd för utvecklarerna, i form av t.ex. avlusningsmöjligheter. [Graham]

enskild testning: Testning av enskilda komponenter skilda från intilliggande komponenter, där de intilliggande komponenterna simuleras av stubbar eller drivrutiner om nödvändigt.

erfarenhetsbaserad teknik: Se erfarenhetsbaserad testdesignsteknik.

erfarenhetsbaserad testdesignsteknik: En testdesignsteknik där man inte följer någon speciell testdesignsteknik, utan använder sin samlade kunskap. Se även felgissning.

erfarenhetsbaserad testning: Testning baserad på testarens erfarenhet, kunskap och intuition.

ersättningsbarhet: Kapaciteten hos programvaruprodukten att användas istället för en annan specificerad programvaruprodukt för samma ändamål och i samma miljö. [ISO 9126] Se även portabilitet.

etablering (IDEAL): Den fas inom IDEAL modellen i vilken man planerar detaljerna för hur en organisation ska kunna nå sina mål. Etableringsfasen består av aktiviteterna: prioritera, utveckla tillvägagångssätt och planera åtgärder. Se även IDEAL.

exakthet: även precision, noggrannhet. Möjligheten, förmågan hos systemet (inkluderat programvara, data och hårdvara) att erbjuda rätt och exakt det resultat och den effekt som eftersträvas. [ISO 9126] Se även funktionalitetstestning.

exekverbar sats: En sats som när den kompileras översätts till objektкод som kan utföra händelser, hantera data eller utföra en beräkning. En exekverbar sats kommer att exekveras procedurmässigt när programmet exekveras.

exekveringsstartpunkt: En exekverbar kodsats eller processteg som utgör startpunkt för en given process.

extreme programming (XP): En programvaruutvecklingsmetodik, som används inom agil programvaruutveckling, i vilken de centrala aktiviteterna innefattar parprogrammering, omfattande kodgranskning, enhetstestning av all kod samt enkel och tydlig kodning. Se även agil programvaruutveckling.

F

fabriksacceptanstestning: Acceptanstestning genomförs vid den plats där produkten utvecklas och utförs av anställda inom leverantörens organisation, för att avgöra om en komponent eller ett system uppfyller kraven, normalt inklusive hårdvara samt mjukvara. Se även alfatestning.

failover testning: Testning genom att simulera fellägen eller faktiskt orsaka fel i en kontrollerad miljö. Efter ett misslyckande testas failovermekanismen för att säkerställa att data inte förloras eller förvanskas och att alla överenskomna servicenivåer upprätthålls (t.ex. funktionstillgänglighet eller svarstider). Se även återhämtningsbarhetstestning.

Failure Mode and Effect Analysis (FMEA): Ett systematiskt angreppssätt för att identifiera och analysera möjliga risker för felsymptom/felyttringar och för att förhindra att de uppstår. Se även Failure Mode, Effects and Criticality Analysis (FMECA).

Failure Mode, Effects and Criticality Analysis (FMECA): En utökning av FMEA där utökningen inkluderar en kritikalitetsanalys som används för att påvisa/kartlägga sannolikheten av att felsymptomsläge inträffar och allvarligheten i konsekvens av ett felsymptomsläge. I resultatet framhävs felsymptomslägen med relativ hög sannolikhet och allvarlighet i konsekvenserna, med hjälp av resultatet kan stödresurser och insatser fokuseras där de har störst värde. Se även FMEA (Failure Mode and Effect Analysis).

faktiskt resultat: Det resultat som produceras eller observeras när en komponent eller ett system testas.

faktiskt utfall: Se faktiskt resultat.

falsk slumpmässighet: En serie data som är genererad som slumpmässig, men ofta med en inbyggd sekvens eller ordning.

falskt-godkänt resultat: Ett test resultat som misslyckas att identifiera närvaron av en defekt fastän defekten existerar i testobjektet.

falskt-negativt resultat: Se falskt-godkänt resultat.

falskt-positivt resultat: Se falskt-underkänt resultat.

falskt-underkänt resultat: Ett test resultat där en defekt är rapporterad fastän det inte existerar någon sådan defekt i testobjektet.

fasinneslutning: Den procentuella andelen av defekter som eliminerats i samma fas av livscykeln för programvaruutvecklingen där de uppstod.

fastestplan: En testplan för en specifik testnivå. Se även testplan.

fastställande av funktionskvalitet: En metod att omvandla användarkrav till designkvalitet, att fastställa de funktioner som utgör kvalitet och att fastställa metoder för att uppnå denna designkvalitet i delsystem och komponenter, och ytterst i specifika element i tillverkningsprocessen.

fastställd konfiguration: Sammanfattande begrepp för att publicera ingående granskade delar i ett system och delarnas versioner. Denna konfiguration fungerar sedan som grund för vidare utveckling och kan endast ändras efter en formell ändringsprocess. [Efter IEEE 610]

FDP (Fault Detection Percentage): Se DDP (Defect Detection Percentage).

fel: En brist i en komponent eller ett system som kan få komponenten eller systemet att misslyckas med att utföra sin önskade funktionalitet, t.ex. en felaktig kodsats eller datadefinition. Ett fel kan, om det uppträder under exekvering, orsaka ett felsymptom/en felyttring hos komponenten eller systemet.

felattack: Se attack.

felfrekvens: Förhållandet mellan felsymptom av en viss kategori och en given mätreferens, t.ex. felsymptom per tidsenhet, felsymptom per antal transaktioner, felsymptom per antal datorkörningar. [IEEE 610]

felgissning: En testdesigntechnik där testarens erfarenhet används för att förutse vilka fel som kan finnas i den komponent eller det system som testas. Tekniken bygger på tidigare fel och används för att utveckla testfall som visar dessa eventuella fel.

felhantering: Processen att identifiera, utreda och vidta åtgärder samt avföra fel. Den omfattar registrering och klassificera av fel samt fastställande av deras påverkan. [Efter IEEE 1044]

felhanteringsråd: En tvärfunktionell grupp intressenter som hanterar rapporterade fel från första upptäckt till slutliga lösning (avlägsning, uppskjutning eller annullering). I vissa fall samma grupp som CCB. Se även CCB (Configuration Control Board).

felhanteringsverktyg: Ett verktyg som underlättar registrering och tillståndsspårning av fel. I felhanteringsverktyg finns ofta stöd för att följa ett arbetsflöde och för styrning, tilldelning, åtgärd och omtestning av händelser eller fel, och det tillhandahåller hjälp för rapportering.

felinjektion: Processen att avsiktligt lägga in fel i ett system för att ta reda på om systemet kan upptäcka och eventuellt återhämta sig från felen. Felinjektion har för avsikt att efterlikna fel som kan uppstå i produktion. Se även feltolerans.

felinjektionsverktyg: Verktyg för att injicera (d.v.s. avsiktligt införa) fel i en komponent eller ett system.

felmaskering: Inträffar när ett fel döljer ett annat fel så detta inte upptäcks.

felplantering: Processen att avsiktligt lägga till fel utöver de som redan finns i en komponent eller ett system i syfte att utvärdera hur många fel som hittas och tas bort, och för att uppskatta hur många fel som finns kvar. Felplantering är oftast en del av utvecklingstestningen (på en förhandsversion) och kan utföras på alla testnivåer (komponent, integration eller system) [efter IEEE 610] 23

felprioriteringsråd: se felhanteringsråd

felrapport: Ett dokument som rapporterar en brist i en komponent eller ett system. Bristen kan medföra att komponenten eller systemet misslyckas med att utföra en önskad funktionalitet. [Efter IEEE 829]

felspårningsverktyg: Se felhanteringsverktyg.

felsymptom: även felyttring. Avvikelse från programvarans eller systemets förväntade beteende eller resultat. [Efter Fenton]

feltillstånd: Den fysiska eller funktionella yttringen av ett felsymptom. Ett system i felsymptomsläge kan t.ex. ge tecken på långsam funktion, felaktiga utvärden eller exekveringsavbrott. [IEEE 610]

feltolerans: 1) efter "fault tolerance": Kapacitet hos en programvaruprodukt att upprätthålla en specificerad prestandanivå då ett programvarufel uppträder eller då dess specificerade gränssnitt inte följs. [ISO 9126] Se även tillförlitlighet, robusthet.

2) efter "error tolerance": Kapacitet hos en komponent eller ett system att fortsätta normal drift trots att det finns felaktiga indata. [Efter IEEE 610]

felträdsanalys, FTA (Fault Tree Analysis): En teknik som används för att analysera orsakerna till fel (defekter). Tekniken modellerar visuellt hur logiska relationer mellan felsymptom, mänskliga misstag, och externa händelser kan i kombination orsaka specifika fel.

feltyp: En enhet i en defekttaxonomi (även defektklassificering). Defekttaxonomier kan identifieras med avseende på en mängd olika överväganden, inbegripet, men inte begränsat till:

- Fas eller utvecklingsaktivitet där felet skapas, t.ex. fel i specifikationen eller fel i koden
- Karaktärisering av defekter, t.ex. ett gränsvärdesfel
- Inkorrekthet, t.ex. en felaktig relationsoperator, ett syntaxfel för aktuellt programspråk eller ett ogiltigt antagande
- Prestandaproblem, t.ex. överdriven exekveringstid, otillräcklig tillgänglighet.

feltäthet: Se defekttäthet.

finess: även feature. Ett kännetecken hos en komponent eller ett system som har specificerade eller underförstådda krav (t.ex. tillförlitlighet, användbarhet eller designrestriktioner). [Efter IEEE 1008]

fiskbensdiagram: Se orsak-verkandiagram.

flexibilitet: En egenskap hos programvara att tillåta enkel implementering av specificerade ändringar. [ISO 9126] Se även underhållbarhet.

formell granskning: En granskning som innebär att man har dokumenterade procedurer och krav för granskning, t.ex. en inspektion.

FPA (Function Point Analysis): även funktionsgradsanalys. En metod som används för att mäta mängden funktioner i ett informationssystem. Mätningen är oberoende av teknologi. Den här mätningen kan användas som grund för mätning av produktivitet, resursbehov och projektstyrning.

fryst testbas: Ett testbasdokument som bara kan ändras genom en formell ändringshanteringsprocess. Se även fastställd konfiguration.

funktionalitet: Programvaruproduktens kapacitet att tillhandahålla funktioner som motsvarar ställda och underförstådda behov när programvaran används under definierade förhållanden. [ISO 9126]

funktionalitetstestning: Testning för att fastställa funktionaliteten hos en programvaruprodukt.

funktionell integration: Ett tillvägagångssätt för integration där komponenter eller system sammankopplas i syfte att få en basfunktionalitet att fungera tidigt. Se även integrationstestning.

funktionell testdesigntechnik: Proceduren att utveckla och/eller välja testfall baserat på en analys av specifikationen eller funktionaliteten hos komponenten eller systemet utan hänvisning till den interna strukturen. Se även black-box-testdesigntechnik.

funktionellt krav: Ett krav som specificerar en funktion som en komponent eller ett system måste utföra. [IEEE 610]

funktionsdriven utveckling: En iterativ och inkrementell programutvecklingsprocess som drivs från ett kundvärderat funktionsperspektiv. Funktionsdriven utveckling används oftast i agil mjukvaruutveckling. Se även agil programvaruutveckling.

funktionsduglighet: även praktisk användbarhet, driftsduglighet. Programvaruproduktens kapacitet att möjliggöra att en användare kan använda och behärska programmet. [ISO 9126] Se användbarhet.

funktionstestning: Testning av systemets funktionella egenskaper baserat på en analys av funktionsspecifikationer för en komponent eller ett system. Se även black-box-testning.

fälttestning: Se betatestning.

förståelighet: Programvaruproduktens kapacitet att hjälpa användaren att förstå huruvida programvaran är ändamålsenlig och hur den kan användas för speciella uppgifter och villkor. [ISO 9126] Se även användbarhet.

förtest: Se öppningstest.

förtroendeintervall: Den tidsperiod inom vilken en åtgärdsplan måste genomföras för att effektivt reducera konsekvensen av risken, vid hantering av projektrisker.

förtroendetest: Se röktest.

förutsägbart resultat: Ett resultat/beteende som är förutsagt/förväntat enligt en komponents specifikation under speciella villkor.

förutsägelse: En programsats som kan resultera i sant eller falskt och kan användas för att fastställa kontrollflödet i därpå följande beslutslogik. Se även kodvillkor.

förutsättning: Miljö- och tillståndsvillkor som måste vara uppfyllda före exekvering av ett test eller en testprocedur.

förväntat resultat: Predikterat beteende hos en komponent eller ett system under specificerade villkor som förutsagts i en specifikation eller annan källa.

förväntat utfall: Se förväntat resultat.

G

genomgång: En steg-för-steg-presentation utförd av författaren till ett dokument med syfte att samla information och skapa en gemensam förståelse av innehållet. [Freedman and Weinberg, IEEE 1028] Se även kollegial granskning.

glasbox-testning: Se white-box-testning.

Goal Question Metric: En programvarumätningsslag metod indelad i tre nivåer: begreppsmässig nivå (målorientering - "goal"), operativ nivå (frågeorientering - "question"), kvantifieringsnivå (mätorientering - "metric").

godkänd test: Se godkänd.

godkänd: även lyckad. Ett godkänt/lyckat test är ett test där det faktiska resultatet motsvarar det förväntade resultatet.

godkända/underkända kriterier: Beslutsregler för att avgöra om genomförda tester av ett testelement (funktion) eller feature ska godkännas eller underkännas. [IEEE 829]

GQM: Se Goal Question Metric.

grannskapsintegrationstestning: En form av integrationstester där alla knutpunkter (noder) som ansluter till en given knutpunkt (nod) är grunden för integrationstestning.

granskare: En person som deltar i en granskning och som ska identifiera och beskriva avvikelser/felaktigheter i produkten eller projektet som granskas. Granskare kan utses för att representera olika kompetensområden och roller i granskningsprocessen.

granskning: En utvärdering av en produkt eller systemstatus med syfte att upptäcka avvikelser från planerat resultat och för att rekommendera åtgärder. Exempel kan vara informella granskningar, inspektioner och genomgångar. [Efter IEEE 1028]

granskningsplan: Ett dokument som beskriver angreppssätt, resurser och schema för planerade granskningsaktiviteter. Det identifierar bland annat dokument och kod som ska granskas, granskningstyper som ska användas, deltagare, start och avslutskriterier som ska appliceras vid formella granskningar, och motivet för valet. Det är en dokumentation av granskningsplaneringsprocessen.

granskningsverktyg: Verktyg som stödjer granskningsprocessen. Typiska funktioner är granskningsplanering och spårning, stöd för kommunikation, stöd för granskningssamarbete och en databas för insamling och rapportering av mätvärden.

grundläggande jämförelsetestning: En black-box-designsteknik där testfallen är utvecklade för att exekvera en kombination av indata enligt begreppet MCDC täckning. [TMap]

grundorsak: En underliggande faktor som orsakar att specifikation och implementation inte stämmer överens.

gränssnittstestning: även gränssytetestning. En typ av integrationstest som berör testning av gränssytor mellan komponenter och system.

gränsvärde: In- eller utvärde som befinner sig på gränsen till en ekvivalensindelning, eller på det minsta avståndet från någon av gränserna, t.ex. ett min- eller maxvärde för ett område.

gränsvärdesanalys: En black-box-testdesignsteknik som används för att utveckla testfall som avser att testa gränsvärden. Se även gränsvärde.

gränsvärdestestning: Se gränsvärdesanalys.

gränsvärdestäckning: Den procentuella täckningsgraden av gränsvärden som har exekverats av en testsvit.

H

heuristisk evaluering av användargränssnitt: En teknik för att bedöma användbarhet utifrån användbarhetsproblem eller design av användargränssnitt. Med denna teknik undersöker granskarna gränssnittet och bedömer dess överensstämmelse med erkända användbarhetsprinciper ("heuristiken").

horisontell spårbarhet: Spårning av krav för en testnivå genom de olika nivåerna av testdokumentation (t.ex. testplan, testdesign, testspecifikation, testfall, testprocedur och/eller testskript).

Kommentar: Betyder ofta att man i praktiken sätter ett unikt nummer för varje krav, (eller designobjekt o.s.v.) och varje dokument refererar tillbaka till det kravets unika nummer. Således kan man spåra ett testfall till dess ursprungliga krav. Notera att det är möjligt att skapa testfall som saknar direkt koppling till krav, designobjekt eller liknande.

hyperlänk: En länk (pekare) inom en webbsida som leder till en annan webbsida.

hyperlänk-testverktyg: Ett verktyg som kontrollerar att alla hyperlänkar är fungerande.

Kommentar: Ett hyperlänkverktyg kontrollerar att alla länkar leder till en existerande webbsida. Verktöget kan inte avgöra om det är rätt webbsida som länken leder till.

högnivåtestfall: Ett testfall utan konkreta värden (på implementationsnivå) för indata eller resultat. Se även lågnivåtestfall.

I

icke överensstämmande: Krav i kravspecifikation eller underförstådda krav som inte är uppfyllda. [ISO 9000]

icke-funktionellt krav: Krav som inte relaterar till funktionaliteten i ett program utan till egenskaper såsom tillförlitlighet, effektivitet, användbarhet, underhållbarhet och portabilitet.

IDEAL: En modell för organisationsförbättringar vilken tjänar som en guide vid initiering, planering och genomförande av förbättringsaktiviteter. IDEAL-modellens namn kommer av dess fem faser: initiering, diagnostisering, etablering, agerande och lärande.

in- och uppspelningsverktyg: Se in- och uppspelningsverktyg.

in- och uppspelningsverktyg: även skriptverktyg. Testverktyg där indata spelas in under manuell testning för att generera automatiska testskript som kan exekveras senare. Dessa verktyg används ofta för att stödja automatisk regressionstestning.

Kommentar: Ett bra sådant verktyg har även möjligheter att modifiera, dvs. ändra i den inspelade sekvensen.

inbäddad iterativ utvecklingsmodell: En underliggande livscykelmodell vid utveckling som tillämpar en iterativ metod för detaljerad design, kodning och testning inom en övergripande sekventiell modell. I detta fall är designdokument på hög nivå utarbetade och godkända för hela projektet men detaljerad design, kodning och testning sker i iterationer.

incident: Se avvikelse.

indata: En variabel som har tilldelats ett värde (oavsett hur den är lagrad eller åtkomstmetod) när den läses/används under exekveringen. Se även invärde.

indatadomän: Den uppsättning data från vilken giltiga indata kan väljas. Se även domän.

indikator för testprestanda: Ett måttetal på hög nivå som mäter effektivitet och används för att vägleda och reglera progressiv testutveckling, t.ex. Defect Detection Percentage (DDP). Se även DDP.

indikator: Måttetal som kan användas för att uppskatta eller förutsäga ett annat måttetal. [ISO 14598]

informationssäkerhet: Egenskaper hos en programvaruprodukt med avseende på dess förmåga att förhindra otillåten användning eller åtkomst till program eller data. [ISO 9126] Se även funktionalitet.

informationssäkerhetstestning: Testning av hur ett system klarar sin specificerade informationssäkerhet. Se även funktionalitetstestning.

informell granskning: En granskning som inte baseras på en formell (dokumenterad) procedur.

inhyrd testning: även insourced testning. Testning som utförs av människor som är samlokaliserade med projektgruppen, men inte arbetskamrater.

initiering (IDEAL): Den fas inom IDEAL modellen i vilken man lägger grunden för en framgångsrik förbättringsaktivitet. Initieringsfasen består av aktiviteterna: fastställ sammanhang, etablera stöd från ledningen samt fastställ nödvändig infrastruktur. Se även IDEAL.

injektion av felaktighet: Se felinjektion.

inkrementell testning: Testning där komponenter eller system integreras och testas, ett eller flera i taget, tills alla komponenter eller system har integrerats och testats.

inlärningsmöjlighet: Kapaciteten hos en programvaruprodukt att möjliggöra för användaren att lära sig dess funktionaliteter. [ISO 9126] Se även användbarhet.

innehållsbaserad modell: En processmodell som innehåller detaljerade beskrivningar av god branschpraxis, t.ex. testpraxis.

innehållsreferensmodell: Se innehållsbaserad modell.

inspektion: En typ av kollegial granskning som innefattar en visuell undersökning av dokument med syfte att hitta fel, t.ex. avvikelser från utvecklingsstandarder eller att dokumentet inte följer dokumentation från högre nivåer. Det är den mest formella granskningstekniken och baseras därför alltid på en dokumenterad procedur. [Efter IEEE 610, IEEE 1028] Se även kollegial granskning.

inspektionsledare: Se moderator.

inspektör: Se granskare.

installationsbarhet: Förmågan hos en programvaruprodukt att kunna installeras i en specificerad miljö. [ISO 9126] Se även portabilitet.

installationshandledning: Tillhandahållna instruktioner, på lämpligt medium, som leder installatören genom installationsprocessen. Det kan vara en manuell handledning, en steg-för-steg-procedur, automatisk hjälp eller liknande processbeskrivning.

installations-wizard: även installationsguide. Tillhandahållen programvara på lämpligt medium som leder installatören genom installationsprocessen. Normalt genomför programvaran installationsprocessen och återmatar information om installationsresultat samt visar valmöjligheter under installationens gång.

instrument: Ett verktyg för programvara som används för att utföra instrumenteringar av kod. Se även instrumentering.

instrumentering: Införande av extra kod i programmet för att göra det möjligt att skaffa information om programmets beteende under exekvering, t.ex. för att mäta kodtäckning.

integration: Processen att kombinera komponenter eller system till större grupper, objekt eller enheter.

integrationstestning för hård- och mjukvara: Testning utförd för att avslöja defekter i gränssnitt och samspel mellan hårdvara och mjukvara. Se även integrationstestning.

integrationstestning: Testning utförd med syfte att hitta fel i gränssnitt och kommunikation mellan programkomponenter, men även mellan program och maskinkomponenter som är integrerade. Integrationstest omfattar också i en V-modell att verifiera att designstrukturen motsvarar integrationen. Se även komponentintegrationstestning, systemintegrationstestning.

integritetsnivå för programvara: Den grad som programvaran uppfyller eller måste uppfylla en uppsättning av intressenter utvald programvara och/eller programvaruvarubaserade systemegenskaper (t.ex. programvarukomplexitet, riskbedömning, skydds- och säkerhetsnivå, förväntat beteende, tillförlitlighet, eller kostnad) vilka anges för att reflektera över programvarans betydelse för dess intressenter.

interoperabilitet: Programvarans kapacitet att samverka med en eller flera specificerade komponenter eller system. [Efter ISO 9126] Se även funktionalitet.

interoperabilitetstestning: även samarbetstestning. Testning av systemet eller komponentens samverkanskapacitet. Se även funktionalitetstestning.

Kommentar: Förmågan hos två eller flera system eller komponenter att utbyta information och att sedan använda denna information, samt att systemen tolkar informationen på samma sätt.

invärde: Värden som ges till ett program, system eller testfall. Se även indata.

Kommentar: Ordets innebörd framgår ofta av sammanhanget, annars bör man specificera vilka indata eller invärden som avses.

Ishikawa-diagram: Se orsak-verkandigram.

J

jämförare: Se testkomparator.

jämförelse efter exekvering: även post-exekveringsjämförelse. Jämförelse mellan faktiskt resultat och förväntat resultat efter det att programvaran har kört färdigt.

K

klassifikationsträd: Ett strukturträd som visar ekvivalensklassfördelning i hierarkisk ordning. Trädet används för att designa testfall i klassifikations-träds metoden. Se även klassifikations-träds metoden.

klassifikationsträdsmetoden: En black-box-testdesignteknik i vilken testfall, beskrivna med utgångspunkt ur ett klassificeringsträd, är skapade att utöva kombinationer representativa för indata- och/eller utdata domäner. [Grochtmann]

kod: Datorinstruktioner och datadefinitioner uttryckta i ett programspråk eller i ett format utmatat av en assembler, kompilator eller annan översättare. [IEEE 610]

kodanalysator: Se verktyg för statisk kodanalys.

kodbaserad testning: Se white-box-testning.

Kommentar: Kodbaserad testning är en testmetod som syftar till att härleda testfall direkt från implementationen, d.v.s. programkoden, där tester exekverar specifikt kontrollflöde eller använder specifika data.

kodgren: även exekveringsväg. Ett basblock i ett program som kan väljas för exekvering där en av två eller flera alternativa exekveringsvägar är möjliga, t.ex. case, jump, goto och if-then-else.

kodgrensvillkor: Se villkor.

Kommentar: Ett villkor styr vilken kodgren (exekveringsväg) som skall väljas, och handlar om att testa utfallen av villkoret.

kodgrensvillkorstäckning: Se villkorstäckning.

kodsats: även kodrad, kodblock. Den minsta exekverbara enheten i programspråk.

kodsatstestning: även testning av kodsatser eller kodrader. En white-box-testdesignteknik där testfall är framtagna för att exekvera kodsatser

kodsatstäckning: även täckning av satser eller kodrader. Den del av exekverbara kodsatser som blivit exekverade av en testsvit, angivet i procent.

kodtäckning av kombinationer av kodgrenvillkor: Se kombinationsvillkorstäckning.

Kommentar: Kodtäckning innebär att ett eller flera testfall har utövat den del av koden som avses. För kombinationer av kodgrenvillkor innebär det specifikt alla kombinatoriska kombinationer (logiska) av villkoren samt att alla kodgrenar är täckta.

kodtäckning: En analysmetod som undersöker hur stor del av koden som har genomlöpts med de tester som har utförts. Genomlöst kod mäts i t.ex. kodsatstäckning, beslutstäckning eller villkorstäckning.

kodvillkor/villkorstäckning: Den procentuella andel av alla utfall från villkor och utfall från kodvillkor som har exekverats i en testsvit. 100 % kodvillkor/villkorstäckning inbegriper både 100 % villkorstäckning och 100 % beslutstäckning.

kodvillkor: En punkt i programmet där kodvillkoret leder till att en av två eller flera möjliga vägar väljs.

kollegial granskning: En granskning av en produkt, utförd av kolleger till den som tagit fram produkten, i syfte att hitta fel och föreslå förbättringar. Exempel är inspektion, teknisk granskning och genomgång.

kombinationstestning av kodgrenar och villkor: Se villkorskombinationstestning.

Kommentar: En testfallsdesignteknik där testfallen designas i syfte att testa alla kombinationer av villkor och kodgrenval.

kombinationsvillkor: Se multipla villkor.

kombinatorisk testning: Ett sätt att identifiera en lämplig delmängd av testkombinationer för att uppnå en förutbestämd nivå av täckning när man testar ett objekt med flera parametrar och där dessa parametrar själva var och en har flera värden, vilket ger upphov till fler kombinationer än det är möjligt att testa inom den föreskrivna tiden. Se även klassifikationsträdsmetoden, n-vis testning, parvis testning, testning baserad på ortogonalitetsmatris.

kombinerad villkorstäckning: Se villkorskombinationstäckning.

kommersiellt från hyllan: Se standardprogramvara.

Kommentar: Programvara som finns att köpa på marknaden. Även kallad tredje parts programvara. Kan vara ett kommersiellt verktyg, program eller produkt.

kompatibilitetstestning: Se interoperabilitetstestning.

Kommentar: Testning av egenskapen att systemet är kompatibelt med andra system. Systemkompatibilitet innebär att två eller flera system skall kunna samarbeta och kommunicera med varandra.

kompilerator: Ett program vars syfte är att översätta program från högnivåspråk till ett språk specifikt för varje maskin. [IEEE 610]

komplett testning: Se uttömmande testning.

komplett täckningsmängd: En mängd av testfall som skapas baserat på den interna strukturen hos en komponent eller specifikation för att säkerställa att 100 % täckningsgrad för ett specificerat täckningskriterium uppnås.

komplexitet: Svårighetsgraden att förstå, underhålla eller testa ett system eller en komponents design eller implementation. Se även cyklomatisk komplexitet.

Kommentar: Det finns många mätvärden i olika verktyg, specifikt statistiska analysverktyg, som påstås mäta någon aspekt av komplexitet.

komponent: En programvarudel som kan testas isolerat från övriga komponenter och isolerat från systemet.

Kommentar: Notera att det även finns mer uttömmande definitioner på vad som exakt måste finnas för att kalla en programdel för komponent inom vissa discipliner och domäner. Grundtanken ligger i återanvändbarhet av komponenter, dvs. att man kan använda en komponent på flera ställen. Övergången mellan hård- och programvara är otydlig, och inom inbyggda system är ordet komponent också mycket specifikt.

komponentintegrationstestning: Testning i syfte att hitta fel i gränssnitt och kommunikation mellan integrerade komponenter.

komponentspecifikation: En beskrivning av en komponents funktionalitet i termer av utvärden i förhållande till invärden vid specificerade villkor. Innefattar också en beskrivning av krävda icke-funktionella egenskaper (t.ex. resursanvändning).

komponenttestning: Testning av individuella komponenter. [Efter IEEE 610]

Kommentar: Komponenttestning är också en testnivå i en modell, d.v.s. det som testas först av utvecklare. Komponenttest är ibland en egen oberoende testfas, men vanligen sker komponenttest parallellt med utvecklingen av koden.

konfiguration: En sammansättning av komponenter eller system som definieras med numrering, karaktär och samhörighet med dess integrerande delar.

konfigurationshantering (KH; KL (Konfigurationsledning); CM (Configuration Management)): Är både en roll och ett arbete. Som roll kallar man det konfigurationsledning, vars arbete är att utföra konfigurationshantering. Det innebär beslut om uppgraderingar till nya versioner samt hantering av parallella versioner, dvs. att spårbarhet av data och program upprätthålls under modifiering. Disciplinen att använda teknisk och administrativ styrning och övervakning.

Kommentar: För test motsvaras detta av att man har kontroll över de ingående delarna, felrättningarna etc. i de olika testprogramsversionerna, inklusive testverktygsversioner och annan testdokumentation.

konfigurationshanteringsverktyg: Verktyg, dvs. system som underlättar identifiering och kontroll av konfigurationsobjekt deras status för ändringar och versioner, och leverans av fastställd konfiguration bestående av konfigurationsobjekt.

konfigurationsidentifiering: En del av konfigurationshanteringen som består av att välja ut de olika konfigurationsobjekten för ett system och registrera deras funktionella och fysiska karaktäristik i teknisk dokumentation. [IEEE 610]

konfigurationsobjekt: även konfigurationsprodukt. En grupp av maskinvara, programvara eller båda, som är utpekad för konfigurationshantering och hanteras som en enhet i konfigurationshanteringsprocessen. [IEEE 610]

konfigurationsrevision: även konfigurationsaudit. Att kontrollera innehållet i konfigurationsobjekt, t.ex. uppfyllelse av standard. [IEEE 610]

konfigurationsstyrning: Ett delmoment i konfigurationshanteringsprocessen bestående av utvärdering, koordinering, godkännande eller icke godkännande och införande av ändringar i konfigurationsobjekt efter det att formell konfigurationsidentifiering har införts. [IEEE 610]

konfigurationstestning: Se portabilitetstestning.

konkret testfall: Se testfall för lågnivåtester.

konsultativ testning: Testning som utförs med råd och vägledning från lämpliga experter utanför testgruppen (t.ex. tekniska experter och/eller områdesexperter).

kontinuerlig representation: En CMM (capability maturity model) struktur vars nivåer anger en rekommenderad ordning för arbetet med processförbättringar inom specificerat processområde.

kontrollant: Se granskare.

kontrolldiagram: Ett statistiskt processtyrningsverktyg som används för att övervaka en process och avgöra om det statistiskt sett är under kontroll. Det visar grafiskt medelvärde och de övre respektive undre kontrollgränserna (de högsta och lägsta värdena) i en process.

kontrollflöde: En abstrakt representation av alla möjliga sekvenser (vägar) i exekvering av en komponent eller ett system.

kontrollflödesanalys: En typ av statisk analys baserad på en representation av unika vägar (ordnade händelser) i en exekvering av en komponent eller system. Kontrollflödesanalys utvärderar integriteten hos kontrollflödesstrukturer och letar specifikt efter kontrollflödesanomalier såsom oändliga loopar och logiskt onåbara processteg.

kontrollflödesgraf: Ett diagram eller graf som representerar de möjliga olika kontrollflöden eller styrvägar som finns i programkoden.

kontrollflödestestning: Ett förhållningssätt till strukturbaserad testning där testfall är utformade för att utföra specifika sekvenser av händelser. Olika tekniker finns för kontrollflödestestning, t.ex. beslutstestning, villkorstestning, och vägtestning, som var och en har sitt specifika tillvägagångssätt och nivå på kontrollflödestestning. Se även beslutstestning, villkorstestning, vägtestning.

kontrollflödesväg: Se väg.

konvergensmått: Ett mätetal som visar framsteg mot ett definierat kriterium, t.ex. konvergens av det totala antalet utförda tester jämfört med det totala antalet planerade tester.

konverteringstestning: Testning av program eller procedurer som används för att konvertera data från ett befintligt system till ett annat nytt eller ersättande system.

kopplingstestning: Se komponentintegrationstestning.

kortslutning: En teknik för programmeringsspråk/programtolkar att utvärdera villkorssammansättningar där ett villkor på ena sidan av en logisk operator inte kan utvärderas om villkoret på andra sidan är tillräckligt för att fastställa det slutliga resultatet.

krav: Ett villkor eller en egenskap som en beställare kräver för att en lösning på ett problem eller ett mål för ett system eller delsystem ska uppfylla ett kontrakt, en standard, en specifikation eller andra formella tvingande dokument. [Efter IEEE 610]

kravbaserad test: Utveckling av testfall med utgångspunkt från krav på systemet eller komponenten.

kravfasen: Den tidsperiod i programvarans livscykel då kraven för programvaruprodukten definieras och dokumenteras. [IEEE 610]

kravhanteringsverktyg: Verktyg som stödjer registrering av krav, kravattribut och noteringar samt underlättar spårbarhet genom nivåer av krav och kravhantering och kravändringar. Vissa kravhanteringsverktyg har också stöd för statisk analys såsom kontroll av överensstämmande och överträdelser av fördefinierade krav eller regler.

kritisk framgångsfaktor: Nödvändiga komponenter för att en organisation eller ett företag med framgång ska kunna uppnå sina mål kallas för kritiska framgångsfaktorer. Se även innehållsbaserad modell.

kundanpassad programvara: Programvara utvecklad specifikt för vissa användare eller kunder. Motsatsen är standardprogramvara.

kvalifikation: Process för att påvisa kapacitet att specificerade krav kan uppfyllas. Notera att termen "kvalificerad" används för att ange status för kvalifikationen [ISO 9000]

kvalitet: Till vilken grad en komponent, ett system eller en process uppfyller ställda krav och/eller användares/kunds behov och förväntningar. [Efter IEEE 610]

kvalitetsegenskap: De viktigaste egenskaperna som påverkar en produkts kvalitet. [IEEE 610]

kvalitetskaraktäristik för programvara: Se kvalitetsegenskap.

kvalitetskaraktäristik: Se kvalitetsegenskap.

kvalitetskontroll: De operativa tekniker och aktiviteter som ingår som en del i kvalitetshantering och som fokuserar på att uppfylla kvalitetskrav. [efter ISO 8402]

kvalitetskontrollpunkt: En speciell milstolpe mellan två faser i ett projekt, där den efterföljande fasen starkt är beroende av kvaliteten på resultaten av den tidigare fasen. En kvalitetskontrollpunkt inkluderar en formell kontroll av dokumenten från den tidigare fasen.

kvalitetskostnad: Den totala kostnaden som resultat av kvalitetsarbete och kvalitetsproblem ofta uppdelat i förbyggande, analys-, interna felsymptoms- och externa felsymptomskostnader.

Kvalitetsrisk: En risk kopplad till kvalitetsattribut. Se också kvalitet.

kvalitetsstyrning: Samordnande aktiviteter för att styra och kontrollera en organisation beträffande kvalitet. Styrning och kontroll beträffande kvalitet generellt, omfattar upprättandet av policy och kvalitetsmålsättningar, kvalitetsplanering, kvalitetskontroll, kvalitetssäkring och kvalitetsförbättringar. [ISO 9000]

kvalitetssäkring: även kvalitetsförsäkrans. Den del av kvalitetshanteringen som fokuserar på att tillgodose den konfidens som krävs för att ställda kvalitetskrav kan uppfyllas. [ISO 9000]

källkodsats: Se kodsats.

känslobaserad kvalitet: Ett sätt att se på kvalitet som innebär att kvalitet inte kan definieras på ett exakt sätt, trots det kan vi ändå avgöra om kvaliteten hos en produkt när vi ser den. I detta synsätt beror kvaliteten på uppfattning och de känslor som produkten väcker hos en individ eller en grupp av individer. [Efter Garvin] Se även användarbaserad kvalitet, produktbaserad kvalitet, tillverkningsbaserad kvalitet och värdebaserad kvalitet.

L

lagring: Se resursanvändning.

lagringstestning: Se resursanvändningstestning.

lastprofil: En specifikation över de aktiviteter som kan inträffa vid användning och som ska utföras vid test av en komponent eller ett system. En lastprofil innehåller ett antal virtuella användare som utför en mängd transaktioner inom en specificerad tidsperiod och enligt en fördefinierad driftsprofil. Se även driftsprofil.

lasttestning: En typ av prestandatestning som utvärderar komponentens eller systemets uppträdande med växande last, t.ex. antalet parallella användare och/eller antalet transaktioner, för att se vilken last som kan hanteras av komponenten eller systemet. Se även prestandatestning, stresstestning.

lasttestverktyg: Ett verktyg som stödjer lasttestning och kan simulera en växande last, t.ex. antalet parallella användare och/eller antalet transaktioner inom en specificerad tidsperiod. Se även prestandatestverktyg.

LCSAJ (Linear Code sequence and Jump), sekventiell kod med hopp: En kodradsnumrering i en källkodslistning eller i kodblocket, som består av tre delar: starten av kodblocket eller listningen som beskriver sekvensen av exekverbara kodrader, slutet av kodblocket i listan, samt den adress till vilket kontrollflödet överförs i slutet av kodsekvensen.

LCSAJ-testning: En strukturell testdesignteknik där testfall är konstruerade med avsikt att uppfylla LCSAJ-täckning. Se även LCSAJ-täckning.

LCSAJ-täckning, täckning av sekventiell kod med hopp: Hur många procent av LCSAJ som har blivit exekverat av en testsvit. 100% LCSAJ innebär också 100% grenvillkorstäckning.

leverabel: En produkt som formellt skall överlämnas till en utpekad mottagare.

leveransdokument: Ett dokument som identifierar testenheter, konfiguration, nuvarande status och annan information levererad till test från utvecklare i början av en testfas. [IEEE 829]

leveransrapport för element: Se leveransdokument.

leveransrapport för testelement: Se leveransdokument.

livscykel för programvara: Den tidsperiod som börjar när en idé för programvaran föds och slutar när programvaran inte längre finns tillgänglig för användning.

Kommentar: En programvaras livscykel innehåller vanligtvis en idéfas, kravfas, designfas, implementeringsfas, testfas, installations och utcheckningsfas, drift- och underhållfas och ibland en indragning av programvaran. Notera att dessa faser kan överlappa eller utföras iterativt.

livscykelmodell: En uppdelning i faser av livstiden hos en produkt eller ett projekt.[CMMI]. Se även livscykel för programvara.

logikdriven testning: Se white-box-testning.

logiktestfall: Se högnivåtestfall.

logiktäckande testning: Se white-box-testning. [Myers]

lågnivåtestfall: Ett testfall med konkreta (implementationsnivå) värden för indata och förväntat resultat. Logiska operatörer från högnivåtestfall ersätts med rätta värden som motsvarar syftet med de logiska operatörerna. Se även högnivåtestfall.

lärande (IDEAL): Den fas inom IDEAL modellen i vilken man lär av erfarenheter och förbättrar sina möjligheter att anpassa sig till nya processer och teknologier i framtiden. Lärande-fasen består av aktiviteterna: analysera och validera samt formulera framtida aktiviteter. Se även IDEAL.

M

manifestet för testprocessförbättring: Ett uttryck inspirerat av det agila manifestet, vilket beskriver de värden som ligger till grund för förbättringar av testprocessen. Dessa värden är:

- flexibilitet framför detaljerade processer,
- best practice framför mallar,
- produktorientering framför processororientering,
- granskning framför kvalitetssäkring(savdelningar),
- affärsdriven framför modelldriven. [Veenendaal08]

MCDC: Se Modifierad testning av villkorsbeslut.

Mean Time Between Failures: Genomsnittstid mellan två på varandra följande felsymptom. MTBF används typiskt i en Reliability Growth Model där man antar att avvikelser omedelbart repareras, som del av felrättningsprocessen. Se även Reliability Growth Model.

Mean Time To Repair: Genomsnittstid det tar från en felyttring tills dess att motsvarande defekt är åtgärdad, inklusive tiden det tar att testa att defekten har åtgärdats.

medberoende beteende: Överdrivet känslomässigt eller psykologiskt beroende av en annan person, specifikt i situationen när man försöker förändra en persons [oönskade] beteende samtidigt som man ändå uppmuntrar det befintliga beteendet. T.ex., när man klagar på sena leveranser av programvara till testavdelningen samtidigt som man njuter av att spela "hjärte" genom att arbeta övertid för att ta igen den förlorade tiden, och på så sätt berättigar förseningen.

medvetet vägval: Val av en mängd indata med syfte att tvinga (forcera) exekveringen att följa en viss väg.

metod för att skriva testfall genom orsak-verkangrafer: En metod eller testteknik där testfall skrivs genom att använda grafer som talar om orsak och verkan. [BS 7925/2]

metodisk testning: Testning baserad på en standarduppsättning av tester, t.ex. en checklista, en kvalitetsnorm, eller en uppsättning generella testfall.

migrationstestning: även översättningstestning. Se konverteringstestning.

milstolpe: Ett delmål i ett projekt, som oftast ska vara uppfyllt innan man går vidare i projektet.

mind-map: Ett diagram som används för att representera begrepp, idéer, aktiviteter eller andra saker, som är kopplade till eller arrangerade kring ett nyckelord eller en central idé. Mind-maps används för att generera, visualisera, strukturera och klassificera idéer samt som en hjälp vid instudering, organisering, problemlösning, beslutsfattande och vid skrivande.

minnesläcka: En minnesåtkomststavvikelse som orsakas av en defekt i ett programs dynamiska minneshantering vilken innebär att en process misslyckas med att frigöra minne efter användning. Kan leda till att programmet och/eller andra samtidigt exekverande processer havererar på grund av minnesbrist.

missat fel: Ett fel som inte upptäcktes på en tidigare testnivå där sådana fel förväntas hittas. Se även Defect Detection Percentage (DDP).

misstag: (1) En mänsklig åtgärd som producerar ett felaktigt resultat. [Efter IEEE 610] Se även mänskligt misstag.

(2) även felaktighet. Se fel.

MITM-attack: Även janusattack. Avlyssning, härmning och/eller ändring och efterföljande förmedling av kommunikation (t.ex. kreditkortstransaktioner) utförd av en tredje part så att en användare är omedveten om den tredje parten närvaro.

modell för inkrementell utveckling: En utvecklingsmodell där projekt delas upp i en serie av inkrement, där varje inkrement levererar en del av funktionaliteten som ingår i projektkraven. Kraven prioriteras och levereras i prioriteringsordning i det tilldelade inkrementet. I vissa, men inte alla, versioner av projektets livscykel följer man en "mini-V-modell", med egna faser för design, kodning och testning.

modell för iterativ utveckling: En process för utveckling där ett projekt är indelat i ett antal, vanligtvis många, iterationer. En iteration är ett fullständigt utvecklingsvarv som resulterar i en utgåva (intern eller extern) av en exekverbar produkt. En delmängd av den slutliga produkten växer fram för varje iteration för att till sist utgöra den slutliga produkten.

Modellbaserad testning: Testning baserad på en modell av komponenten eller systemet som testas, t.ex. reliability growth model, användningsmodeller som driftsprofiler eller beteendemodeller som beslutstabell eller tillståndsbaserad testning.

modellerad driftsmiljötest: Statistisk testning med hjälp av en modell av systemets funktion baserad på en sannolik användning av systemet. [Musa]

modelleringsverktyg: Ett verktyg som stödjer skapande, förändring och verifiering av modeller för program eller system. [Graham]

moderator: Den person som är ledare och huvudansvarig för en inspektion eller annan granskningsprocess.

modifierad testning av villkorsbeslut: En testteknik där testfall är avsedda att exekvera villkor som påverkar kodgrensvalet i programkod. Varje villkorsresultat påverkar valet av kodgren.

modifierad testning av villkorskombinationer: Se modifierad testning av villkorsbeslut

modifierad täckning av villkorsbeslut: Procent av alla villkorsbeslut, som oberoende påverkar kodgrenval exekverat av testfallssviten. 100 % MCDC täckning medför 100 % bågtäckning.

modifierad täckning av villkorskombinationer: Se modifierad täckning av villkorsbeslut

modul: Se komponent.

modultestning: Se komponenttestning.

mognad: (1) Förmågan hos en organisation med avseende på hur verksamma och effektiva dess processer och arbetssätt är. Se även Capability Maturity Model Integration, Test Maturity Model Integration.

(2) Kapacitet hos en programvaruprodukt att undvika negativa effekter som följd av fel i programvara. [ISO 9126] Se även tillförlitlighet.

mognadsmodell: En strukturerad samling aspekter som beskriver en organisations mognad som även kan användas för att definiera och förstå en organisations processer. En mognadsmodell tillhandahåller ofta ett gemensamt språk, en delad vision och ett ramverk för att prioritera förbättringsaktiviteter.

mognadsnivå: Uppnådd processförbättringsgrad över en fördefinierad mängd processområden, i vilka alla mognadsmål för alla processområden måste vara uppnådda. [TMMi]

MTBF: Se Mean Time Between Failures.

MTTR: Se Mean Time To Repair.

multipla villkor: Två eller fler separata villkor som är sammankopplade med en logisk operator (AND, OR eller XOR), t.ex. "A>B AND C>1000".

mutationsanalys: Metod som används när man vill avgöra fullständigheten hos en testfallssvit genom att mäta i vilken grad den kan särskilja olika varianter (mutanter) av programmet. Se även felplantering.

Kommentar: Målet med mutationstestning är att skaffa en komplett testsvit som "fullständigt" testar en aspekt av programmet/systemet. Man jämför två varianter av systemet, ett korrekt och en variation av programmet/systemet, sida vid sida. Om testsviten inte hittar någon skillnad på den "rätta" varianten och den som har varierats "felaktigt" så har man en mutant. Därefter skapas ett nytt testfall, som inte godkänner mutanten, och adderas till testsviten. Denna procedur att hitta mutanter bör göras automatiskt och sker vanligen på kodnivå. Att hitta testfall kan ibland vara svårt att göra automatiskt.

mutationstestning: även komparativ testning. Se sida-vid-sida testning.

Myers-Briggs Type Indicator (MBTI): Ett mått på psykologiska preferenser som representerar olika typer av människors personligheter och sätt att kommunicera.

mått: Ett mätvärde (numeriskt värde) eller kategori som man tilldelar en egenskap hos en enhet genom att utföra mätning. [ISO14598]

mänskligt misstag: Se misstag.

mätetal: Ett värde som anger hur många gånger den valda enheten ingår i ett mätvärde eller en storhet. [ISO14598]

mätning: Processen (utförandet) att tilldela ett numeriskt värde eller en kategori till en enhet för att beskriva en egenskap hos enheten. [ISO14598]

mätningsskala: En skala som bestämmer vilka begränsningar som gäller för analys av data. [ISO14598]

möjlig väg: En väg för vilken det finns invärden och förutsättningar som möjliggör att vägen exekveras.

N

nedlusning: Se felinjektion. [Abbott]

negativ testning: Tester med syfte att påvisa att en komponent eller ett system inte fungerar. Negativ testning är mer relaterad till testares attityd än till ett specifikt angreppssätt eller testdesigntechnik, t.ex. testning med ogiltiga indata värden eller testning med undantag (eng. exceptions). [Efter Beizer]

nerifrån och upp-testning: även bottom-up-testning. En inkrementell integrationstestteknik, där komponenterna på den lägsta nivån testas först, därefter sker en successiv integration och testning av systemet i nivåer. Processen upprepas tills alla komponenter har integrerats till ett system på högsta nivån. Se även integrationstestning, uppifrån och ner-testning.

nivåtestplan: En testplan som vanligtvis berör en enskild testnivå. Se även testplan.

noggrannhetstestning: Den testaktivitet som ska säkerställa noggrannheten hos en programvaruprodukt.
Se även exakthet.

N-switch-testning: En variant på test av tillståndsovergångar där varje testfall är designad att utöva alla giltiga sekvenser av N+1 övergångar. [Chow] Se även tillståndsbaserad testning.

N-switch-täckning: Den procentuella täckningen av N+1 tillståndsovergångar som har exekverats genom ett eller flera testfall ur en testfallssvit. [Chow]

n-wise-testning: En black-box-testdesignteknik där testfall designas för att exekvera alla diskreta kombinationer av en uppsättning med n ingångsparametrar. Se även testning baserad på ortogonalitetsmatris, parvis testning.

nyckelordsdriven testning: En testteknik där man väljer testfall baserat på vissa nyckelord, ofta affärsdrivet och kopplat till den funktionella och logiska användningen av systemet. Se även datadriven testning.

nyckelprestandatal: Se prestandatal.

O

oberoende testning: Separation av ansvar, vilket säkerställer att man uppfyller kravet på en objektiv testning. [Efter DO-178b]

Kommentar: I detta (test)sammanhang syftas oftast på revision av ett system, eller på testfall, eller testorganisationens uppdelning i nivåer.

odelbart villkor: Ett tillstånd som inte kan brytas ned, dvs. ett tillstånd som inte innehåller två eller flera enkla villkor förenade med en logisk operator (AND, OR, XOR).

ogiltighetstestning: Testning med invärden som inte bör tolereras av komponenten eller systemet. Se även feltolerans, negativ testning.

omdömetest: Se röktest.

omtestning av felrättningar: Se omtestning.

omtestning: Testning som innebär omexekvering av de testfall som tidigare misslyckats för att man på så sätt ska verifiera att ändringsåtgärder har lyckats. Jmf. regressionstestning.

Kommentar: Omexekvering med ett testfall som tidigare felat, för att avgöra om rättning verkligen är korrigerad, dvs. testning att en felrättning verkligen har tagit bort den kända avvikelsen. Regressionstestning kontrollerar dessutom att rättningen inte har medfört andra och oönskade sideeffekter.

onåbar kod: Kod som inte kan nås och därför är omöjlig att exekvera.

onåbar väg: En exekveringsväg som det är omöjligt att nå, oberoende av vilken kombination av invärden som väljs.

open source-verktyg: Ett verktyg som är tillgänglig för alla potentiella användare i form av källkod, oftast via internet; dess användare har tillåtelse, oftast på licens, att studera, förändra, förbättra och, ibland, för att distribuera programvaran.

operational profiling: TBD

operationell profil: Även driftsprofil. Representationen av en mängd uppgifter utförda av en komponent eller system som kan ha baserats på användarnas beteende vid interaktion med komponenten eller systemet och sannolikheten för att dessa inträffar. En uppgift är snarare logisk än fysisk och kan utövas med flera datorer eller utövas i diskret tidsrymd.

orakel: Se testorakel.

orsaksanalys: Analys av defekter för att identifiera grundorsaken till att de uppstått. [CMMI]

orsak-verkandigram: En grafisk representation för att organisera och visualisera sambanden mellan olika potentiella grundorsaker till ett problem. Möjliga grundorsaker till en verkligt eller potentiell defekt eller avvikelse organiseras i kategorier och underkategorier i en horisontell trädstruktur, med [den potentiella] defekten eller avvikelsen som rot. [efter Juran]

orsak-verkangraf: En orsak-verkangraf är en grafisk representation av indata och/eller stimuli (orsaker) med deras associerade utdata (effekter). Denna graf eller relation (ofta arrangerad i en beslutstabel) kan användas till att skapa testfall.

orthogonal matris: En 2-dimensionell matris konstruerad enligt matematiska egenskaper, så att två godtyckliga kolumner i matrisen ger alla kombinationer av par bestående av varje siffra i matrisen.

outsourcad testning: Testning utförs av personer som varken är anställda kollegor eller samlokaliserade med projektgruppen.

P

Pareto-analys: En teknik för beslutsfattande, som bygger på statistiska metoder, vilken används för att identifiera en begränsad mängd faktorer som har en stor påverkan på det slutliga resultatet. En stor majoritet av problem (80%) uppstår på grund av ett mindre antal (20%) nyckelorsaker.

parprogrammering: En metod för programvaruutveckling där varje kodrad (i produktion och/eller test) i en komponent är skriven av två programmerare vid en gemensam dator. Detta för att implicit utföra kodgranskning samtidigt som kod skrivs.

partestning: Två testare som arbetar tillsammans för att hitta brister i programvaran. Vanligtvis delar dessa på en dator och turas om att ha kontroll över datorn vid testningen.

parvis integrationstestning: En sorts integrationstester som riktar sig mot komponentpar som samverkar, enligt en anropsgraf.

parvis testning: En black-box-testdesigntechnik i vilken testfall är designade att exekvera alla möjliga diskreta kombinationer av varje par av indata. Se även n-wise testning, testning baserad på ortogonalitetsmatris.

pekare: Ett dataelement som specificerar platsen för ett annat dataelement, t.ex. ett dataelement som specificerar adressen till nästa post som ska processas. [IEEE 610]

planning poker: En samsynsbaserad estimeringsteknik, som i huvudsak används för att uppskatta arbetsinsatser eller den relativa storleken på användarberättelser vid agil programvaruutveckling. Det är en variant av den Wide Band Delphi-metoden som använder en kortlek vars värden representerar de enheter som teamet estimerar. Se även agil programvaruutveckling, Wide Band Delphi.

portabilitet: även flyttbarhet. Anger hur lätt en programvaruprodukt kan flyttas från en hårdvaru- eller programvarumiljö till en annan. [ISO 9126]

portabilitetstestning: Testning av portabiliteten/flyttbarheten hos en programvaruprodukt från en hårdvaru- eller programvarumiljö till en annan.

prestanda: Förmågan hos ett system eller en komponent att uppfylla ställda krav på funktionalitet samtidigt som satta begränsningar för beräkningstider och systemkapacitet hålls. [Efter IEEE 610] Se även effektivitet.

prestandaprofilering: Uppgiften att analysera, t.ex. identifiera flaskhalsar baserat på genererade mätetal, och trimma prestandan i en programvarukomponent eller ett system med hjälp av verktyg.

prestandatal: En abstrakt måttstock för tidseffektivitet och resurseffektivitet som används som vägledning och kontroll i utveckling för förbättring av programvara, t.ex. ledtidsglidning vid mjukvaruutveckling. [CMMI]

prestandatestning: Testning vars syfte är att utvärdera prestanda hos ett system, en systemdel eller en komponent. Testningen avser att demonstrera att testobjektet uppfyller specifika prestandakrav. Prestandatester omfattar både last- och stresstester. Se även effektivitetstestning.

prestandatestverktyg: Ett verktyg som stödjer prestandatestning som vanligtvis har två huvudfaciliteter, lastgenerering och mätning av testtransaktioner. Lastgenerering kan simulera antingen flera användare eller stora volymer av indata. Svarstider kan mätas och loggas från utvalda transaktioner under testexekvering. Dessa verktyg kan normalt tillhandahålla rapporter som baseras på testloggar och diagram som visar last jämfört med svarstider.

prioritet: Graden av betydelse för verksamheten som kan tilldelas en produkt (sak), t.ex. fel.

PRISMA: Ett systematiskt angreppssätt för riskbaserad testning som använder identifiering och analys av produktrisker för att skapa en produktriskmatris som tar ställning till sannolikhet och konsekvens.

probeffekt: Ett mätinstruments inverkan på komponenten eller systemet när komponenten eller systemet mäts av t.ex. ett prestandatestverktyg eller en övervakare. Prestanda kan till exempel bli sämre när ett prestandatestverktyg används.

problem: Se fel.

problemadministration: även defektadministration, bristadministration. Se felhantering.

problemrapport: även defektrapport, bristrapport. Se felrapport.

procedurtestning: Testning fokuserad på att säkerställa att en komponent eller ett system kan användas tillsammans med användares nya eller existerande affärsprocedurer eller driftsprocedurer.

process: En mängd samverkande aktiviteter som transformerar indata till utdata. [ISO 12207]

Processanpassad testning: Testning som följer en uppsättning definierade processer, t.ex. definierade av en extern part såsom en standardiseringskommitté. Se även standardanpassad testning.

processcykeltest: En black-box-teknik där testfallen är designade för att exekvera affärsprocedurer och processer. [TMap] Se även procedurtestning.

processförbättring: En plan med aktiviteter designade för att förbättra effektiviteten och mognaden i en organisations processer, avser även resultatet av en sådan plan. [CMMI]

processmodell: Ett ramverk, i vilket processer av samma typ klassificeras, t.ex. en testförbättringsmodell.

processreferensmodell: En processmodell som tillhandahåller en generisk samling av best practice och förbättringsförslag i form av en steg för steg-metod.

processutvärdering: En disciplinerad utvärdering av en organisations programvaruutvecklings-processer mot en referensmodell [efter ISO 15504].

Product RiSk MAnagement: Se PRISMA.

produktbaserad kvalitet: Ett sätt att se på kvalitet som innebär att kvaliteten hos en produkt definieras via ett antal väldefinierade kvalitetsattribut, vilka ska kunna mätas objektivt och kvantitativt. Kvalitetsskillnader mellan produkter av samma typ kan spåras tillbaka till hur specifika kvalitetsattribut har implementerats. [Efter Garvin] Se även användarbaserad kvalitet, tillverkningsbaserad kvalitet, upplevd kvalitet och värdebaserad kvalitet.

produktionsacceptanstestning: Se driftsacceptanstestning.

produktkaraktäristik för programvara: Se kvalitetsegenskap.

produktrisk: En risk direkt relaterad till testobjektet, uttryckt i sannolikhet och påverkan, för att en produkt inte uppfyller ställda krav. Se även risk.

programfiness: Se finess.

programinstrument: Se instrument.

programtestning: Se komponenttestning.

programvara: Ett antal styrinstruktioner, d.v.s. kodsatser som fungerar som en helhet som tillsammans med data och maskinvara utgör ett system. [IEEE 610]

programvaruattack: Se attack.

programvarukvalitet: Den totala funktionaliteten och egenskaperna hos en programvaruprodukt med syfte att tillfredsställa fastställda och underförstådda krav. [Efter ISO 9126] Se även kvalitet.

projekt: Ett projekt är en unik mängd av samordnade och kontrollerade aktiviteter där start- och sluttid för aktiviteterna är underordnade och samstämmiga med speciella krav, såsom krav på tid, kostnad och resurser. [Efter ISO9000]

projektretrospektiv: Ett strukturerat sätt att ta tillvara erfarenheter och att skapa specifika handlingsplaner för att förbättra nästa projekt eller projektfas.

projektrisk: En risk som är relaterad till ledning och kontroll av (test) projektet, t.ex., brist på personal, hårda deadlines, ändrade krav, etc. Se även risk.

projekttestplan: Se överordnad testplan.

projektutvärderingsmöte: Se retrospektivmöte.

påverkansanalys: Bedömningen av hur omfattande ändringar av utvecklingsdokumentation, testdokumentation och komponenter som behövs, för att genomföra en viss förändring med specificerade krav.

Q

QFD: Quality Function Deployment. Se fastställande av funktionskvalitet

R

Rational Unified Process: Ett patenterat tillämpbart iterativt utvecklingsprocessramverk som består av fyra projektlivscyklifaser: förberedelse, etablering, konstruktion och överlämning.

Reaktiv testning: Testning som dynamiskt svarar på det aktuella systemet som testas och de testresultat som erhållits. Typisk reaktiv testning har en reducerad planeringscykel och testfaserna design och implementation utförs inte förrän testobjektet erhållits.

registrator: Se sekreterare.

registrerings- och uppspelningsverktyg: Se in- och uppspelningsverktyg.

regleringstestning: Se testning för typgodkännande.

regressionshämmande testning: Testning med olika tekniker för att hantera riskerna med regression, t.ex. genom att designa återanvändbar testvara och genom omfattande testautomatisering på en eller flera testnivåer.

regressionstestning: Testning av ett program där testfallen exekveras i syfte att säkerställa att fel inte blivit introducerade eller blottlagda som ett resultat av utförda ändringar eller tillägg i programmet. Regressionstestning utförs när programvaran eller dess miljö har ändrats. Jämför omtestning.

Reliability Growth Model: En modell visande den ökning i tillförlitlighet under en tidsperiod där kontinuerlig testning av komponenter eller system och rättning av fel som visar sig i tillförlitlighetsförsymptom.

resultat: Konsekvensen eller utfallet från exekveringen av ett test. Se även faktiskt resultat (eng. actual result), förväntat resultat (eng. expected result).

resursanvändning: Egenskapen hos programvaruprodukten att använda bestämt antal och bestämd typ av resurser, t.ex. det primär- och sekundärminne som används av programmet och storleken på temporära filer, när programvaran utför sin funktion under bestämda förutsättningar. [ISO 9126] Se även effektivitet.

resursanvändningstestning: Testprocessen för att säkerställa resursanvändningen för en programvaruprodukt. Se även effektivitetstestning.

retrospektivmöte: Ett möte i slutet av ett projekt där projektets medlemmar utvärderar projektet och lärdom noteras för att användas i kommande projekt.

revision: även audit.

Oberoende granskning med syfte att kontrollera att bestämda aktiviteter har utförts på rätt sätt enligt givna instruktioner, standarder, specifikationer eller procedurer baserat på objektiva kriterier som specificerar:

- (1) format och innehåll hos den producerade produkten
- (2) processen som styr produktionen av produkten
- (3) hur standarder och riktlinjer skall tillmötesgå. [IEEE 1028]

revisionsspår: även revisionsväg. En uppföljningsväg i en process där man kan spåra tillbaka till ursprungsindata genom hela processen. Detta underlättar felanalys och processrevision. [Efter TMap]

risk: En faktor som kan resultera i en framtida negativ konsekvens oftast uttryckt i sannolikhet och påverkan.

riskanalys: En process för att analysera identifierade projekt- och produktrisker för att fastställa sannolikheten för att de inträffar samt den påverkan som fås om så sker.

riskbaserad testning: Ett angreppssätt för testning som reducerar nivån för produktrisker och informerar intressenter om riskernas status, med början i initiala faserna i projektet. Det innefattar identifiering av produktrisker samt användning av risknivåer för att styra test processen.

Kommentar: Att utföra riskkontroll genom testning möjliggör återkoppling som ger information om kvarvarande risker genom att mäta effektiviteten i borttagandet av allvarliga fel och effektiviteten i katastrofplaner.

riskbedömning: Processen att identifiera och därefter bedöma en given projekt- eller produktrisk för att avgöra dess risknivå, typiskt genom att tilldela grader av sannolikhet och konsekvens. Se även produktrisk, projektrisk, risk, riskkonsekvens, risknivå, risksannolikhet.

riskhantering: Ett systematiskt tillvägagångssätt i syfte att identifiera, analysera, rangordna, definiera åtgärder för att åtgärda risker, hänskjuta risker, minimera risker eller avfärda risker.

riskidentifiering: Processen för att identifiera risker genom att använda tekniker som idékläckning, checklistor och felhistorik.

riskkategori: Se risktyp.

riskkonsekvens: Skadan som kommer att uppstå ifall risken blir ett verkligt resultat eller händelse.

riskkontroll: En process där beslut tas och skyddande åtgärder implementeras för att reducera riskerna till (eller behålla riskerna på) en viss nivå.

riskminskning: Se riskkontroll.

risknivå: Viktigheten för en risk definierad genom egenskaperna påverkan och sannolikhet hos risken. Risknivån kan användas för att avgöra hur mycket testning som ska utföras. En risknivå kan uttryckas kvalitativt (t.ex. hög, medium, låg) eller kvantitativt.

risksannolikhet: Den uppskattade sannolikheten att en risk kommer att bli ett verkligt resultat eller händelse.

risktyp: En uppsättning risker grupperade efter en eller flera gemensamma faktorer såsom kvalitetsattribut, orsak, lokalisering eller potentiella konsekvenser av riskerna. En speciell kategori av produktrisktyper är relaterade till den typ av testning som kan minska (kontrollera) dessa risker. Till exempel kan risken att användarinteraktion missförstås minskas genom användbarhetstestning.

robusthet: Den nivå på vilken en komponent eller ett system kan fungera rätt med ogiltiga indata eller stressad miljö. [IEEE 610] Se även feltolerans.

robusthetstestning: Testning för att fastställa robustheten hos en programvaruprodukt.

RUP: Se Rational Unified Process.

röktest: Ett urval av alla definierade/planerade testfall som täcker den huvudsakliga funktionaliteten hos en komponent eller ett system, som räcker för att förvissa sig om att de mest kritiska funktionerna av programmet fungerar utan att fördjupa sig i detaljer. Inom industrin är dagligt byggande och röktest "best practice". Se även öppningstest.

S

S.M.A.R.T. metodik för mål: En metod där målen definieras mycket specifikt snarare än generellt. SMART är en akronym som härleds från attributen för det mål som skall definieras: Specifikt, Mätbart, Accepterat, Realistiskt, Tidsatt.

samexisterande: Förmågan hos en programvara att samexistera med annan oberoende programvara i en gemensam miljö där dessa delar gemensamma resurser. [ISO 9126] Se även portabilitet.

scenariobaserad test: Se användningsfallsbaserad testning.

SCRUM: Ett iterativt, inkrementellt ramverk för hantering av projekt, vanligen använt i agila programvaruutvecklingsmetoder. Se även agil programvaruutveckling.

sekreterare: En person som ansvarar för att i ett loggningsformulär dokumentera varje defekt som påtalats och varje förslag till processförbättringar under ett granskningsmöte. Sekreteraren måste säkerställa att loggningsformuläret är läsbart och går att förstå.

servicebarhetstestning: Se underhållbarhetstestning.

sessionsbaserad testning: En testmetodik i vilken testaktiviteter planeras i sessioner som består av samtidig testdesign och testexekvering och där sessionerna inte får avbrytas. Sessionsbaserad testning används ofta vid utforskande testning.

sessionsbaserad teststyrning: En metod för att mäta och styra sessionsbaserad testning, t.ex. utforskande testning.

SFMEA (Software Failure Mode and Effect Analysis): Se FMEA.

SFMECA (Software Failure Mode, Effects, and Criticality Analysis): Se FMECA.

SFTA (Software Fault Tree Analysis): Se FTA (Fault Tree Analysis).

Shewhart-diagram: Se kontrolldiagram.

sida-vid-sida testning: Testning där två eller flera varianter av komponenter eller system exekveras med samma invärden, varefter man jämför utvärden och analyserar skillnaderna. [IEEE 610]

simulator: Utrustning, program eller system som används vid testning och beter sig likadant som ett givet system eller en komponent, då det ges en mängd av kontrollerade invärden. [Efter IEEE 610, DO178b] Se även emulator.

Kommentar: Notera att egenskaper som tidsberoende faktorer oftast inte går att testa med denna metod.

simulering: Återgivning av valda beteendemönster och egenskaper hos ett fysiskt eller abstrakt system med hjälp av ett annat system.

skalbar testning: Test som utförs för att fastställa skalbarhet hos programvaran.

skalbarhet: Kapaciteten hos en programvara att bli uppgraderad till en anpassad ökad belastning.

skriptspråk: Ett programmeringsspråk där exekverbara testskript skrivs. Ofta används ett testexekveringsverktyg (t.ex. in- och uppspelningsverktyg).

skripttestning: Testexekvering utförs genom att följa dokumenterade testsekvenser.

skrivbordstest: Testning av programvara eller specifikation genom manuell simulering av en exekvering. Se även statisk testning.

slumpbaserade tester: En black-box-teknik där testfallen är valda, möjligen med användning av pseudoslumpmässiga genereringsalgoritmer, för att generera möjliga operationella scenarion. Denna teknik kan användas för icke funktionella attribut som t.ex. tillförlitlighet och prestanda.

slutkriterier: Se avslutskriterier.

slutlig testrapport: Ett dokument som summerar testaktiviteter och resultat. Det innehåller också en utvärdering av motsvarande testelement jämfört med avslutskriterier. [Efter IEEE 829]

sluttillstånd: Tillstånd hos miljö och testobjekt som måste vara uppfyllda efter exekvering av ett test eller en testprocedur.

Software Process Improvement: Ett aktivitetsprogram framtaget för att förbättra prestandan och mognaden i en organisations programvaruprocesser samt resultaten av att ett sådant program använts. [Efter CMMI]

specificerade indata: Invärden till vilka en specifikation beskriver förväntat utvärde.

specifikation: Ett dokument som specificerar (idealiskt på ett komplett, precist och verifierbart sätt) krav, design, beteende och egenskaper hos en komponent eller ett system och. Ofta ingår procedurer som används för att avgöra om dessa mål har blivit uppfyllda. [Efter IEEE 610]

specifikationsbaserad teknik: Se black-box-testdesigntechnik.

specifikationsbaserad testdesigntechnik: Se black-box-testdesigntechnik.

specifikationsbaserad testning: Se black-box-testning.

SPI: Se Software Process Improvement.

spårbarhet: Möjligheten att identifiera relaterade element i dokumentation och programvara, som t.ex. systemkrav, med tillhörande tester. Se även horisontell spårbarhet, vertikal spårbarhet.

spårbarhetsmatris: En tvådimensionell tabell vilken visar sambandet mellan två entiteter (t.ex. krav och testfall). Tabellen används för att fastställa och uppnå täckning, för att spåra framåt och bakåt från en entitet till en annan, och för att bedöma påverkan av föreslagna ändringar.

stabilitet: Kapaciteten hos en programvaruprodukt att inte påverkas negativt av modifieringar i programvaran. [ISO 9126] Se även underhållbarhet.

standard: En mängd formella, tvingande alternativt rådgörande, krav framtagna för att säkerställa ett konsekvent arbetssätt. (t.ex. ISO/IEC standarder, IEEE standarder och organisatoriska standarder). [Efter CMMI]

standardanpassad testning: Testning som uppfyller en uppsättning krav som fastställts av en standard, t.ex. en branschspecifik standard för test eller en standard rörande testning av säkerhetskritiska system. Se även processanpassad testning.

standardprogramvara: En programvaruprodukt som är utvecklad för den allmänna marknaden, dvs. för ett stort antal kunder, och som levereras till många kunder i samma format. Se även COTS.

standardtestning: Se testning för typgodkännande.

startkriterier: Den mängd av generiska och specifika villkor som tillåter en process att fortsätta med ett definierat arbete, t.ex. en testfas. Syftet med startkriterier är att förhindra start av ett arbete som annars skulle medföra mer (onödig) insats jämfört med insatsen att eliminera de felaktiga startkriterierna. [Gilb and Graham]

statisk analys: Analys av programvaruarterfakter, t.ex. krav eller programvarukod, utan att dessa artefakter exekveras. Vanligtvis utförs statisk analys med hjälp av verktyg.

statisk kodanalys: Analys av källkod utförd utan att programvaran exekveras.

statisk testning: Testning av en programvaruarterfakt, t.ex. krav, design, eller kod utan att exekvera dessa artefakter, t.ex. granskningar eller statisk analys.

statiskt analysverktyg: Se verktyg för statisk analys.

statistikbaserad test: En testdesigntechnik där man använder sig av statistisk spridning av invärden för att konstruera representativa testfall. Se även modellerad driftsmiljötest.

statusredovisning: Ett moment i konfigurationshantering som består av att dokumentera och rapportera information som är nödvändig för att hantera en konfiguration på ett effektivt sätt. Informationen innehåller en lista över godkända identifierade konfigurationer, status på föreslagna ändringar av konfigurationen och status på implementation av godkända ändringar. [IEEE 610]

stegrepresentation: En modellstruktur man genom att uppnå målen inom ett antal processområden etablerar en mognadsnivå, varje nivå bygger upp en bas för de efterföljande nivåerna. [CMMI]

STEP: Se Systematic Test and Evaluation Process.

stresstestning: En typ av prestandatest som utförs för att utvärdera ett system eller en komponent på eller över gränsen för dess förväntade eller specificerade arbetsbelastning, eller med reducerad tillgänglighet till minne eller servrar. [Efter IEEE 610] Se även prestandatest, lasttest.

strukturbaserad teknik: Se white-box-testdesignteknik.

strukturbaserad testdesignteknik: Se white-box-testdesignteknik.

strukturbaserad testning: Se white-box-testning.

strukturell testning: Se white-box-testning.

strukturerad granskningsgenomgång: Se genomgång.

strukturell täckning: även strukturell kodtäckning. Mätning av kodtäckning som baseras på den interna strukturen hos koden eller systemet.

stubbe: En speciellt utvecklad programkod, eller programskelett, vars syfte är underlätta testning av en programdel eller komponent, som är beroende av eller anropar stubben.

Kommentar: Används ofta för att ersätta en modul som ännu inte utvecklats, för att man ska kunna testa andra delar.

styrkort: En representation av sammanställda mätningar som representerar arbetet med att uppnå långsiktiga mål. Ett styrkort innehåller mätningar av prestanda över tiden eller vid slutet av ett bestämt tidsintervall. Se även balanserat styrkort, dashboard och corporate dashboard.

SUMI (Software Usability Measurement Inventory): En frågebaserad metod för användbarhetstest, vilken mäter programvarans kvalitet från ett slutanvändarperspektiv. [Veenendaal]

SUMI: Se Software Usability Measurement Inventory.

syntaxtestning: En black-box-testdesignteknik för att skapa testfall för en komponent eller ett system som baseras på definitionen av indataområden och/eller utdataområden.

system av system: Heterogena multipla distribuerade system som är inbyggda på flera olika nivåer i nätverk och över flera ihopkopplade områden, som adresserar storskaliga problem och syften interdisciplinärt utan att ha ett gemensamt kontrollsystem.

system under test: Se testobjekt.

system: En samling komponenter organiserade på ett sätt så att en specifik funktionalitet eller mängd av funktionalitet uppnås. [IEEE 610]

Systematic Test and Evaluation Process: En strukturerad testmetodik som även kan användas som en innehållsbaserad modell för testprocessförbättring. Systematic Test and Evaluation Process (STEP) kräver inte att förbättringar genomförs i någon specifik ordning. Se även innehållsbaserad modell.

systemintegrationstestning: Testning vid integrationen av ett system. Vid integrationstestning testas gränssnitten mellan systemet och dess miljö.

systemtestning: En testprocess där ett integrerat system testas i syfte att verifiera att systemet motsvarar ställda krav. [Hetzel]

säkerhet: Förmågan hos ett system att uppnå acceptabla risknivåer för att människor, verksamhet, programvara, egendom och miljö inte ska komma till skada vid en speciell användning. [ISO 9126]

säkerhetskritiska system: Ett system vars felsymptom eller haveri kan leda till dödsfall, orsaka skador på personer, utrustning eller miljön.

säkerhetstestning: Testning för att avgöra säkerheten hos en programvaruprodukt.

T

TDD: Se testdriven utveckling.

teknisk granskning: En kollegial gruppdiskussion där fokus är att uppnå samstämmighet om vilka tekniska angreppssätt som ska användas. [Gilb och Graham, IEEE 1028] Se även kollegial granskning.

Test Maturity Model Integration (TMMi): Ett fem-steps ramverk för förbättring av testprocess relaterad till CMMI (Capability Maturity Model Integration), som beskriver de viktigaste delarna i en effektiv testprocess.

test: En mängd av ett eller flera testfall. [IEEE 829]

testanalys: Processen att analysera testbasen och fastställa testmål.

testare: En person med rätt kompetens för aktiviteten som är involverad i testningen av en komponent eller ett system.

testarkitekt: (1) En person som tillhandahåller vägledning och strategisk ledning för en testorganisation och för dess relation med andra discipliner. (2) En person som definierar sättet som testning är strukturerad för ett givet system, inklusive områden som testverktyg och testdatahantering.

testautomatisering: Användandet av programvara för att kontrollera exekvering av test, jämförelse mellan förväntat resultat och verkligt resultat, konfigurerings av testmiljö och andra testkontroll- och testrapporteringsfunktionaliteter.

testavslut: Aktiviteter utförda för att samla in data från tidigare avslutade testaktiviteter, fakta och mätvärden. Testavslutsfasen består av avslutande och arkivering av testvara, utvärdering av testprocessen samt framtagande av en utvärderingsrapport. Se även testprocess.

testavslutskriterier: Se avslutskriterier.

testbara krav: Krav som är skrivna på sådant sätt att de möjliggör etablering av testdesign (och därefter testfall) samt exekvering av tester för att kontrollera om kraven har blivit uppfyllda. [Efter IEEE 610]

testbarhet: Kapaciteten hos programvaruprodukten att tillåta ändrad programvara att kunna testas. [ISO 9126] Se även underhållbarhet.

testbarhetsgranskning: En detaljerad kontroll av testbasen för att bestämma om den har en tillfredsställande kvalitet för att tjänstgöra som indokument till testprocessen. [Efter TMap]

testbas: All dokumentation från vilken man kan härleda krav för en komponent eller ett system, dvs. den dokumentation som testfall baseras på. Om ett dokument endast kan läggas till genom en formell process så kallas testbasen för en frusen testbas. [Efter TMap]

testbädd: även testbänk. Se testmiljö.

testcharter: En redogörelse (lista) för testmål och möjliga testidéer. Testlistor används bl.a. vid utforskande testning. Se även utforskande testning.

testchef: En senior testledare som leder testledare. Se även testledare.

testcykel: Genomförande av en testprocess på en enskild identifierbar utgåva av ett testobjekt.

testdata: Datavärden som existerar (till exempel i en databas) före exekverandet av ett test och som påverkar eller påverkas av den testade komponenten eller det testade systemet.

Kommentar: Observera att vissa (ISTQB) definierar testdata som de indata som krävs för att köra ett testfall, medan andra (IEEE) inkluderar även testresultat i begreppet.

testdatahantering: Processen att analysera krav på testdata, designa testdatastrukturer, skapa och underhålla testdata.

testdesign: (1) Se testdesignspecifikation.

(2) Process för överföring från generella testmålsättningar till konkreta testvillkor och testfall.

testdesignspecifikation: Ett dokument som specificerar testvillkor för ett testelement, detaljerat angreppssätt och tillhörande högnivåtestfall. [Efter IEEE 829] Se även testspecifikation.

testdesigntechnik för icke-funktionella egenskaper: Procedur för att skapa och välja testfall för en komponent eller ett system utan referens till dess interna struktur. Se även black-box-testdesigntechnik.

testdesigntechnik för strukturella tester: Se white-box-testdesigntechnik.

testdesigntechnik: En metod att härleda, välja och/eller specificera testfall.

testdesignverktyg: Ett verktyg som stödjer testdesignaktiviteten genom att generera indata från en specifikation, som kan finnas i en CASE-databas, t.ex. kravhanteringsverktyg. Data kan också genereras från speciella testvillkor som finns i själva verktyget.

testdriven utveckling: Ett sätt att utveckla programvara där testfall tas fram, och ofta automatiseras, innan programvaran som dessa testfall ska exekveras mot har tagits fram.

testdrivrutin: Se drivrutin.

testelement: Ett individuellt element för test. Normalt är det ett testobjekt och flera testelement. Se även testobjekt.

testexekvering: även testgenomförande. Processen att exekvera en testprocedur mot en komponent eller ett system som är under test för att producera ett faktiskt resultat.

testexekveringsfas: Den tidsperiod i en programvaras utvecklingslivscykel då komponenter inom en programvaruprodukt exekveras och programvaruprodukten utvärderas för att avgöra om kraven är rätt implementerade. [IEEE610]

testexekveringsplattform: även testharness. En testmiljö som består av de stubbar och drivrutiner som behövs för att genomföra en test.

Kommentar: En testexekveringsplattform kan möjliggöra testning av en del av ett system genom att simulera miljön i vilken testet skall exekvera. En testexekveringsplattform kan också användas för att instrumentera systemet och rapportera testresultat.

testexekveringsschema: Testprocedurer ingående i testexekveringsschemat i rätt sammanhang och i den ordning de skall exekveras.

testexekveringsteknik: Metod som används för testexekvering, antingen manuell eller automatiserad.

testexekveringsverktyg: En typ av testverktyg som utför exekvering av annan programvara genom automatiserade testskript, t.ex. in- och uppspelningsverktyg. [Fewster och Graham]

testfall: En samling indata, förutsättningar inför exekvering, förväntat resultat och sluttillstånd efter exekvering, som är framtagna för ett specifikt objekt eller testvillkor. Används för att verifiera uppfyllelsen av ett specifikt krav hos en utpekad programdel. [Efter IEEE 610]

testfallsdesigntechnik: Se testdesigntechnik.

testfallsgenerator: Se verktyg för förberedelse av testdata.

testfallsspecifikation: Ett dokument som specificerar en mängd av testfall (mål, indata, testaktiviteter, förväntat resultat och exekveringsförutsättningar) för ett testobjekt. Se även testspecifikation.

testfallssvit: Se testsvit.

testfas: En bestämd uppsättning av testaktiviteter som är samlade inom en hanterbar del av projektet, t.ex. exekveringsaktiviteter för en testnivå. [Efter Gerrard]

testfel: Se fel.

testförbättringsplan: En plan för att uppnå organisatoriska testprocessförbättringar baserad på en djup förståelse för organisationens styrkor och svagheter samt dess testprocesser och testprocesstillgångar. [efter CMMI]

testimplementation: Process för att utveckla och prioritera testprocedurer, skapa testdata och, valfritt, att förbereda för testexekveringsplattform och skriva automatiserade testskript.

testincident i programvara: Se avvikelse.

testincident: Se avvikelse.

testincidentrapport för programvara: Se avvikelserapport.

testincidentrapport: Se avvikelserapport.

testindata: Data från en extern källa in till testobjektet som används under testgenomförande. Den externa källan kan vara hårdvara, programvara eller användare.

testinfrastruktur: Organisatoriska artefakter som är nödvändiga för att utföra tester och består av testmiljö, testverktyg, kontorsmiljö och processer.

testjämförelse: Processen att identifiera skillnader mellan det verkliga resultatet (som har producerats av en komponent eller en programvara som testas) och det förväntade resultatet. Testjämförelse kan utföras under testexekvering (dynamisk jämförelse) eller efteråt.

testkomparator: Ett testverktyg som jämför det verkliga resultatet (som har producerats av objektet som testats) med det förväntade resultat för det specifika testfallet.

testkoordinator: Se testledare.

testkrav: Se testvillkor.

testkörning: Exekvering av en delmängd av en test på en specifik version av testobjektet.

testkörningslogg: Se testlogg.

testledare: Den person som ansvarar för att projektleda testaktiviteter och resurser samt utvärdera testobjekt. Personen leder, reglerar, administrerar, planerar och styr utvärderingen av testobjekt.

testledning: Planering, uppskattning, övervakning och reglering av testaktiviteter. Testledning utförs typiskt av en testledare. Även strukturering och åtkomst av testfall kan räknas till testledning.

testledningsverktyg: Verktyg som stödjer testledningen. Exempel på funktioner i testledningsverktyg är: stöd för oberoende versionshantering, gränssnitt för testexekveringsverktyg, stöd för spårbarhet av test, loggning av testresultat osv.

testleverabel: Varje test[arbets]produkt som ska levereras till ytterligare någon utöver test[arbets]produktens författare. Se även leverabel.

testlogg: En kronologiskt ordnad logg bestående av detaljinformation rörande en testexekvering. [IEEE 829]

testloggning: Processen att registrera information om exekverade tester i en testlogg.

testmiljö: En miljö innehållande hårdvara, instrumentation, simulatorer, programvaruverktyg och andra hjälpmedel som krävs för att utföra testning. [After IEEE 610]

testmål: En mängd av avslutskriterier.

testmålsättning: Ett syfte eller en anledning som ligger till grund för utveckling och exekvering av testfall.

testmängd: Se testsvit.

testning av begränsad tillståndsmaskin: Se tillståndsbaserad testning.

testning av databasintegritet: Testning av metoder och processer som används för åtkomst och hantering av data(baser), för att säkerställa att åtkomstmetoder, processer och dataregler fungerar som väntat. Vid åtkomst av databas ska också test utföras för att säkerställa att data inte förstörs och att data inte oväntat raderas, uppdateras eller skapas.

testning av dataflöde: En testdesign teknik där testfallen är designade för att utöva definitions- och användningspar av variabler.

testning av dataintegritet: Se testning av databasintegritet.

testning av delkomponent: Se ekvivalensklassfördelning. [Beizer]

testning av dokument: Testning av kvaliteten på dokument, t.ex. användarmanualer och installationsinstruktioner.

testning av icke-funktionella egenskaper: Testning av egenskaper som inte relaterar till funktionalitet. Exempel på sådana egenskaper är tillförlitlighet, effektivitet, användbarhet, underhållbarhet och portabilitet.

testning av installationsbarhet: Testning av en programvaruprodukts installationsbarhet. Se även portabilitetstestning.

testning av kombinerade villkor: Se villkorskombinationstestning.

testning av samtidiga händelser: Tidsrelaterad testning, där syftet är att avgöra hur händelseförloppet hanteras av komponenten eller systemet. Detta händelseförlopp bestående av två eller flera aktiviteter sker inom samma tidsintervall, antingen genom tidsstyrning eller någon speciell ordning inom intervallet av aktiviteterna eller av samtidig exekvering. [After IEEE 610]

testning av villkorsbeslut: Se modifierad testning av villkorsbeslut

testning av ändamålsenlighet: Den testaktivitet som ska säkerställa ändamålsenligheten hos en programvaruprodukt.

testning baserad på ortogonal matris: Ett systematiskt angreppssätt för testning med hjälp av alla parkombinationer av variabler härledda från ortogonala matriser. Angreppssättet reducerar antalet av alla kombinationer till alla parkombinationer att testa. Se även n-vis testning, parvis testning.

testning för typgodkännande: Den testprocess som bestämmer att komponenten eller systemet uppfyller givna regler, d.v.s. överensstämmer med viss accepterad standard.

testning för överensstämmelse: Se testning för typgodkännande.

testning med hjälp av beslutstabeller: Testning med hjälp av en beslutstabell, vilken sammanfattar villkor och dess resulterande händelser eller beslut. Se även beslutstabell.

testning utifrån användarscenarion: Se användningsfallsbaserad testning.

testning: Process som omfattar alla livscykelaktiviteter för en programvaruprodukt, både statiska och dynamiska, samt behandlar planering, förberedelse och utvärdering av programvaruprodukter för att avgöra om de uppfyller specificerade krav, för att demonstrera att de är klara för användning och för att hitta fel.

testnivå: En grupp av testaktiviteter som organiseras och hanteras tillsammans. En testnivå är kopplad till ansvarsfördelningen i ett projekt. Exempel på testnivåer är komponenttest, integrationstest, systemtest och acceptanstest. [Efter TMap]

testobjekt: Den komponent eller det system som testas. Se även testelement.

Kommentar: Vid test av två eller flera objekt används även benämningen testelement.

testomständigheter: Se testvillkor.

testorakel: En källa som används för att jämföra förväntat resultat med aktuellt resultat från programvaran som testas. Ett orakel kan vara ett befintligt system (för benchmarktestning), en kvalitetssäkrad användarmanual, annan programvara eller en persons speciella kunskaper, men får inte vara själva koden. [Efter Adrion]

testplan: Ett dokument som beskriver målsättning, angreppssätt, resurser och tidplaner för de avsedda testaktiviteterna. Där identifieras bland annat testelement, funktioner som ska testas, testuppdrag som ska göras, vem som ska utföra testuppdragen, nivån på testberoende, testmiljö, testdesigntechniker, mättekniker, förutsättningar och sluttillstånd som ska användas med logisk grund, samt risker som kräver åtgärdsplaner. Testplanen är en nedteckning av planeringsprocessen. [efter IEEE 829]

testplanering: Att ta fram eller uppdatera en testplan.

testpolicy: Ett högnivådokument som beskriver principer, angreppssätt och större mål för organisationen när det gäller testning.

testprocedur: även testinstruktion. Se testprocedurspecifikation.

testprocedurspecifikation: Ett dokument med detaljerade instruktioner som beskriver hur exekveringen skall gå till för ett eller flera testfall. Kallas även för testskript eller manuellt testskript. [Efter IEEE 829] Se även testspecifikation.

testprocess: Den grundläggande testprocessen omfattar testplanering och kontroll, testanalys och -design, testimplementation och -exekvering, utvärdering av avslutskriterier och rapportering, samt aktiviteter för testavslut.

testprocessförbättrare: Person som genomför förbättringar av testprocessen, baserat på en testförbättringsplan.

testprocessgrupp: En samling [test]specialister som möjliggör skapande, underhåll och förbättringar av de testprocesser som används i en organisation. [efter CMMI]

testprogressrapport: Ett dokument där man summerar testaktiviteter och resultat med syfte att rapportera framsteg för testaktiviteter jämfört med en fastställd konfiguration (såsom den ursprungliga testplanen) och för att informera ledningen om risker och alternativ som kräver beslut.

testprotokoll: Se testlogg.

testrapport: Se slutlig testrapport och rapport för testprogress

testrapportering: Insamling och analys av data från testaktiviteter och därefter sammanställande av denna data i en rapport som information till intressenter.
Se även testprocess.

testregistrering: Se testloggning.

testreproducerbarhet: En testegenskap där samma resultat nås varje gång testen exekveras.

testresultat: Se resultat.

testscenario: Se testprocedurspecifikation.

testschema: En lista av aktiviteter, uppgifter eller händelser för en testprocess, där man identifierar möjliga startdatum och starttider och/eller slutdatum och sluttider, och beroenden mellan dessa.

testsession: En fortlöpande tidsperiod utan avbrott där tiden används för att utöva test. I utforskande testning är varje testsession koncentrerat till en lista, men testare kan också utforska nya möjligheter och problem vid en session. Testaren skapar och utövar testfall när dessa behövs och loggar resultatet. Se även utforskande testning.

testskript: även testkod. Syftar vanligen på den automatiserade testproceduren som används med en textexekveringsplattform.

testspecifikation: Ett dokument som består av en testdesignspecifikation, testfallsspecifikation och/eller testprocedurspecifikation

testspecifikationsteknik: Se testdesignteknik.

teststeg: Se testnivå.

teststrategi: En högnivåbeskrivning av de nivåer som skall testas och hur testningen skall gå till i dessa nivåer inom ett eller fler projekt.

teststyrning: En testledningsaktivitet som hanterar utveckling och tillämpning av vad som planerats. Se även testledning.

testsvit: Ett antal testfall för en komponent eller ett system, där utdata från ett testfall ofta används som förutsättning för nästa.

testteknik: Se testdesignteknik.

testtyp: även testsort, typ av test, testgrupp. Gruppering av testfall efter sort eller typ av testning som testfallen utför, av en viss kvalitet eller attribut, t.ex. gränssnittstest, regressionstest, robusthetstest. Grupperingen är oberoende av testfas och testnivå. [Efter TMap]

testtäckning: Se täckningsgrad.

testuppdrag: Syftet med test för en organisation, ofta dokumenterad som en del av testpolicy. Se även testpolicy.

testuppskattning: En beräknad uppskattning av testrelaterade resultat (t.ex. tidsåtgång, slutdatum, kostnader, antal testfall etc.) som är användbar även om indata är ofullständigt, ovisst eller oklart.

testuppsättning: Se testmiljö.

testutfall: Se resultat.

testutvärderingsrapport: Ett dokument som produceras i slutet av testprocessen för att summera alla testaktiviteter och resultat. Det innehåller också utvärdering av testprocessen och erfarenhetsreflektioner.

testvara: även testartiklar, testartefakter. De artefakter (producerade under testprocessen) som krävs för att planera, designa och exekvera test. Exempel på testartiklar är dokumentation, skript, indata, förväntade resultat, databaser, testmiljö och andra programvaror eller produkter som använts i testningen. [Efter Fewster and Graham]

testverktyg för informationssäkerhet: Ett verktyg som tillhandahåller stöd för test av säkerhet och sårbarhet.

testverktyg: Ett system eller program vars syfte är att underlätta testning.

testvillkor: Ett objekt eller en händelse hos en komponent eller ett system som kan verifieras med ett eller flera testfall. Ett objekt eller en händelse kan i detta fall vara en funktion, en transaktion, ett kvalitetsattribut eller en strukturell programvarudel.

testövervakning: även testbevakning. En testledningsaktivitet som ger återkoppling till och synlighet för testaktiviteter. Information av intresse kan inhämtas manuellt eller automatiskt och användas för att mäta avslutskriterier (t.ex. täckning). Informationen kan också användas för att fastställa projektets framåtskridande. Exempel på vanlig testövervakningsinformation är:

- avklarad andel testförberedelse
- antal exekverade testfall
- subjektiva reflektioner från testare
- testkostnader

tidsuppträdanden: Se prestanda.

tillförlitlighet: Kapaciteten hos en programvaruprodukt att utföra specificerad funktionalitet under vissa förutsättningar för en viss tidsperiod eller med ett visst antal användningar av programvaran. [ISO 9126]

tillförlitlighetstestning: Processen att utföra testning i syfte att bestämma tillförlitligheten hos en programvaruprodukt.

tillgänglighet: Den grad till vilken en komponent eller ett system är i drift och tillgängligt för användning. Uttrycks ofta i procent. [IEEE 610]

Se även tillgänglighetstestning (accessibility testing)

tillgänglighetstestning (accessibility testing): Testning för att fastställa hur lättanvänt komponenten eller systemet är för människor med funktionshinder. [Gerrard]

Se även tillgänglighet

tillståndsbaserad testning: En testteknik som innebär att man gör testfall avsedda att testa en eller flera tillståndsovergångar. Se även test av tillståndsovergångssekvenser av längd N+1..

Kommentar: Användbar teknik på alla nivåer av testning. Ett tillståndsdigram består av tillstånd och tillståndsovergångar. Denna modell kan skapas direkt från koden eller på ett mera schematiskt tillstånd, t.ex. ett användningsfall.

tillståndsdigram: Ett diagram som skildrar de tillstånd som komponenten eller systemet kan anta och visar de händelser och omständigheter som orsakar och/eller är resultatet av förändringar från ett tillstånd till ett annat. [IEEE 610]

tillståndstabell: En tabell som för varje tillstånd och för varje händelse visar tillåtna och otillåtna övergångar i ett system eller en komponent.

tillståndsovergång: En övergång mellan två tillstånd i ett system eller en komponent.

tilltalande: En egenskap hos ett program eller system som innebär att programmet eller systemet är tilltalande för användaren. [ISO 9126] Se även användbarhet.

tillverkningsbaserad kvalitet: Ett sätt att se på kvalitet som innebär att kvaliteten hos en produkt eller tjänst avgörs av i vilken grad den stämmer överens med dess ursprungliga design och krav. Kvaliteten i produkten uppkommer genom de använda utvecklings- eller tillverkningsprocesserna. [Efter Garvin] Se även användarbaserad kvalitet, produktbaserad kvalitet, upplevd kvalitet och värdebaserad kvalitet.

TMMi: Se Test Maturity Model integration.

tolerans mot felaktighet: Se feltolerans.

Total Quality Management: En ledningsfilosofi som fokuserar på kvalitet i hela organisationen och som bygger på att alla medarbetare deltar. Målet är att uppnå långsiktig framgång dels genom nöjda kunder och dels genom att gagna organisationen och samhället. Total Quality Management består av aktiviteterna planering, organisering, vägledning, styrning och försäkring. [Efter ISO 8402]

TPA (Test Point Analysis): även testmängdsanalys. En formelstyrd testberäkningsmetod som baserar sig på funktionsgradsanalys (FPA). [TMap]

TPG: Se Test Process Group.

TPI Next: Ett fortlöpande affärsdrivet ramverk för testprocessförbättring som beskriver nyckelelementen i en ändamålsenlig och effektiv testprocess.

TQM: Se Total Quality Management.

transaktionsanalys: Analys av transaktioner mellan människor och internt inom en persons medvetande. En transaktion har formen av ett stimulus-respons-par. Transaktioner sker mellan människor och mellan s.k. egotillstånd (personlighetssegment) internt inom en persons medvetande.

trepunktsestimering: En testestimeringsmetod som använder estimerade värden för "bästa fall", "värsta fall" och "mest troliga fall" rörande det som estimeras, för att definiera graden av säkerhet rörande det erhållna estimatet.

trådtestning: En variant av komponentintegrationstestning, där den progressiva integrationen av komponenter följer implementationen av delmängder av krav, i motsats till integrationen av komponenter i hierarkiska nivåer.

typgodkännande: Kapacitet hos programvaran att ansluta till standarder, gängse normer eller legala bestämmelser och liknande föreskrifter. [ISO 9126]

typprov: (1) En standard med vilken mätningar och jämförelser kan göras.

(2) En test som används för att jämföra komponenter eller system med varandra eller med en standard som i (1). [Efter IEEE 610]

Kommentar: Till exempel testning av protokoll vid datakommunikation, såsom SEMCO.

täckning av villkorsbeslut: Se modifierad täckning av villkorsbeslut

täckningsanalys: Analys av den mättnivå, procentuellt uttryckt, till vilket ett specificerat objekt har blivit täckt av en testsvit. Ofta avses att man uppfyller en på förhand given testnivå, där flera testfall kan behövas för att nå den.

täckningsgrad: Den nivå, uttryckt i procent, till vilket ett specificerat objekt har blivit exekverat av en testsvit.

täckningsmätning för Chows N-övergångar: Se täckning av tillståndsovergångssekvenser av längd N+1.

täckningsobjekt: Den del man mäter täckning på, t.ex. ekvivalensklasser eller kodrader.

U

undantagshantering: Beteendet hos en komponent eller ett system som blivit utsatt för felaktigt invärde, antingen inmatat av en användare eller från en annan komponent eller system, eller p.g.a. ett internt fel.

underhåll: även förvaltning. Modifiering av en programvaruprodukt efter leverans för att rätta defekter, förbättra prestanda eller andra egenskaper, eller anpassa produkten till en förändrad miljö. [IEEE 1219]

Kommentar: Med miljö menas här den miljö som programvaran används i, dvs. datorer och kringutrustning som programvaran är beroende av.

underhållbarhet: Hur lätt en programvara kan modifieras för att rätta fel, tillgodose nya krav, underlätta framtida underhåll eller anpassa till förändrad miljö. [ISO 9126]

Kommentar: Med miljö menas här den miljö som programvaran används i, dvs. datorer och kringutrustning som programvaran är beroende av.

underhållbarhetstestning: Testprocess för att bestämma underhållbarheten hos en programvaruprodukt.

Kommentar: Några specifika förvaltningsmål kan vara att det finns dokumentation, testfall och kunskap om hur systemet är uppbyggt.

underhållstestning: även förvaltningstestning. Test av modifiering av ett system som tagits i drift eller test av hur en ändrad omgivning påverkar ett drifttaget system.

underkänd: En test anses underkänd om dess faktiska resultat inte stämmer mot förväntat resultat.

uppifrån och ner-testning: även top-down-testning. En inkrementell integrationstestteknik, där komponenterna på den högsta nivån testas först med simulering av lågnivåkomponenter med hjälp av stubbar. De testade komponenterna används sedan för att testa lågnivåkomponenterna. Processen upprepas tills alla lågnivåkomponenter har testats. Jmf nerifrån och upp-testning. Se även integrationstestning.

utdata: En variabel till vilken en komponent skriver ett utvärde.

Kommentar: En variabel kan finnas inuti eller utanför en komponent.

utdatadomän: även utdatamängd, resultatmängd. Mängden av alla möjliga utdata.

utdatavärde: även utvärde. En instans av utdata.

utfall av villkor: Utvärdering av ett villkor, som kan vara antingen sant eller falskt.

utfall: Se resultat.

utfallet från ett kodvillkor: Resultatet av ett beslut, som avgör vilket specifikt kontrollflöde, dvs. vilken kodgren som kommer att exekveras.

utforskande testning: även upptäckande testning. En informell testdesignteknik där testaren aktivt styr utveckling av testfall under det att de exekveras och använder information från testningen för att utveckla nya och bättre testfall. [Efter Bach]

uttömmande testning: även fullständig testning. Ett angreppssätt där testsviten består av alla kombinationer av indata och förutsättningar.

utvecklingstestning: Formell eller informell testning som utförs under utvecklingen av en komponent eller ett system. Utförs oftast i utvecklingsmiljön av utvecklarna själva. [Efter IEEE 610]

utvärderare: Person som genomför en utvärdering; medlem i ett utvärderingsteam.

utvärdering: Se testning.

utvärderingsledare: Den person som leder en processutvärdering. I vissa fall, t.ex. i CMMi och TMMi, vilka innehåller formella utvärderingar, måste huvudprocessutvärderaren vara formellt utbildad och ackrediterad.

utvärderingsrapport: Ett dokument som sammanfattar resultaten av en utvärdering, d.v.s. slutsatser, rekommendationer och observationer. Se även processutvärdering.

utövad: även körd, motionerad. En programdel utövas/motioneras av ett testfall när ett invärde medför en exekvering av programdelen, såsom sats, villkor eller andra strukturella konstruktioner.

V

validering: Bekräftelse genom undersökning och genom framläggande av sakliga bevis för att kraven för en specifikt avsedd användning eller specifik tillämpning har uppfyllts. [ISO 9000]

variabel: En namngiven lagringsplats i en dator som är nåbar av en programvara genom att programmet refererar till namnet på lagringsplatsen.

verifiering av bygge: En uppsättning automatiserade tester som validerar integriteten hos varje nytt bygge och verifierar dess nyckel-/kärnfunktionalitet, stabilitet och testabilitet. Detta är en branschpraxis när det sker täta leveranser av byggen (t.ex. i agila projekt) och det körs på varje nytt bygge innan det släpps för vidare testning. Se även röktest.

verifiering: Bekräftelse genom undersökning och genom framläggande av sakliga bevis för att specificerade krav har uppfyllts. [ISO 9000]

verksamhetsbaserad testning: Ett angreppssätt för testning där testfall tas fram baserat på beskrivningar av och/eller kunskap om verksamhetsprocessen.

verktyg för att mäta täckningsgrad: Se verktyg för test av täckningsgrad.

verktyg för förberedelse av testdata: Verktyg som interagerar med databaser, filer eller dataöverföringar för att förbereda testdata för testexekvering.

verktyg för informationssäkerhet: Ett verktyg som stöder operativ säkerhet, t.ex. en brandvägg.

verktyg för injektion av felaktighet: Se felinjektionsverktyg.

verktyg för statistisk analys: Ett verktyg som utför statistisk analys.

verktyg för statistisk kodanalys: En kategori av verktyg för statistisk analys som enbart riktar in sig på analys av källkod. Verktöget analyserar koden med avseende på t.ex. tillämpningsgrad av kodstandarder, kvalitetsmått eller dataflödesavvikelser.

verktyg för stresstestning: Ett verktyg (program) som stödjer stresstestning. Se även stresstestning.

verktyg för test av täckningsgrad: Ett verktyg som visar hur strukturella element, t.ex. kodsatser och kodgrenar, har exekverats av en testsvit.

versionshantering: Se konfigurationskontroll.

vertikal spårbarhet: Spårning av krav genom lager av utvecklingsdokumentation ner till komponenter.

villkor: Ett logiskt uttryck som kan resultera i Sant eller Falskt i en utvärdering, t.ex. $A > B$. Se även villkorstestning.

villkorskombinationstestning: En strukturell testteknik där testfallen designas för att exekvera kombinationer av enstaka villkorsresultat, villkor som finns i en sats.

villkorskombinationstäckning: Den procentuella täckningen av kombinationen av alla villkor i ett kodgrensvillkor som finns i en kodrad och dess resultat som har exekverats av en testfallssvit. Med 100 % täckning av villkorskombinationer är det underförstått att 100 % MCDC täckning också har uppfyllts. Se även kodtäckning av kombinationer av kodgrensvillkor.

villkorstestning: En testteknik vars syfte är att skapa testfall på programkod, vilka exekverar villkorets olika möjliga utfall.

villkorstäckning: Det procentuella utfallet av villkor som har exekverats av en testsvit. 100 % villkorstäckning kräver att varje villkor i varje beslutspunkt måste testas som SANT och FALSKT.

V-modell: Ett ramverk som beskriver aktiviteterna inom programvaruutveckling. V-modellen beskriver ur ett testperspektiv hur testaktiviteter kan integreras i varje fas av livscykeln för programvaruutveckling.

volymtestning: Testning där systemet utsätts för hantering av stora datavolymer. Se även resursanvändningstestning.

väg: En sekvens av exekverbara satser i en komponent som börjar vid en startpunkt och avslutas vid en slutpunkt.

vägtestning: Teknik för att skapa testfall som exekverar valda vägar hos en komponent.

vägtäckning: Den procentuella delen av vägar som har exekverats genom testfall i en testfallssvit. 100 % vägtäckning innebär LCSAJ-täckning.

värdebaserad kvalitet: Ett sätt att se på kvalitet som innebär att kvaliteten hos en produkt avgörs av produktens pris. Produkter eller tjänster är av hög kvalitet om de tillhandahåller önskad prestanda för en acceptabel kostnad. Kvaliteten hos en produkt avgörs i form av en beslutsprocess hos en intressent där tids, ansträngnings och kostnadsaspekter vägs mot varandra. [Efter Garvin] Se även användarbaserad kvalitet, produktbaserad kvalitet, tillverkningsbaserad kvalitet och upplevd kvalitet.

W

WAMMI: Se Website Analysis and MeasureMent Inventory.

WBS: Se Work Breakdown Structure.

Website Analysis and MeasureMent Inventory (WAMMI): En enkätbaserad testteknik inom användbarhet för att bedöma webbsidors mjukvarukvalitet ur ett slutanvändarperspektiv.

white-box-teknik: Se white-box-testdesignteknik.

white-box-testdesignteknik: Procedur för att härleda och/eller välja testfall baserat på en analys av den interna strukturen hos en komponent eller ett system.

white-box-testning: Testning baserad på analys av den interna strukturen hos en komponent eller ett system.

Wide Band Delphi: En testberäkningsteknik som är baserad på expertis med målsättning att göra en exakt (eller noggrann) uppskattning genom alla teammedlemmars gemensamma kompetens.

wild pointer: En pekare som refererar till en plats utanför områdesgränser för pekaren eller till en plats som inte finns. Se även pekare.

Work Breakdown Structure: Ett sätt att beskriva arbetsmoment, deras förhållande till varandra och till den slutliga produkten. [CMMI]

Å

återhämtningsbarhetstestning: Testprocess som syftar till att säkerställa återhämtningsbarheten hos en programvaruprodukt. Se även tillförlitlighetstestning.

återhämtningsförmåga: Programvaruproduktens kapacitet att återuppta en specificerad nivå av prestanda och att återfå direkt påverkade data när fel inträffat. [ISO 9126] Se även tillförlitlighet.

återhämtningstestning: Se återhämtningsbarhetstestning.

återstartskrav: En uppsättning testaktiviteter som måste repeteras när testning återupptas efter ett avbrott. [Efter IEEE 829].

återstartskriterier: Testaktiviteter som måste repeteras när testningen återupptas efter ett avbrott.

Ä

ändamålsenlighet: Förmågan att producera avsett resultat. Se även effektivitet.

ändamålsenlighet: Programvaruproduktens kapacitet att tillhandahålla ändamålsenlig funktionalitet för specificerade uppgifter och användarmål. [ISO 9126] Se även funktionalitet.

ändringshantering: (1) Ett strukturerat sätt att överföra individer och organisationer från ett nuläge till ett önskeläge. (2) Ett kontrollerat sätt att åstadkomma en förändring, eller föreslagen förändring, i en produkt eller en tjänst. Se även konfigurationshantering.

ändringsråd: Se Configuration Control Board.

ändringsstyrning: Se konfigurationskontroll.

Ö

öppningstest: Ett exempel på röktest med syfte att avgöra om en komponent eller ett system är redo för finjustering eller fortsatt testning. En öppningstest är typiskt något som utförs i början av en testexekveringsfas. Se även röktest.

överensstämmelse: Grad av likformighet, standardisering och avsaknad av motsägelser bland dokument eller delar av en komponent eller ett system. [IEEE 610]

övergripande integrationstestning: Se systemintegrationstestning.

övergripande testplan: En testplan som normalt riktar sig mot flera testnivåer. Se även testplan.

övervakare: Programvaruverktyg eller hårdvaruenhet som används samtidigt med en komponent eller system vid testning och som övervakar, spelar in och/eller analyserar beteendet hos komponenten eller systemet. [Efter IEEE 610]

övervakningsverktyg: Se övervakare.

Bilaga A

Förteckning över icke-normgivande källor som använts vid framtagande av denna ordlista.

- [Abbott] J. Abbot (1986), *Software Testing Techniques*, NCC Publications.
- [Adrion] W. Adrion, M. Branstad and J. Cherniabsky (1982), Validation, Verification and Testing of Computer Software, in: *Computing Surveys*, Vol. 14, No 2, June 1982.
- [Bach] J. Bach (2004), Exploratory Testing, in: E. van Veenendaal, *The Testing Practitioner – 2nd edition*, UTN Publishing, ISBN 90-72194-65-9.
- [Beizer] B. Beizer (1990), *Software Testing Techniques*, van Nostrand Reinhold, ISBN 0-442-20672-0
- [Chow] T. Chow (1978), Testing Software Design Modelled by Finite-State Machines, in: *IEEE Transactions on Software Engineering*, Vol. 4, No 3, May 1978.
- [CMM] M. Paulk, C. Weber, B. Curtis and M.B. Chrissis (1995), *The Capability Maturity Model, Guidelines for Improving the Software Process*, Addison-Wesley, ISBN 0-201-54664-7
- [CMMI] M.B. Chrissis, M. Konrad and S. Shrum (2004), *CMMI, Guidelines for Process Integration and Product Improvement*, Addison Wesley, ISBN 0-321-15496-7
- [Fenton] N. Fenton (1991), *Software Metrics: a Rigorous Approach*, Chapman & Hall, ISBN 0-53249-425-1
- [Fewster and Graham] M. Fewster and D. Graham (1999), *Software Test Automation, Effective use of test execution tools*, Addison-Wesley, ISBN 0-201-33140-3.
- [Freedman and Weinberg] D. Freedman and G. Weinberg (1990), *Walkthroughs, Inspections, and Technical Reviews*, Dorset House Publishing, ISBN 0-932633-19-6.
- [Gerrard] P. Gerrard and N. Thompson (2002), *Risk-Based E-Business Testing*, Artech House Publishers, ISBN 1-58053-314-0.
- [Gilb and Graham] T. Gilb and D. Graham (1993), *Software Inspection*, Addison-Wesley, ISBN 0-201-63181-4.
- [Graham] D. Graham, E. van Veenendaal, I. Evans and R. Black (2007), *Foundations of Software Testing*, Thomson Learning, ISBN 978-1-84480-355-2
- [Grochtmann] M. Grochtmann (1994), Test Case Design Using Classification Trees, in: *Conference Proceedings STAR 1994*.
- [Hetzel] W. Hetzel (1988), *The complete guide to software testing – 2nd edition*, QED Information Sciences, ISBN 0-89435-242-3.
- [McCabe] T. McCabe (1976), A complexity measure, in: *IEEE Transactions on Software Engineering*, Vol. 2, pp. 308-320.
- [Musa] J. Musa (1998), *Software Reliability Engineering Testing*, McGraw-Hill Education, ISBN 0-07913-271-5.

- [Myers] G. Myers (1979), *The Art of Software Testing*, Wiley, ISBN 0-471-04328-1.
- [TMap] M. Pol, R. Teunissen, E. van Veenendaal (2002), *Software Testing, A guide to the TMap Approach*, Addison Wesley, ISBN 0-201-745712.
- [Veenendaal] E. van Veenendaal (2004), *The Testing Practitioner – 2nd edition*, UTN Publishing, ISBN 90-72194-65-9.

Bilaga B

Förslag och kommentarer till förbättringar av denna ordlista är varmt välkomna.

När förslag och kommentarer skickas glöm inte att ange följande information:

- ditt namn och kontaktmöjligheter,
- versionsnummer för den ordlista kommentaren gäller (denna ordlista, v2.1),
- detaljerad information om vilken del av ordlistan kommentaren gäller och
- information om föreslagen ändring och en kort motivering varför ändringen ska införas.

Förslag och kommentarer skickas till *review@sstb.se*.