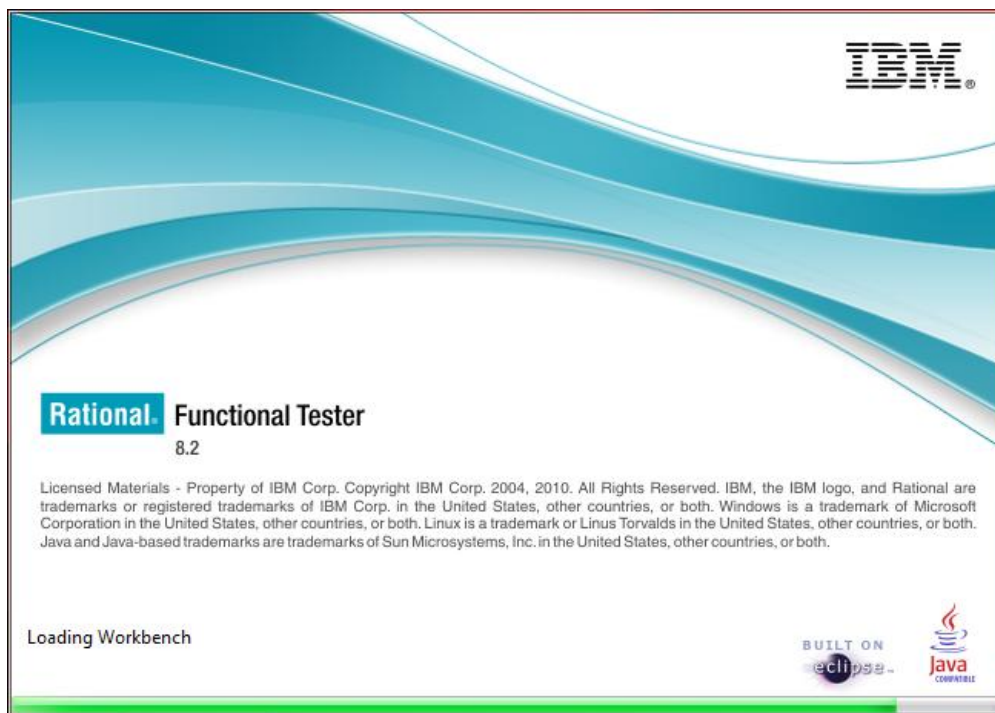


IBM Rational Functional Tester 8.2



Alat Functional Tester je dostupan u dva integrisana razvojna okruženja i dva skript jezika. Functional Tester Java Scripting koristi Java programski jezik i IBM Rational platformu. Functional Tester VB.NET Scripting koristi VB.NET jezik i Microsoft Visual Studio .NET razvojno okruženje.

Ovaj alat se koristi za:

- Izvršavanje potpunog funkcionalnog testiranja. Snimanje i reprodukciju skriptova, kako se krećete kroz vašu aplikaciju i testiranje stanja objekata kroz verifikacione tačke.
- Kreiranje i ažuriranje jednostavnih i čitljivih objektno-orijentisanih test skriptova. Pored snimanja automatskih test skriptova, Functional Tester sadrži wizard za generisanje koda, na primer za automatsko kreiranje verifikacionih tačaka. Test skripte se implementiraju po izboru testera, u Javi ili VB.NET-u.

Objektno-orijentisana tehnologija koristi se u Functional Tester-u za generisanje skripti brzim snimanjem funkcionalnosti aplikacije koju testiramo. Alat koristi objektno orijentisane tehnologije da bi identifikovao objekte njihovim unutrašnjim svojstvima, a ne koordinatama ekrana. Ukoliko se lokacija ili tekst nekog objekta promene, Functional Tester može to da pronađe pri reprodukciji. Tehnologija testiranja objekata u ovom alatu nam omogućava da testiramo bilo koji objekat u aplikaciji pod testom, uključujući i svojstva objekata i podatke. Možemo testirati objekte u Javi, VB.NET, Windows-u ili veb bazirane aplikacije, bez obzira da li su vidljive ili skrivene u interfejsu.

Kada snimate skript, Functional Tester automatski kreira mapu test objekata za aplikaciju koju testiramo. Mapa test objekata navodi test objekte dostupne u aplikaciji, bez obzira da li se trenutno prikazuju ili ne. Takođe, možete da kreirate novu mapu test objekata, bilo da se zasniva na postojećoj mapi ili dodavanjem novih objekata po potrebi. Ova mapa obezbeđuje da se na brz način

dodaju objekti u skript. Kako mapa test objekata sadrži prepoznata svojstva za svaki objekat, možete lako ažurirati takve informacije na jednoj lokaciji. Sve skripte koje koriste tu mapu test objekata dele ažurirane informacije.

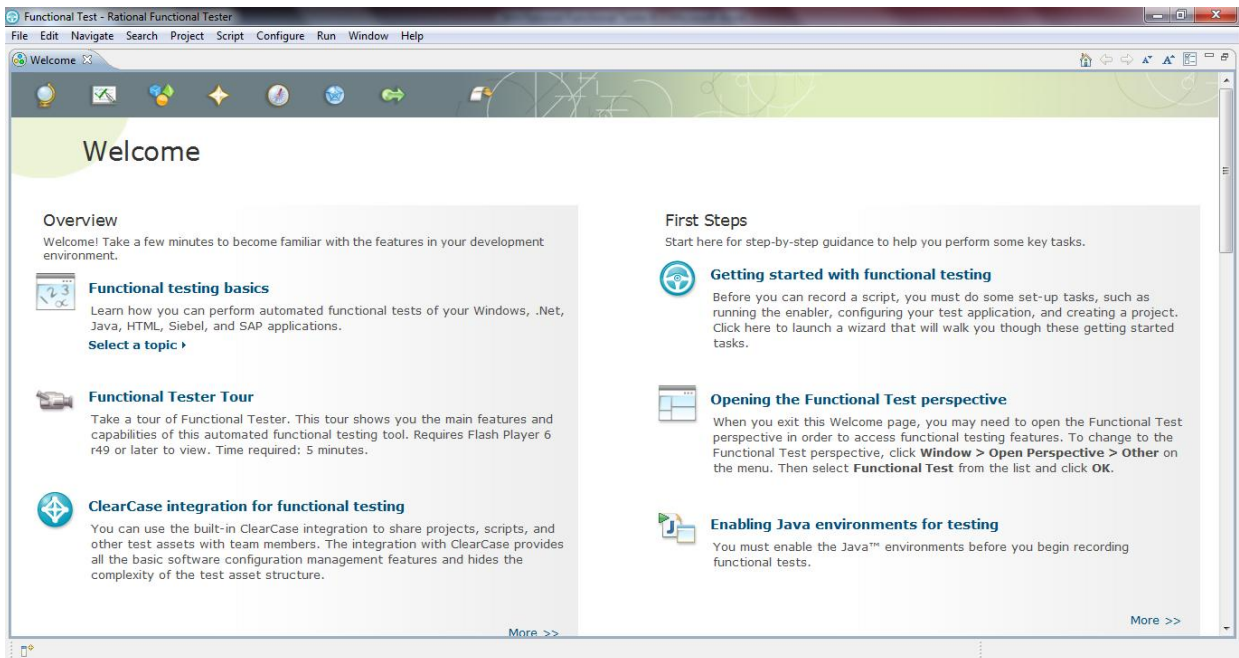
Tokom snimanja skripta, tester može dodati verifikacione tačke, koje služe da potvrde stanje objekata u aplikaciji koja se testira. Verifikaciona tačka snima informacije objekta (bazirane na tipu verifikacione tačke) i skladište ih u podacima u osnovnom fajlu. Functional Tester ima verifikacionu tačku za svojstva i 5 verifikacionih tačaka za podatke (meni, tabela, tekst, stablo i lista). Tester može koristiti VP Comparator da analizira razlike između izgenerisanog (ono šta treba da se dobije) i modifikovanog (ono što je dobijeno testiranjem) osnovnog fajla.

Ovaj alat ima platformsku nezavisnost i nezavisnost u izboru veb pregledača. Na primer, možete snimati skript na operativnom sistemu Windows, a da ga pokrećete na Linux-u. Takođe moguće je snimiti skript korišćenjem Mozilla Firefox-a, Internet Explorer-a ili Netscape-a. Iz razloga što skript ne sadrži referencu na veb pregledač tokom snimanja, moguće je reprodukovati skriptu koristeći neku od podržanih verzija Mozilla Firefox-a, Internet Explorer-a ili Netscape-a.

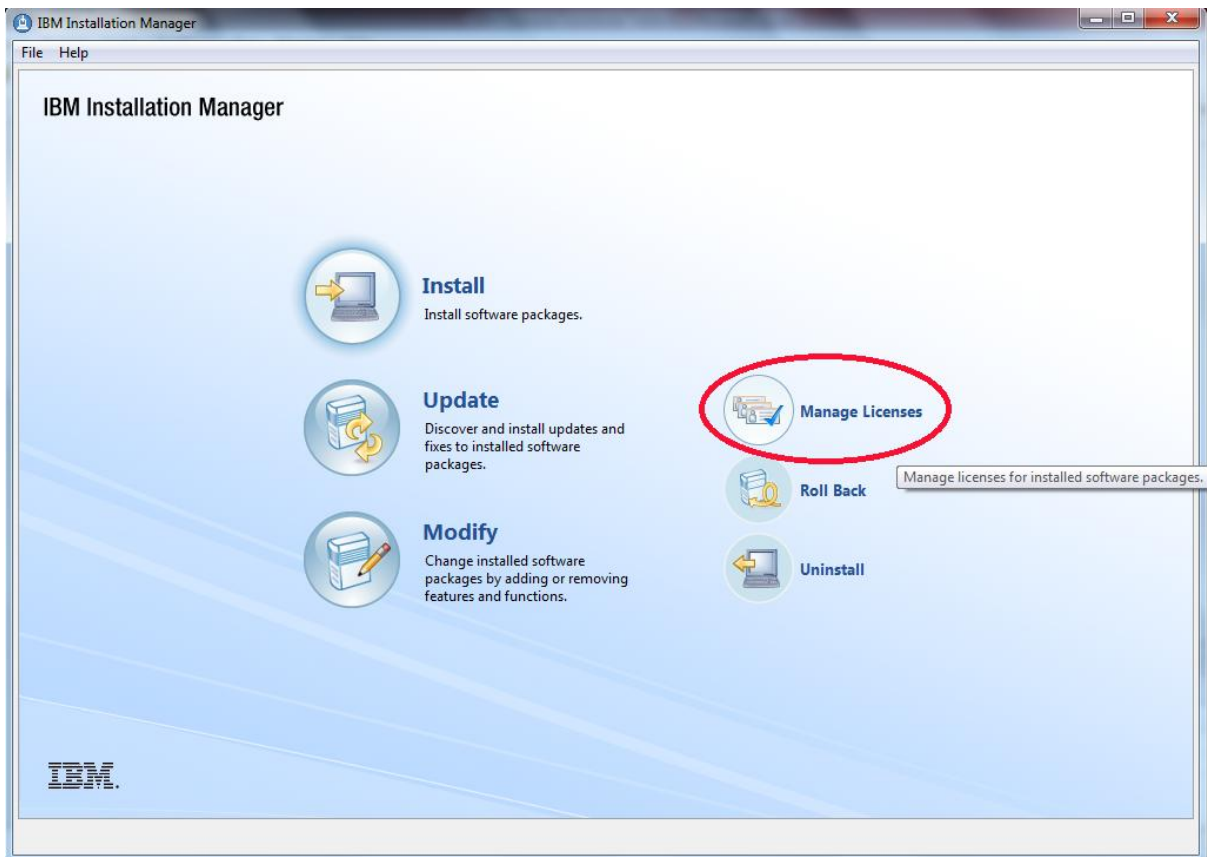
Functional Tester je integrisan sa Rational Quality Test Manager-om, koji nam omogućava da snimamo i reprodukujemo skript ili da pogledamo log iz alata Quality Manager. Naravno, ove integrisane funkcije mogu da se koriste, samo ukoliko imate instaliran Quality Manager.

Podržane aplikacije za funkcionalno testiranje:

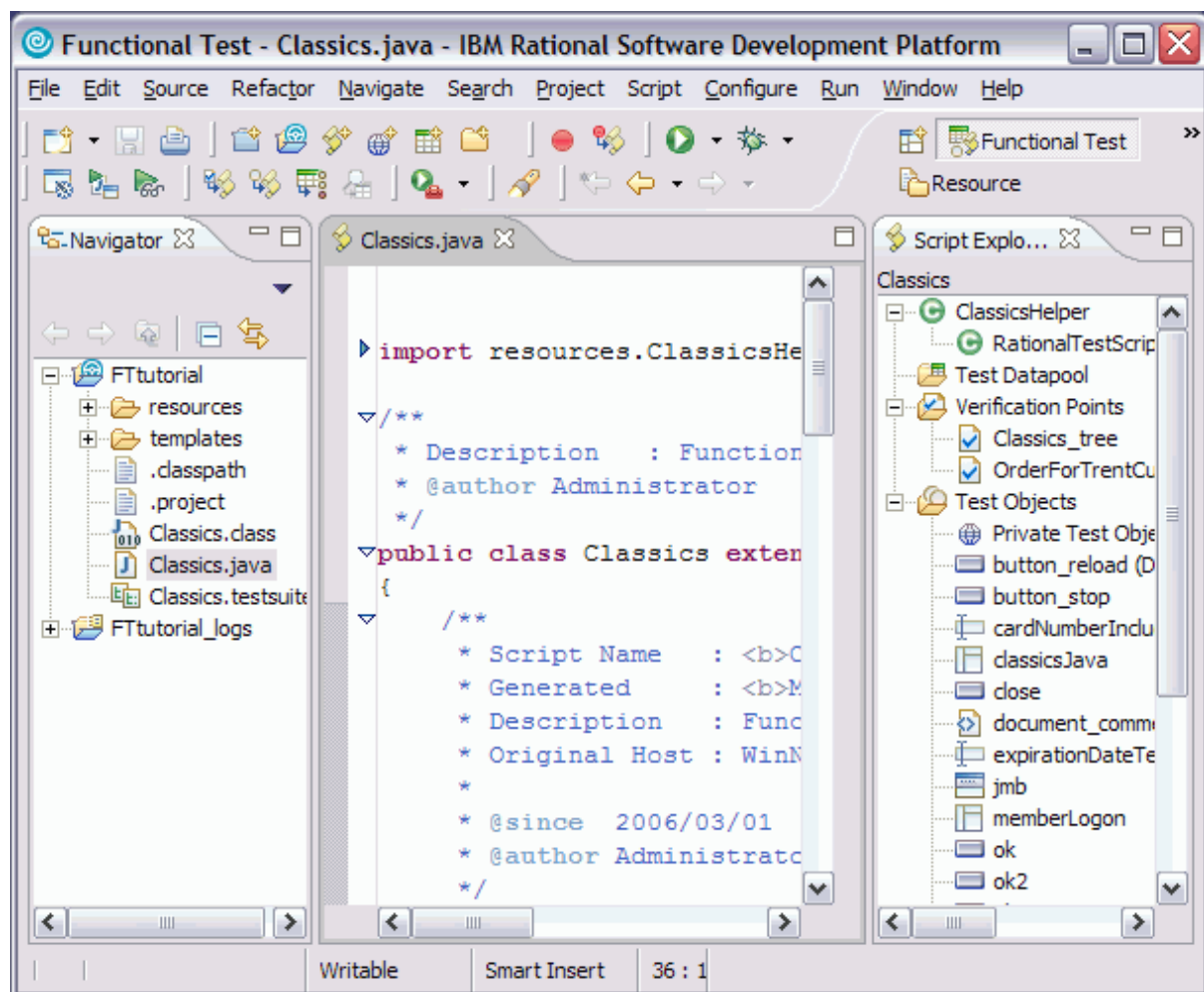
HTML	HTML aplikacije
Java	Java aplikacije
Windows	Windows aplikacije sa Win32 kontrolama i MFC kontrolama
.NET	.NET aplikacije i Win32 aplikacije koje sadrže . NET kontrole
Visual Basic	Visual Basic aplikacije
Silverlight	Silverlight aplikacije
Eclipse	Eclipse aplikacije i stand-alone SWT aplikacije
SAP	SAP GUI za Windows, SAP GUI za HTML i aplikacije koje su razvijane koristeći SAP WebDynPro framework
Siebel	Siebel aplikacije
Flex	Flex aplikacije
AJAX	Ajax bazirane veb aplikacije
Dojo	HTML aplikacije sa Dojo kontrolama
GEF apps	GEF aplikacije
PowerBuilder	.NET aplikacije i Win32 aplikacije koje su kreirane koristeći Sybase PowerBuilder
PDF	Adobe PDF dokumenta
zSeries	zSeries aplikacije
iSeries	iSeries aplikacije
pSeries	pSeries aplikacije



Učitavanje licence radi se iz alata *IBM Installation Manager*. Klikne se na dugme **Manage Licenses**, nakon toga čekirati **Import product Activation Kit** i kliknuti na Next. Zatim pronaći JAR fajl koji predstavlja licencu i učitati ga. U alatu će se zatim pojaviti ta učitana licenca i datum trajanja licence.



Osnovne informacije o alatu



Functional Tester je alat koji se prikazuje kao klasičan Java editor, jer u osnovi predstavlja nadogradnju (plugin) za alat Eclipse. Sa leve strane nalazi se prozor za navigaciju kroz projekte i fajlove (Functional Test Projects view). Svaki projekat ima svoj naziv i sastoji se od skriptova. Kada dva puta kliknete na određeni skript u tom prozoru, taj skript će postati aktivan i otvoriće se u centralnom prozoru. Skript se može prikazati i kao Java kod, tako što se na dnu centralnog prozora aktivira tab Java, umesto Script.

Sa desne strane nalazi se prozor Script Explorer, u kome se prikazuju sve verifikacione tačke i objekti koji su mapirani u trenutno aktivnom skriptu. Iz ovog prozora moguće je pokrenuti Verification Point Editor i ažurirati verifikacione tačke ili startovati Object Map Editor i ažurirati objekti.

Podešavanje alata Rational Functional Tester

IBM obezbeđuje JRE (Java Runtime Environment), koji se instalira uz alat. Kada želite da testirate svoje Java ili HTML aplikacije, pokrenite i podesite okruženja i aplikacije.

IBM Rational Functional Tester podržava nekoliko varijanti za prikazivanje loga (izveštaja). Mi ćemo koristiti HTML log. Koraci za podešavanje su:

- 1) Kliknite na **Window > Open Perspective > Other** da biste otvorili perspektivu za funkcionalno testiranje.
- 2) U Open Perspective dijalogu, odaberite **Functional Test** opciju.
- 3) Da proverite da li su podešeni HTML izveštaji kliknite na **Window > Preferences**.
- 4) U levom panelu prozora Preferences, izaberite **Functional Test > Playback** i kliknite na **Logging**.
- 5) Odaberite **html** kao tip izveštaja i kliknite na **OK**.

Pre nego što počnemo da snimamo test skript, kreiraćemo novi projekat:


- 1) Kliknite na **File > New > Functional Test Project**.
- 2) U **Project name** stavite Primer1.
- 3) U **Project location** stavite: C:\Temp. Direktorijum će biti kreiran ukoliko ne postoji.
- 4) Nemojte izabrati **Add the project to Source Control** checkbox, ako je prikazan.
- 5) Nemojte izabrati **Associate the Functional Test Project with current Rational Project** checkbox, ako je prikazan.
- 6) Kliknite na **Finish**.

Nakon toga videćete taj projekat u *Project view*, u levom panelu Functional Test perspektive.

Takođe, moguće je generisati pojednostavljene skriptove i Java test skriptove. Sa pojednostavljenim skriptovima i dozvoljenim vizuelnim karakteristikama aplikacije, može da se pređe na Java skripte, ali ne i obrnuto. Zato je pre nego što počnete sa snimanjem skripte, potrebno uraditi sledeće:

- 1) Idite na **Window > Preferences**.
- 2) U levom panelu u Preferences prozoru kliknuti na **Simplified Scripting**.
- 3) Zatim izabrati **Enable Simplified Scripting**.
- 4) U podstavki Application Visuals, izabrati **Enable capturing of application visuals, Insert Data Driver Commands, Enable capturing of verification on test data**.
- 5) Izabrati **Apply** i kliknuti na **OK**.

Snimanje jednostavnog tekst skripta

- 1) Za početak snimanja skripte kliknuti na **Record a Functional Test Script** ikonicu () u Functional Test toolbar-u.
- 2) Odaberite Primer1 projekat koji ste kreirali.
- 3) U **Script name**, stavite Skript1. Ne treba odabrati opciju **Add script to Source Control**, ako je dostupna.
- 4) Kliknuti na **Next**.

Kada kreirate test skriptu, test datapool i drugi test assets su kreirani. Koristite podrazumevane vrednosti za Test Datapool: Private Test Datapool i za Datapool Record


Selection Order: Sequential. Privatni test datapool je povezan sa samo jednim skriptom i nije dostupan drugim skriptovima. Kada koristite sekvencijalni poredak, test scenario pristupa datapool zapisima u cilju da ih organizuje u datapool.

- 5) Kliknuti na **Finish**.

Alat Functional Tester će se automatski minimizovati i biće prikazan Recording Monitor (prozor za pauziranje ili stopiranje izvršavanja skripta).




Sada ćemo videti kako se testira. Otvorićemo *ClassicsJavaA* aplikaciju:

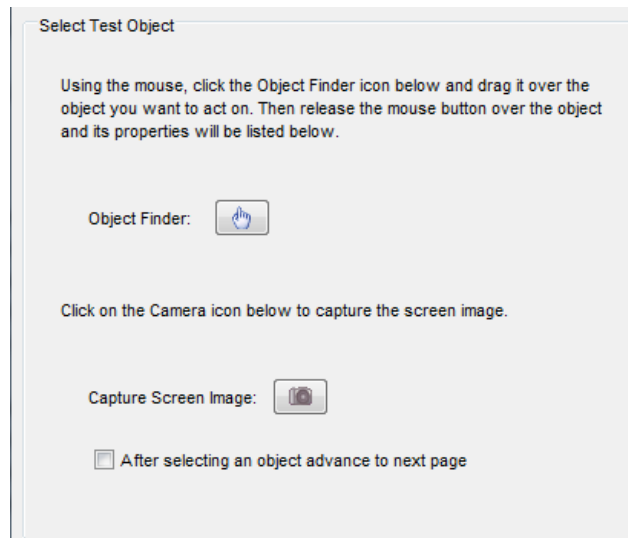
- 1) Da startujete aplikaciju, kliknite na ikonicu **Start Application** .
- 2) U prozoru, odabrati destinaciju ka JRE i kliknuti na **Run**.
- 3) Odabrati recimo: Bach u listi kompozitora, kliknuti na Violin Concertos, zatim ići na tab sa detaljima (Details). Kliknuti na Place Order. U prozoru za logovanje korisnika, ostavite postojećeg kupca (Trent Culpito), a u polje za šifru unesite xxxx.
- 4) Kliknuti na OK.
- 5) U polju za račun, staviti za broj kartice na primer 1234 1234 1234 1234, i 12/12 u polje za istek važenja kartice.

Kreiranje verifikacione tačke

Svaki test ima jedno ili više mesta na kojima se vrši verifikacija objekata koji se testiraju. Verifikaciona tačka služi da proveriti da li se određena akcija izvršila ili da proveriti stanje objekta. Može se kreirati verifikaciona tačka osobina, verifikaciona tačka slika ili devet tipova verifikacije podataka. Kada kreiramo verifikacionu tačku, snimamo informaciju o stanju nekog objekta u aplikaciji koja će biti upoređena sa stanjem tog objekta prilikom reprodukcije skripta.

Sada ćemo pokazati kako se snima verifikaciona tačka podataka:

- 1) U Recording Monitor-u, kliknuti na **Insert Verification Point** ili **Action Command button** .
- 2) U prozoru za selektovanje objekta testiranja, nije potrebno čekirati opciju *After selecting an object advance to next page*, ako je ta opcija čekirana.



- 3) Koristićemo Object Finder (👉) da odaberemo stablo sa kompozitorima u aplikaciji koju testiramo. Kliknuti i držati Object Finder i prevući kursor miša na objekat koji želimo. Kada kursor miša dovedemo do stabla, videćemo crveni okvir oko objekta i biće prikazan naziv objekta (`javax.swing.JTree`). Kada se završi dohvatanje objekta, pustiti dugme miša i podaci o objektu će biti prikazani na dnu prozora Select Test Object.
- 4) Kliknuti na Next.
- 5) U prozoru Select an Action page, proverite da li je izabrano Perform Data Verification Point i kliknuti na Next.
- 6) U Data Value polju kod Insert Verification Point Data Command, odabрати Tree Hierarchy test. Tada će biti zabeležene informacije o celom stablu koje smo odabrali.
- 7) U polju za ime verifikacione tačke upisati naziv (`Classics_tree`) i kliknuti Next.
- 8) Verification Point Data page prikazuje sve podatke koji se čuvaju. Ako se kvačica pojavljuje pored stavke, ta stavka će biti testirana. Podrazumevano je da su sve stavke čekirane. Ako nisu izabrane, možete kliknuti na Check All dugme.
- 9) Kliknuti na Finish.

Verifikaciona tačka može biti i slika.

- 1) U Recording Monitor-u, kliknuti na **Insert Verification Point** ili **Action Command button** (🔗)
- 2) U prozoru za selektovanje objekta testiranja, nije potrebno čekirati opciju *After selecting an object advance to next page*, ako je ta opcija čekirana.
- 3) Koristićemo Object Finder (👉) da odaberemo sliku nekog albuma u aplikaciji koju testiramo. Kliknuti i držati Object Finder i prevući kursor miša na neku sliku. Kada kursor miša dovedemo do slike, videćemo crveni okvir oko objekta i biće prikazan naziv objekta (`javax.swing.JLabel`). Kada se završi dohvatanje objekta, pustiti dugme miša i podaci o objektu će biti prikazani na dnu prozora Select Test Object.
- 4) Kliknuti na Next.
- 5) Odabрати Perform Image Verification Point i kliknuti na Next.
- 6) Uneti ime verifikacione tačke (npr. `Album_image`).
- 7) Proveriti da li je odabrana opcija Select full image.
- 8) Verifikaciona tačka, odnosno slika koju hvatamo na ekranu, biće prikazana u desnom prozoru. Kliknuti na Finish.

Sada ćemo prikazati korišćenje verifikacione tačke svojstva. Proveravaćemo ispravnost određenog teksta.

- 1) U aplikaciji ClassicCD, kliknuti na Order > View Existing Order Status. Nemojte kliknuti na polje za lozinku.
- 2) Kliknuti na OK. Testiraćemo labelu "Order for Trent Culpito" u prozoru koji smo dobili.
- 3) U Recording Monitor-u, kliknuti na **Insert Verification Point** ili **Action Command button** (🔧).
- 4) U prozoru za selektovanje objekta testiranja, čekirajte opciju *After selecting an object advance to next page*.
- 5) Koristićemo Object Finder (👉) da odaberemo labelu „Order for Trent Culpito“. Kliknuti i držati Object Finder i prevući kursor miša do labele. Kada kursor miša dovedemo do labele, videćemo crveni okvir oko objekta i biće prikazan naziv objekta (`javax.swing.JLabel`). Kada se završi dohvatanje objekta, pustiti dugme miša i podaci o objektu će biti prikazani na dnu prozora Select Test Object.
- 6) Odabrati drugu opciju Perform a Properties Verification Point i kliknuti na Next.
- 7) Potvrditi da je IncludeChildren polje postavljeno na None.
- 8) U polju za naziv verifikacione tačke prihvatiti postavljenu vrednost.
- 9) Ostaviti čekiranu opciju „Use standard properties“. Svojstva testiranog objekta i vrednosti biće prikazane u tabeli. Za svaku stavku može se odabrati da li želimo da je testiramo ili ne. Takođe, moguće je editovati vrednosti svojstava u koloni sa vrednostima.
- 10) U koloni sa svojstvima izabrati tekst i vidljiva svojstva koja će biti testirana tokom reprodukcije. Možda ćete morati da kliknete na checkbox dva puta da biste izabrali opciju.
- 11) Kliknuti na Finish.
- 12) Zatvoriti prozor „View Existing Orders“.

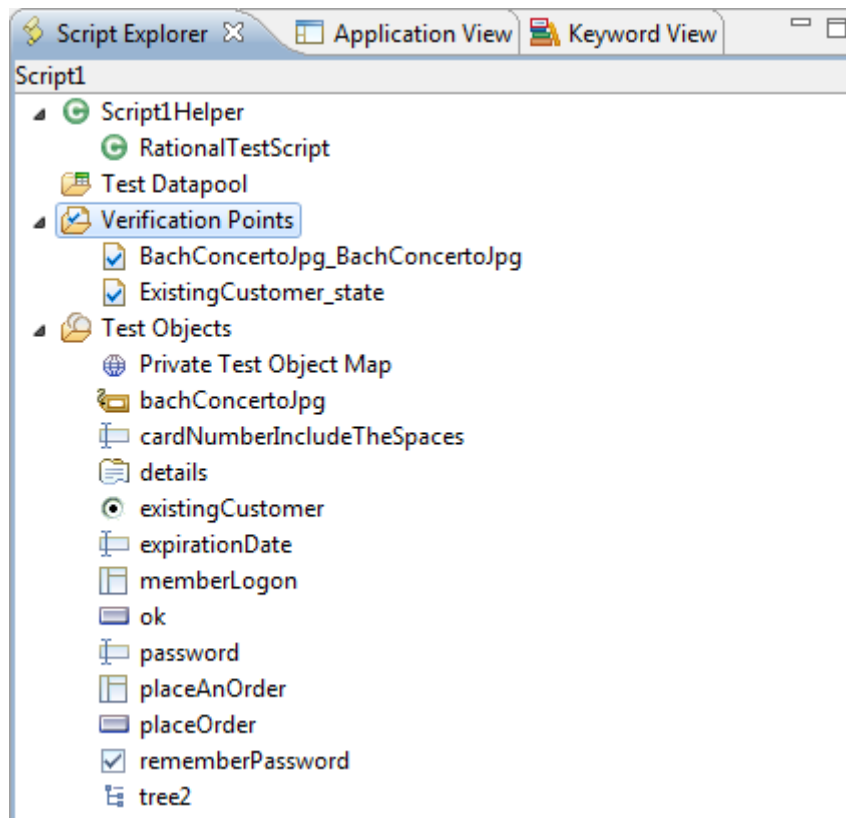
Pokretanje skripta

Snimljene skriptove možemo pokrenuti i proveriti ispravnost njihovog izvršavanja.

- 1) Da bismo pokrenuli skript, potrebno je da kliknemo na dugme za Playback (▶) u glavnom toolbaru.
- 2) Alat će vas pitati kako želite da se zove naziv log fajla koji će se kreirati i koji će predstavljati izveštaj izvršenog test skripta. Nakon toga će se Rational Functional Tester minimizovati i pokrenuće se Playback Monitor. Dok se skript izvršava, poruke će biti prikazane u Playback Monitor-u. Sve akcije koje su zabeležene u test skriptu biće prikazane prilikom izvršavanja (reprodukcije) skripta, čak i verifikacione tačke. Kada se reprodukcija završi, biće prikazan rezultat izvršavanja pokrenutog testa, kao HTML log u posebnom prozoru. Svaka akcija biće prikazana u izveštaju, a on će obuhvatiti i verifikacione tačke. Kod verifikacionih tački treba da piše Pass, što označava da su te tačke uspešno prošle izvršavanje i označene su zelenim fontom. Takođe, testovi koji uspešno prođu izvršavanje svih koraka, postaju uspešni testovi sa oznakom Pass.
- 3) Zatvoriti log.

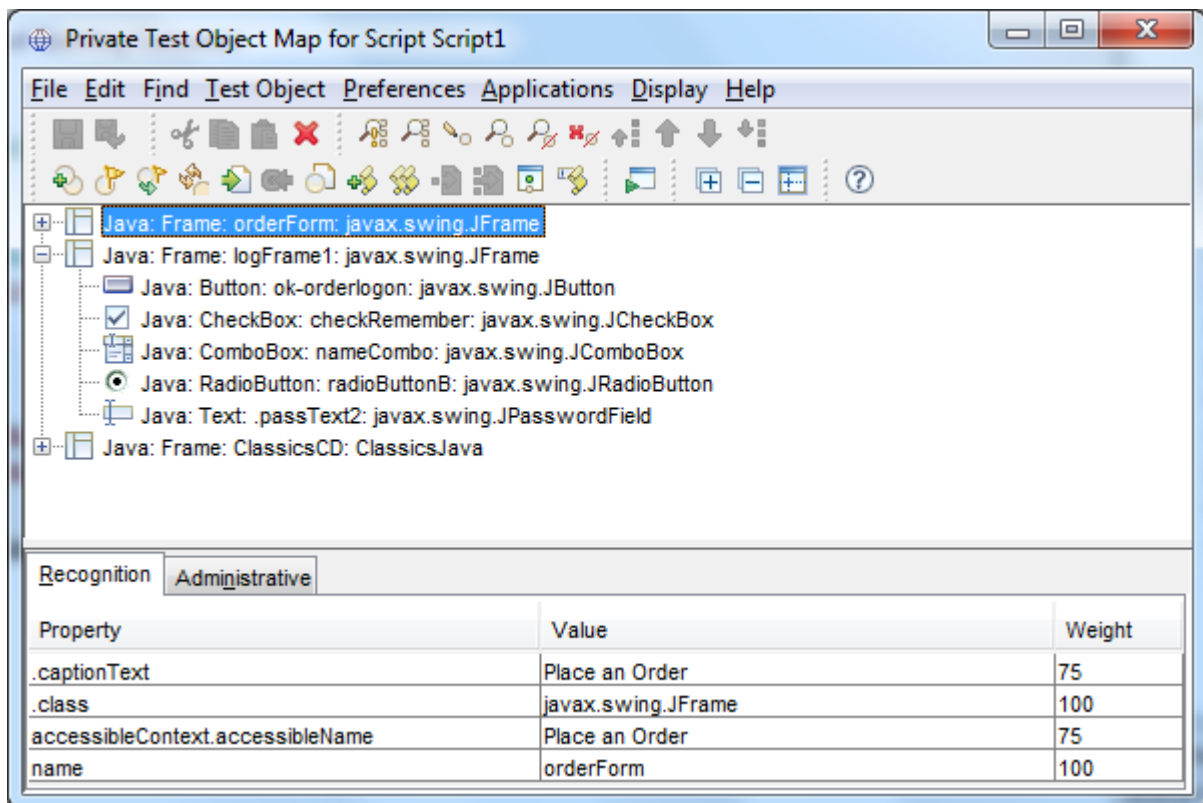
Pregled verifikacionih tačaka

Unutar verifikacione tačke moguće je izmeniti podatke. Verifikacione tačke pokazuju se u prozoru “Script Explorer” u podstablu Verification Points. Kada dva puta kliknete na verifikacionu tačku, otvara se Verification Point editor, gde se može ažurirati verifikaciona tačka za sledeća izvršavanja tog test primera.



Takođe možemo pregledati i izmeniti podatke unutar mape objekata. U prozoru Script Explorer otvoriti podstablo Test Objects. U prvoj stavki, Private Test Object Map, prikazane su sve reference na objekte koji su bili aktivni tokom snimanja.

- 1) Kliknuti na Private Test Object Map () da biste ih otvorili.



Otvoriti sledeću stavku: `Java: Frame: logFrame1: javax.swing.JFrame`.

Objekat za ovaj frejm obuhvata dugme, checkbox, combobox, radio dugme i tekstualno polje za unos lozinke. Kliknuti na neki objekat da biste videli podatke o vrednostima koje su dodeljene svojstvima tog objekta i eventualno ih menjati. Ovde možete na lak način da unesete nove objekte (kliknuti desnim dugmetom miša i odabrati `Test Object > Insert Object`).

Izvršavanje regresivnih testova

Možemo sada da probamo da izvršimo test skript na drugoj aplikaciji. Potrebno je da promenimo naziv aplikacije koju testiramo u samom skriptu, komandom: `startApp(„ClassicsJavaB“);`

- 1) Kliknimo na pokretanje skripta (▶).
- 2) Alat će nas pitati da li želimo da prepíšemo sadržaj postojećeg loga. Kliknuti na Yes. Skript će krenuti brzo izvršavanje, jer neki delovi aplikacije `ClassicsJavaA` su isti kao kod `B`, dok se negde razlikuju (na primer: tekst pored checkboxa).
- 3) Nakon reprodukcije će se otvoriti log sa porukama, gde će biti eventualna upozorenja i greške koje su nastale izvršavanjem skripta.

Korišćenje komparacije prilikom ažuriranja verifikacione tačke

Nakon što izvršimo puštanje skripta, možemo koristiti Verification Point Comparator, kako bismo uporedili podatke o verifikacionoj tački, koja je izvršena i koja je snimljena u skriptu. Verifikaciona tačka pruža osnovna svojstva ili podatke o nekom objektu na kome radimo verifikaciju. Ako izvršavanje verifikacione tačke ne uspe, to znači da smo ili našli defekat (bug) ili je izvršena namerna promena u aplikaciji. Ako je promena namerno izvršena, možemo da ažuriramo informacije u trenutku verifikacije, tako da test nastavlja da važi za buduće verzije.

- 1) U logu kliknite na View Results link na kraju verifikacione tačke, gde je utvrđena greška (FAIL).

Verifikaciona tačka nije prošla, jer je uz pomoć komparacije utvrđeno da su vrednosti drugačije. Na primer ukoliko hvatamo sliku u aplikaciji ClassicsA, i istu skriptu pokrenemo u aplikaciji ClassicsB, može da se desi da su visina i širina te slike različite.

- 2) Kliknuti na Load Baseline to Edit button (🔧) u toolbaru za komparaciju.
- 3) Kliknuti na Replace Baseline with actual value button (🔄) i učitajte se podaci o aktuelnoj slici.
- 4) Zatvoriti Comparator.
- 5) Za upoređivanje razlika može se koristiti i dugme predviđeno za to – Jump to First Difference button (🔍).

Dodavanje novog objekta u mapu

- 1) Kliknuti na Applications > Run u meniju za mapu objekata.
- 2) Izabrati ClassicsJavaB.
- 3) Kliknuti na OK.
- 4) U aplikaciji izabrati određeni CD i kliknuti na Place Order, tako da se otvori prozor za logovanje korisnika aplikacije.
- 5) Kliknuti na Test Object > Insert Object(s) u meniju.
- 6) Ako je selektovano, isključiti opciju After selecting an object advance to next page.
- 7) Koristiti Object Finder alat da izaberete checkbox “Remember the Password”.
- 8) Kliknuti na Next.
- 9) Kliknuti na Finish.
- 10) Novi checkbox objekat će sada biti prikazan u mapi objekata.

Promena podešavanja za prepoznavanje

- 1) U glavnom meniju kliknuti na Window > Preferences.
- 2) Kliknuti na Functional Test > Playback > ScriptAssure.
- 3) Kliknuti na dugme Advanced.
Primitite da u podrazumevanim podešavanjima stoji da je kriterijum prihvatljivosti veći od 10000, što ukazuje da jedno važno svojstvo može biti pogrešno. Promenite taj skor na 5000, pa vidite šta se dešava.
- 4) Izaberite Use Default checkbox pored tog polja.
- 5) Upisati 4000 u polje i kliknuti OK, pa ponoviti izvršavanje skripta nad aplikacijom ClassicsB.

- 6) Nakon utvrđivanja razlika, vratiti podešavanja na stara. Ovom izmenom, cilj je da skripta i pored malih promena u aplikaciji ClassicsB prolazi uspešno, što ranije nije bio slučaj.

Korišćenje regularnih izraza

Regularne izraze koristićemo da ispitamo vrednost nekog svojstva.

- 1) Kliknuti desnim dugmetom miša na checkRemember vrednost kod lozinke i kliknuti na Convert Value to Regular Expression. Alat Functional Tester će označiti vrednost koju proglasite za regularni izraz dodavanjem ikonice „xy“ ispred vrednosti teksta.
- 2) Dva puta kliknuti na polje za vrednost kako biste ga ažurirali.
- 3) Obrišati reč check i postaviti regularni izraz `[rR]ember`
Kliknuti van ćelije. Ovaj uzorak omogućava da se koristi reč „remember“ sa malim ili velikim početnim slovom.
- 4) Desnim dugmetom miša kliknuti na Remember Password vrednost i izabrati Convert Value to Regular Expression, da biste uradili konverziju.
- 5) Dva puta kliknuti na vrednost i editovati: `Remember.*Password`
Tačka će ovde označavati bilo koji karakter na toj poziciji (na primer u nekoj verziji može na tom mestu biti blanko znak, a u nekoj drugoj verziji može biti više blanko znakova).
- 6) Kliknuti na Next, pa na Finish.
- 7) Izabrati File > Save u mapi objekata, kako biste snimili izmene. Nakon toga će regularni izrazi biti proveravani prilikom ponovnog izvršavanja skripta.