

ETF SARAJEVO

ODSJEK ZA R & I

SPECIJALNA POGLAVLJA SOFTWARESKIH SISTEMA

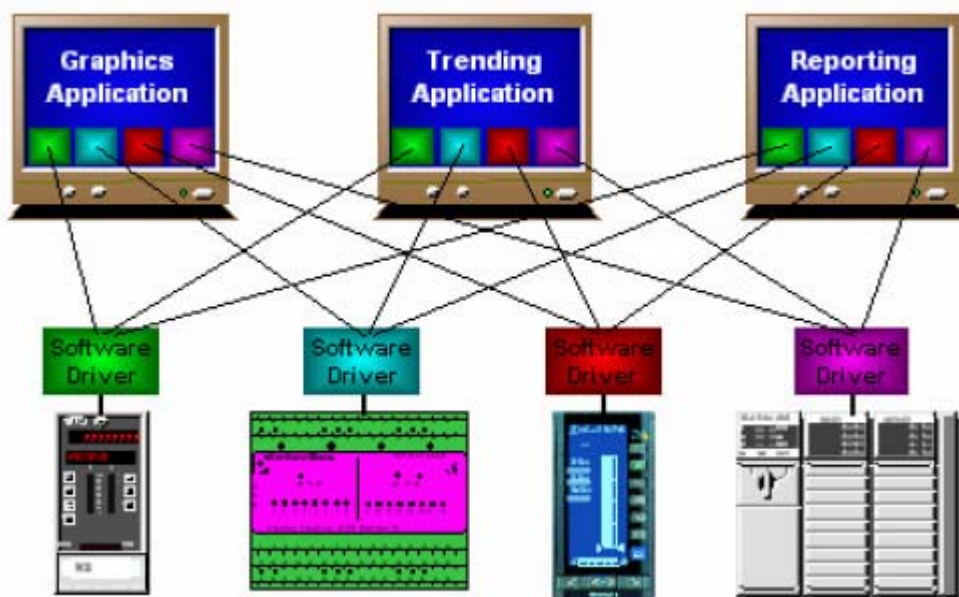
Sarajevo, Avgust 2004

KARAKTERISTIKE GENESIS32 HMI PAKETA ICONICSA

Da bi se razumjela arhitektura i softwareska organizacija Iconicsovog Genesis32 HMI softwareskog paketa, potrebno je na početku dati kratki pregled koncepta na kojem je cijela ova arhitektura izgradjena, a to je Microsoft-ov koncept razmjene podataka izmedju Windows aplikacija (OLE – object linking and embedding) , i na njemu izgradjen OPC – **O**LE for **P**rocess **C**ontrol koncept .

OPC je standardizirani pristup za povezivanje izvora podataka (naprimjejr. PLC – programmable logic controllers), digitalnih regulatora i kontrolera, I/O (ulazno/izlaznih) uredjaja, baza podataka itd., sa HMI (Human machine interfejs , interfejsom čovjek – mašina), klijentskim aplikacijama (grafika, trendiranje, alarmiranje, itd.). Taj standard unapredjuje interfejs izmedju klijent i server aplikacija obebjedjujući univerzalno podržavan i dobro dokumentovan mehanizam, da komunicira podatke iz izvora podataka ka klijent aplikaciji. Uključen je ne samo detaljni opis o tome kako prenjeti podatke, nego i specifične informacije o drugim atributima koji su dopunski uz ove podatke, kao što su informacije o opsezima podataka (data range), tipu podataka, flagovima kvaliteta, kao i informacijama o datumu i vremenu kada je podatak očitao sa izvora.

Da bi se bolje razumjela razlika izmedju stare/tradicionalne klijent-server arhitekture i OPC bazirane, posmatrajmo ilustraciju na slijedećoj slici:



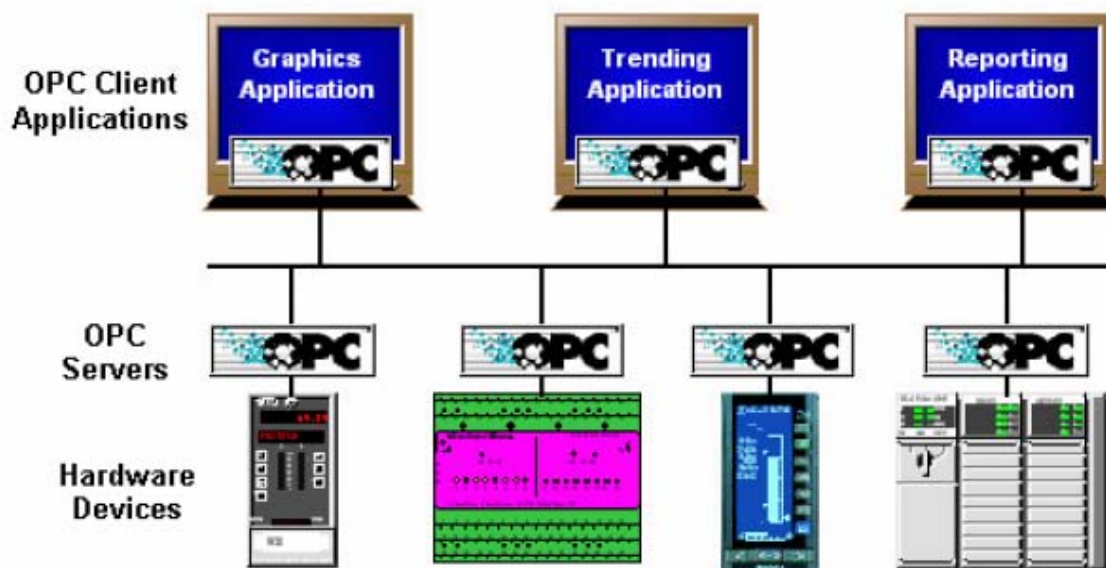
Stara/tradicionalna klijent-server arhitektura

Kod ove arhitekture, od svakog klijenta se zahtjevalo da napiše svoj vlastiti set servera za uređaje (svaka linija na prethodnoj slici je poseban softwareski interfejsni drajver). Ovo je bilo uzrokovano činjenicom da, mada se protokol uređaja ili servera ne mora promijeniti, arhitekture različitih klijent aplikacija (od različitih vendora), su različite.

Ovo je vodilo ka povećanom opterećenju na mašinu, dodatnim aktivnostima na postavljanju svakog drajvera, povećanju troškova projekta, i vremena potrebnog za implementaciju rješenja.

Ako se uređaj ili protokol servera promjenio ili ažurirao, tada svaka od klijentskih aplikacija se morala takodjer promijeniti.

Naredna slika pokazuje OPC arhitekturu, koja je uvedena od strane OPC Fondacije, uglavnom da se prevaziđe ovaj problem.



OPC bazirana klijent-server arhitektura

Slijedeći OPC arhitekturu, uređaj treba samo da ima jedan standardni drajver, koji je OPC server koji je usaglašen sa standardom (OPC compliant). OPC usaglašena klijent aplikacija se sada može povezati sa uređajem, bilo lokalno ili preko mreže. Nadalje, konekcije se mogu ostvariti sa više servera istovremeno.

Bilo koja OPC klijent aplikacija se može povezati sa svakim OPC serverom. Drugim riječima, OPC nudi istinsku plug-and-play mogućnost, u oblastima HMI i industrijske automatizacije i integracije procesnih i poslovnih baza podataka.

Tipovi OPC servera mogu biti:

- OPC Data Access (DA) – serveri za pristup podacima
- OPC Alarms and Events (AE) – serveri za alarme i događaje
- OPC Historical Data Access (HDA) - serveri za pristup historijskim podacima

KARAKTERISTIKE GENESIS32 ENTERPRISE IZDANJA

ICONICS kao proizvođač HMI softwareskog paketa GENESIS32, nudi slijedeće samostalne (standalone) OPC klijent aplikacije, za grafike, skupljanje i trendiranje podataka i alarmiranja:

- **GraphWorx32**
- **TrendWorx32**
- **AlarmWorx32**

Svaka klijent aplikacija je ActiveX kontejner koji je baziran na internet/intranet spremnoj objektnoj tehnologiji i takodjer nudi Microsoftovu moćnu Visual Basic za aplikacije (Visual Basic for Applications-VBA) tehnologiju.

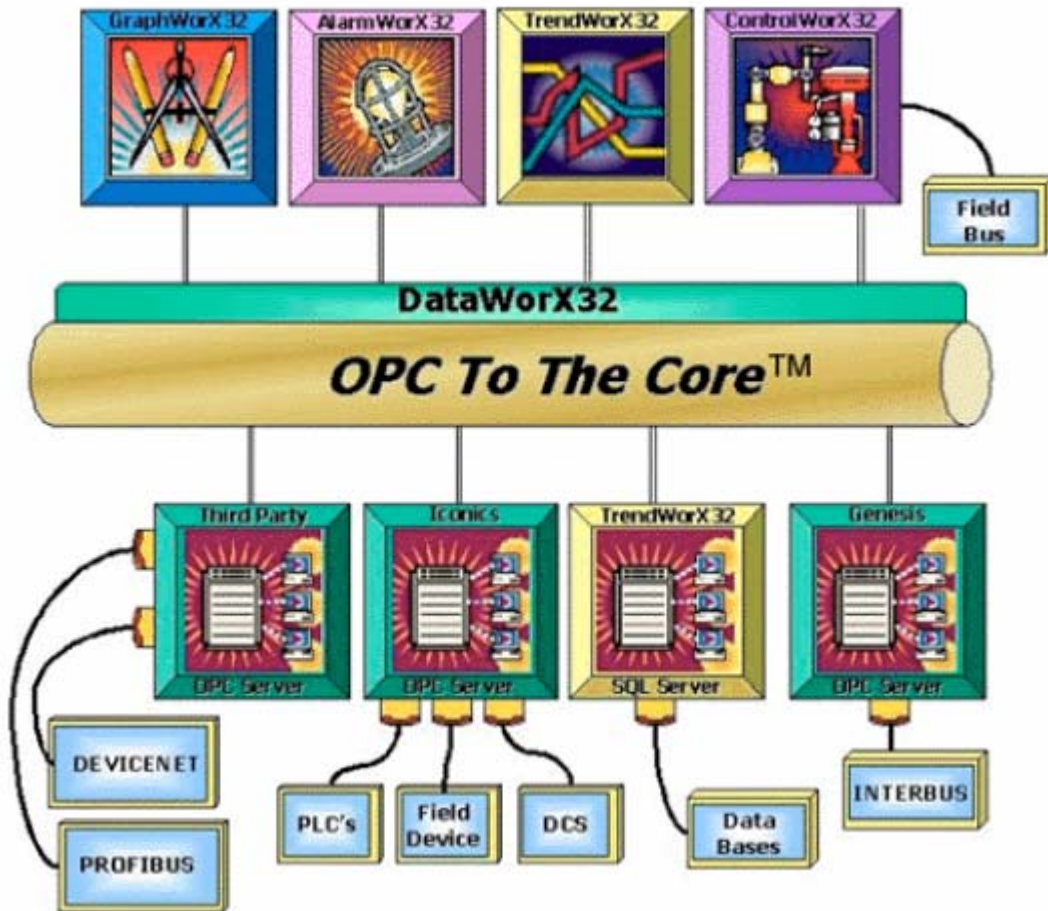
GENESIS32 je skup ovih softwareskih komponenti koji obezbjedjuje kompletno rješenje za HMI/SCADA softwareske aplikacije. Osim gore pomenuta tri glavna modula, slijedeće aplikacije su takodjer uključene:

- ScriptWorX32
- ProjectWorX32
- Screen Manager
- DataSpay
- GenBroker
- Dr.DCOM
- Security Server

ICONICS takodjer nudi niz drugih softwareskih modula da kompletira GENESIS32 Enterprise Edition. Slijedeći moduli su dodatne softwareske opcije:

- WebHMI
- DataWorX32
- AlarmWorx32 Multimedia
- ControlWorX32
- MobileHMI
- ReportWorX
- Symbol Library
- ActiveX ToolBox
- ICONICS OPC Servers
- OPC ToolWorx
- ActiveX ToolWorX

Naredna slika pokazuje kako svaki od ovih softwareskih modula se uklapa u OPC Automation arhitekturu.



Slijedi kratki pregled ovih gore pobrojanih modula:

GraphWorX32 – je samostalna aplikacija koja predstavlja OPC klijent softwareski paket za HMI , nudeći bogat skup alata za kreiranje animiranih grafika. Ovaj paket je onaj koji se koristi za vizuelizaciju procesa i za interfejs Operatora sa procesom kojeg nadzire i vodi. GraphWorX32 kombinuje crteže, operativne procedure, izvještavanje o kvalitetu i management direktno u procesne displeje koristeći bogati izbor ActiveX kontrolnih elemenata i VBA (Visual Basic za aplikacije) skript jezik.

Osnovne karakteristike su:

- istinski 32 bitni , više-končani (multi-threaded), multi platformski dizajn
- Univerzalna OPC konektivnost sa OPC data serverima
- korištenje Visual Basica za aplikacije (VBA)
- objektno orijentirana grafika
- moćni alati za kreiranje displeja i animacioni alati za njih
- displeji sa fiksnom i skalabilnom veličinom
- uranjanje (embedding) ActiveX kontrola i OLE objekata
- dinamička animacija sa ažuriranjem do 50 msec
- Web-friendly: može biti embediran u VB i HTML stranice

- ActiveX dokument server. Može se gledati korištenjem Web browsera kao što su MS IE i Netscape Navigator.
- Napredna biblioteka simbola
- TrendWorX32 i AlarmWorx32 objekti se mogu uroniti (embedd) unutar GraphWorx32 obezbjedjujući na taj način cjelovitiji Operatorski interfejs.
- Ugradjeni (built-in) izrazi i kalkulacije
- Lokalno i globalno aliasiranje
- Slojni ekrani kao kod AUTOCAD-a
- Grafičke matrice (templates) koje štede vrijeme
- Prelazak sa jednog interfejsnog jezika na drugi

TrendWorX32 je takodjer samostalni OPC klijent softwareski paket za prikupljanje podataka u realnom vremenu, pohranjivanje (logging) podataka u bazu podataka, trendiranje i real-time i historijskih podataka, i obezbjedjivanje moćnog skupa alata za izvještavanje i analizu. TrendWorX32 nudi otvoreno rješenje za aplikacije koje zahtijevaju skalirane i distribuirane performanse u realnom vremenu, koristeći najnovije OPC HDA specifikacije za kreiranje Plug and Play historijskih data servera i klijenata.

Osnovne karakteristike su:

- istinski 32 bitni , više-končani (multi-threaded), multi platformski dizajn
- Univerzalna OPC konektivnost sa OPC data serverima
- Opslužuje logiranje podataka koristeći OPC Historical Data Access (OPC HDA).
- Koristi Visual basic za aplikacije (VBA) za skripting
- OPC Trending ActiveX , sa fleksibilnošću podržavanja vremenski baziranih trendova, XY plotova, logaritamskih trendova, bar grafova, strip čartova, i cirkularnih čart trend displeja.
- Runtime editiranje trend prozora.
- Višestruki nivoi zumiranja za detaljne analize
- Trend i log izrazi i kalkulacije
- Statističke informacije o trendiranim podacima
- Idealni batch trendovi
- Mješani podatci realnog vremena i historije u istom trend prikazu
- Logger se može izvršavati kao NT Servis
- Logira podatke koristeći ADO/OLEDB tehnologiju
- Podržava MS Access , MS SQL Server , Oracle i MSDE baze podataka
- Loguje batch informacije
- Filtrira i kompresira podatke koristeći niz raznih funkcija
- Pridružuje Operatorske komentare logiranim podacima
- Rasporedjuje historijske izvještaje
- Trigeruje izvještaje bazirane na stanjima podataka
- Publikuje izvještaje na Webu
- Lagana promjena interfejsnog jezika

AlarmWorX32 je također samostalni OPC klijent softwareski paket za distribuirano prikupljanje alarma i događaja kroz čitav industrijski kompleks ili objekat. On je kompletan sistem za management i upravljanje alarmima.

Korisnik može umrežiti sve računare i izvore alarma i događaja za otkrivanje, sortiranje, filtriranje, gledanje, izvještavanje i potvrđivanje OPC server alarma i događaja.

Osnovne karakteristike:

- istinski 32 bitni , više-končani (multi-threaded), multi platformski dizajn
- Univerzalna OPC konektivnost sa OPC data serverima
- Opslužuje alarme koristeći OPC alarme i događaje
- Koristi Visual basic za aplikacije VBA skripting
- Mrežna klijent-server arhitektura
- Alarmni server se može izvršavati kao NT servis
- Konfigurira granice, brzinu promjene, devijacije i digitalne alarme.
- Alarmi na izraze i kalkulacije
- Prati i loguje operatorske akcije
- Podržava auto-arhiviranje alarmnih zapisa
- Podrška za redundantni printer
- Alarm ActiveX kontrolni elementi za gledanje i potvrđivanje aktivnih alarma: može se uroniti u Visual basic i HTML Internet/intranet bazirane Web stranice.
- ActiveX kontrole za analizu Historijskih alarma: pareto čartovi, izvještaji, histogrami, itd.
- Sistem za traženje i otkrivanje grešaka u alarmima (troubleshooting): ekranski helpovi i informacije za Operatora.
- Publikovanje alarma na Webu
- Jednostavna promjena interfejsnog jezika

ScriptWorX32 je također samostalna aplikacija koja dozvoljava kreiranje i management Visual Basic za aplikacije (VBA) skripti. ScriptWorX32 jedinstveni multitasking i multiprocesor okružaj dozvoljava da višestruke skripte se istovremeno izvršavaju. Korisnički definirane VBA skripte mogu izvršavati OPC read-write operacije na bilo kojem OPC serveru, koji se povezuje sa uređajima u pogonima i ofisima fabrike ili postrojenja, kao što su PLC-jevi, DCS i ostali OPC kompatibilni uređaji i serveri.

ScriptWorX32 simultano izvršava kalkulacije, upravlja bazama podataka, i izvršava sve operacije koje su raspoložive u VBA jeziku da pristupi izvještajima ili izvrši operacije sa recepturama.

Kao dio Genesis32 Enterprise Edicije, on nudi i skripting na nivou projekta.

Osnovne karakteristike su:

- istinski 32 bitni , više-končani (multi-threaded), multi platformski dizajn
- izvršava se na MS Win95, Win98, WinME, WinNT, Win2K, i WinXP.
- Moćni VBA kontejner za skripting
- Uključuje Visual Basic za aplikacije VBA softwareski alat
- Simultano izvršava VBA skripte
- Automatski kreira VBA skripte koristeći Skript Wizard

- Izvršava skripte na bazi vremenskog rasporedjivanja
- Izvršava skripte na bazi OPC data događaja
- Izvršava skripte na bazi OPC alarma
- lako koristiva konfiguracija
- Skript monitor obezbjedjuje mogućnosti analize
- Kompilira VBA skripte za višestruko korištenje
- Interfejsira se sa *MES* (management enterprise systems)/ *ERP* (enterprise resource planning) sistemima na nivou korporacije ili velikog preduzeća.
- Integrira MS Access, MS SQL Server, Oracle i MSDE baze podataka
- Obezbjedjuje globalni skripting ka drugim aplikacijama.
- Obezbjedjuje puni OLE automation interfejs
- Integrira se sa aplikacijama drugih vendora.

ProjectWorX32 je vrsta super konfiguratora koji integrira sve GENESIS32 aplikacije u jedan format lagan za manipulaciju. GENESIS32 sadrži brojne aplikacije sa višestrukim komponentama, što zahtjeva dobro trenirane projektante softwareskih aplikacija da kreiraju, primjene i upravljaju projektima. Svaka od ovih komponenti zahtjeva kompleksni setup od mnogo različitih tipova konfiguracionih fajlova, baza podataka, i OPC servera, kao i setinge za runtime i sigurnost. Sve to je objedinjeno u ProjectWorX32 korisničkom interfejsu koji je ustvari jedan kontejner uronjen u Microsoft Management kontolu (MMC).

ProjectWorX32 konzola pogodno konsoliduje sve GENESIS32 fajlove u jednu lako dostupnu lokaciju, omogućujući korisniku da:

- kreira nove GENESIS32 aplikacione fajlove i OPC server konfiguracije
- organizira sve GENESIS32 aplikacione fajlove u posebne projekte
- uvozi aplikacione fajlove u projekte
- starta i zaustavlja GENESIS32 aplikacije
- pakuje projektne fajlove u jedan, komprimirani fajl , i raspakuje projektne fajlove za lagano prenošenje na više računara na kojima će se izvršavati
- aktivira sve projekte u runtime modu
- kontrolira layout prozora koristeći Screen Manager
- konfigurira sigurnosne setinge na password zaštićene projekte
- backupira projektne fajlove na Microsoft Visual SourceSafe
- traži imena za zamjenu imena mašina na kojima će se programi izvršavati kao i za OPC tagove kroz sve projektne fajlove koristeći globalni find/replace utility.
- Publikuje projekte na Webu.
- Kontroliše i nadzire runtime funkcije za sve aplikacije koristeći GenTray.
- Lansira GENESIS32 systemske alate
- Generira sveobuhvatne izvještaje za sve fajlove i projekte.
- Uvozi i upravlja sa aplikacijama od ostalih vendora.

Screen Manager

Screen Manager (manager ekrana) je utility program za upravljanje layoutom displeja koji se koriste kao operatorski interfejs. Manager ekrana podržava konfiguraciju ne samo za kompjuter sa jednim monitorom nego takodjer i sistem sa više monitora, omogućavajući da definicije različitih layouta ispune ekrane. Manager ekrana podržava ne samo GENESIS32 aplikacione dokumente nego i dokumente drugih vendora, (napr. postavljanje CAD dokumenta ili MS Word dokumenta).

DataSpy

OPC DataSpy je utility program za spajanje sa OPC serverima, i služi kao test aplikacija a i kao pogodan alat za gledanje OPC informacija. Može testirati OPC servers koji su u saglasnosti sa OPC 1.0 i 2.0 Data Access (DA) standardom, kao i sa OPC 1.0 Alarm and Events (AE) standardom. Alati za analizu omogućuju brzi troubleshooting OPC servera drugih Vendra.

GenBroker

DCOM je metod komunikacije koji se koristi u GENESIS32. Da bi se medjutim, prevazišli neki problemi u korištenju DCOM-a , a prije svega oko time-outa, bilo je potrebno eliminirati Windows callbackove (povratne pozive), i smanjiti broj različitih funkcionalnih callova putem Windows DCOM komunikacija.

DCOM glavni nedostaci su:

- njegovi kompleksni setinzi često čine njegovu konfiguraciju neefikasnom kada je klijent udaljen.
- ne radi dobro kroz različite NT domene
- konekcije putem DCOM-a putem sporih i nepouzdanih mreža (napr. preko Interneta) su praktično nemoguće.
- nije lako konfigurirati prolaz kroz firewallove.

Kao odziv na ove poteškoće, ICONICS je opremio GENESIS32 sa **GenBroker**, provjerenim utilitijem sa visokom brzinom komunikacije. Primjenjujući mrežu aplikacija, **GenBroker** koristi **TCP/IP** i **SOAP/XML** kanale da postigne real-time i sigurne komunikacije izmedju Web Browser klijenata i WebHMI servera. Sve, izuzev metoda komunikacije ostaje isto kao i kod DCOMa.

Dr. DCOM

Dr. DCOM je utility za pomoć u postavljanju DCOM konekcija za umrežene OPC klijente i servere. On testira OPC interfejse i na lokalnim mašinama kao i na onima u mreži. Postoji i autokonfiguracioni Wizard da asistira u brzom postavljanju mrežnih riješenja.

Sigurni Server (security server)

Security Server je utility program za upravljanje pravima pristupa korisnicima i grupama korisnika unutar GENESIS32 ansambla (suite) aplikacija. Konfiguracioni alat dozvoljava kreiranje kastomiziranih grupa kao i doznačavanje pojedinaca ovim

grupama(korisnik se može doznačiti u više od jednu grupu). Razne programske opcije se mogu ograničiti , na bazi toga ko je logiran na računaru za vrijeme **Runtime**.

Konfigurirše takve detalje kao što su:

- bazični, napredni i NT integrirani modovi sigurnosti
- restrikcije pristupa fajl ovima
- OPC point/tag dozvole upisivanja
- dozvola vremenskog logiranja
- kontrola stanice logiranja
- korisnički definirana sigurnost
- mangement korisnik/grupa računa i njegovog passworda (maksimalna starost passworda, minimalna dužina passworda, itd)
- zaključavanje računa kod pogrešnog logiranja
- automatski logout nakon određenog vremena neaktivnosti za visokonivovske korisnike sa velikim pravima
- pristup razvojnom modu (development mode)
- preključenje izmedju taskova
- pristup prozorima
- online konfiguriranje
- nekoliko individualnih karakteristika i funkcija se nudi od strane raznih klijent aplikacija unutar GENESIS32. Naprimjer, ograničava ko ima pravo pristupa štampanju, izvještajima, zumiranju, meni komandama, preključenju ekrana, manipulacijama u trend ekranima, potvrdama alarma, i mnogo više.

WebHMI

WebHMI obezbjedjuje OPC i vizualizacionu informaciju iz bilo kojeg ICONICS GENESIS32 OPC baziranog proizvoda na bilo kojem računaru sa MS Internet Explorerom ili Netscape Navigator Web Browserom. Baziran upotpunosti na MS DNA arhitekturi, WebHMI koristi ActiveX i DCOM tehnologiju da obezbjedi Web Baziranu vizuelizaciju. Integrira korisnikovu HMI aplikaciju sa tradicionalnim Web browserima da obezbjedi daljinski i jeftini monitoring glavnih proizvodnih informacija iz procesa. WebHMI dozvoljava ne samo gledanje displej informacija nego takodjer obezbjedjuje sigurnost, unos podataka, i interakciju u realnom vremenu sa korisničkom aplikacijom. WebHMI dozvoljava korisniku da gleda preko Interneta ili intraneta standardne operatorske prikaze i animacije, trendove sa živim real-time podacima, izvještaje i alarme.

Osnovne karakteristike su:

- Izvršava se na MS Win95, Win98, WinME, WinNT, Win2K, WinXP
- lagani (thin) Web klijent
- Web mogućnosti sa nultim obimom instalacije
- isporučuje ActiveX kontrole i HTML stranice
- potreban je MS IE 5.5 ili viša verzija
- obezbjedjuje OPC real-time podatke preko Weba
- aploduje (upload) grafiku , trendove, alarme itd.
- idealan za gledanje udaljenih HTML kontrolera

GEN-OPC SERVER

GEN-OPC Server obezbjedjuje OPC interfejs za ranije verzije GENESIS HMI softwarea kao što su ICONICS GENESIS Control Series (GENESIS for DOS), i GENESIS for Windows (GFW). Sa njim se korisnik može povezati sa ranijim, još postojećim verzijama GENESIS HMI paketa. Imena tagova koja se koriste u ovim starijim verzijama ICONICS HMI paketa se pojavljuju kao OPC tagovi u okviru Tag Browsera , što omogućuje jednostavno uspostavljanje konekcija putem point-and-click spajanja.

DataWorX32

DataWorX32 obezbjedjuje tehnologiju, koja uključuje OPC server-to- OPC server premošćenje podataka (data bridging), OPC agregaciju kao i mission-critical OPC baziranu redundantnost. Ova data-bridge tehnologija omogućava povezivanje bilo kojeg OPC servera sa drugim OPC serverom, obezbjedjujući plug-and-play konektivnost medju njima. Naprimjer, pomoću ovog softwarea možemo spojiti Allen Bradley PLC OPC server sa Siemens PLC OPC serverom i lagano i transparentno djeliti podatke. Ti podatci se mogu takodjer koristiti i u izrazima i kalkulacijama.

Osovne karakteristike su:

- istovremeni i simultani rad i kao OPC klijenta i servera
- Izvršava se na MS Win95, Win98, WinME, WinNT, Win2K, WinXP
- potpuni 32-bitni, multi-konični, multi-platformski dizajn softwarea
- višenivovska redundantnost OPC podataka
- OPC server –to- OPC server most podataka (data bridge)
- agregacija (objedinjenje) OPC podataka
- analizator statistike i performanse
- ugradjeni izrazi i kalkulacije
- podržava interfejs sa OPC browserom
- izvršava se kao NT servis
- integrirani monitor podataka u realnom vremenu
- online konfiguriranje OPC tagova
- NT nivoi sigurnosti
- podrška za COM/DCOM i OLE automation
- CSV import/export podataka za editiranje u bloku

Multimedijalni alarmi (AlarmWorX32 multimedia)

AlarmWorX32 multimedia je opcija za paket AlarmWorX32 , koja obezbjedjuje OPC kompatibilnu multimedijalnu tehnologiju, za daljinsko prenošenje alarma i upozorenja. Na raspolaganju su različiti multimedijalni "agenti ", kao naprimjer:

- paging (uključivo i SMS/TAP podrška)
- dvosrtani paging
- telefon sa podrškom prijema (call-in) i slanja (call-out) poruke
- email
- Fax

- najavljivanje alarma glasom putem text-to-speech konvertora
- najavljivanje alarma glasom putem zapisanih *.wav fajlova
- video
- pop-up prozorima
- oznakama na kompjuterskim ekranima (marquee)
- trenutačnim porukama (instant messaging)

ControlWorX32

ControlWorX32 je kontrolna aplikacija koja vrši upravljanje procesom ili postrojenjem. Njen konfigurator je u potpunosti kompatibilan sa svim IEC 6-1131 standardnim programskim jezicima i u potpunosti je u saglasnosti sa industrijskim standardima. Takodjer je u potpunoj saglasnosti sa OPC serverima. Korisnik može kreirati, testirati, i debugirati IEC 6-1131 PLC Soft logičke i procesne aplikacije sa lakoćom i elegantnošću zrelog softwareskog paketa.

Osnovne karakteristike

- IEC 6-1131 kompatibilan i ima podršku za standardne industrijske programske jezike za razvoj kontrolnih i upravljačkih aplikacija kao što su : SFC (sequential function chart – sekvencijalne graf funkcije), FBD (Function block diagram – funkcionalni blok dijagram), LD (ladder diagram – stepenasti ili šalt dijagram), IL (Instruction list – instrukcione liste), ST (structured text – strukturirani tekst).
- OPC konektivnost i drajveri za Interbus-S, CAN, Profibus , Devicenet, Fieldbus i mnoge druge industrijske baseve u polju postrojenja (field bus).
- podrška višestrukim i multiplim dokumentima
- brzo i multitasking izvršenje u realnom vremenu (i do milisekundne brzine).
- mješavina IEC 31 grafičkih jezika
- Editiranje u okviru prozora “ object & alarm” , objektno bazirano editiranje SFC koraka, IL instrukcija
- moćni ugrađeni debager u realnom vremenu za testiranje kontrolne logike
- online konfiguriranje: nema potrebe za upravljanjem putem interupta, da bi se implementirala nova ili promjenila postojeća logika
- kreiranje višestruko iskoristivih (reusable) kontrolnih objekata.

Mobilni HMI interfejs (MobileHMI)

MobileHMI , ICONICS-ov softwareski proizvod Web omogućene mobilne telefonije, spaja korisnike sa real-time proizvodnim i procesnim podacima putem bežičnih PDA (portable digital assistant – dlanovnika), kao što su Palm, Pocket PC , Web omogućeni mobilni telefoni kao i drugi prenosivi uređaji. MobileHMI , je prvi ICONICS-ov softwareski proizvod koji je inkorporirao novu Microsoftovu NET. framework tehnologiju i Mobile Internet Toolkit. Koristi Wireless Application Protocol (WAP) tehnologiju, koja je trenutačno vodeći standard za aplikacije u bežičnim mrežama.

WAP obezbjedjuje uniformnu tehnološku platformu sa konzistentnim formatima za isporučivanje Internet i intranet baziranih informacija i servisa za digitalne mobilne telefone i druge bežične uređaje. MobileHMI podržava sve uređaje koji koriste WAP, WML, cHTML i HTML tehnologije.

ReportWorX.NET

Ovaj softwareski alat je dizajniran od temelja (bottom-up), korištenjem Microsoft .NET i Microsoft Excel tehnologije i napisan je u Microsoft C# (C-sharp) programskom jeziku, te VB.NET i ADO.NET jezicima. ReportWorX.NET brzo i efikasno pretvara ogromnu količinu podataka iz bilo kojeg izvora podataka u detaljne izvještaje. Time ICONICS u okviru GENESIS32 softverskog ansambla obezbjedjuje najnaprednije izvještavanje koje je danas na raspolaganju, koristeći najnaprednije Microsoftove tehnologije. ReportWorX.NET koristi Microsoft .NET tehnologiju da izveze podatke u radne listove i forme izvještaja i kontrolira izvršenje ovih Excel formi bez potrebe da se Excel vidljivo izvršava. Dupliciranje Workbookova, manipulacije, ažuriranja čartova i podataka, ponovno izvršavanje formula i automatsko štampanje i raspoređivanje se u potpunosti izvršavaju od strane ReportWorX.NET-a.

Biblioteka simbola

GraphWorX32 podržava korištenje smart simbola i displeja. Smart simboli sadrže konekcije sa tačkama podataka (data-point). Ako loadujemo smart simbol u naš displej u animacionom modu, simbol će početi da trenutno prikazuje podatke, na bazi konekcije sa tačkom podataka. GraphWorX32 koristi dva različita moda za biblioteku simbola:

- standardnu autonomnu biblioteku simbola
- dokabilnu(priključivu) toolbar (alatnu letvicu) simbola

Biblioteka simbola je raspoloživa u GraphWorX32 kao priključiva alatna letvica ili kao plutajući prozor (floating window) unutar displej fajlova. Može se slobodno pomjerati iznad GraphWorX32 aplikacije ili se može priključiti na bilo koju stranu GraphWorX32 displeja. Osobine biblioteke simbola su u potpunosti implementirane u priključivoj alatnoj letvici. Korisnik može kreirati, promjeniti ime, i izbrisati i kategorije fajlova i simbole unutar priključive alatne letvice, kao i otključati kategorije sa passwordima i promjeniti izgled ikona simbola u view panelu. Kategorija fajl formata priključive alatne letvice je u potpunosti kompatibilna sa prethodnim verzijama biblioteke simbola, i obratno.

ActiveX toolbox

ActiveX toolbox obezbjedjuje korisniku alate za kreiranje njegovih vlastitih kastomiziranih ActiveX kontrolnih elemenata i njihovo uranjanje (embedding) u ActiveX kontejnere kao što je GraphWorX32. Korisnik može brzo kreirati aplikacije i profesionalno sa visoko performansnim ActiveX kontrolama koji su istinski OPC klijenti.

Osnovne karakteristike su:

- dizajniran za MS Win95, Win98, WinNT i Win2K

- prvi softwareski proizvod sa OPC omogućenim ActiveX kontrolama
- uključuje primjere sa Visual Basicom i HTML primjere
- integriše se sa Visual Basic i Microsoft Officeom.
- integriše se sa HTML i Web stranicama
- povećava i proširuje ICONICS WebHMI rješenja
- realistični 3D ActiveX kontrolni elementi
- napredni ActiveX kontrolni elementi sa klizačem (slider)
- periodični ActiveX da trigeruje Visual Basic skripte
- moćni data/event ActiveX kontrolni elementi
- uključuje: (gauge) mjerni instrument, prekidač (switch), i druge kontrole
- multi-konični, multi-procesorski dizajn
- integrirani browser za OPC tagove
- brza dinamička animacija
- potpuna dokumentacija
- ažuriranja bez treptaja (flicker-free) i sa visokom performansom
- višerezolutni i skalabilni ActiveX kontrolni elementi
- periodično raspoređeni ActiveX : trigeruje skripte na vremenski baziranim intervalima.
- Event ActiveX upravljani sa podatkom : spaja skripte na bilo koji OPC tag.
- OPC Alarm ActiveX : trigeruje skripte na bazi OPC alarma
- timerski ActiveX: jednostavni vremenski upravljani ActiveX.
- Numerički ActiveX: različiti načini da se pokažu numerički podatci.

OPC ToolWorX

OPC ToolWorX je alat za kreiranje OPC 1.0 i 2.0 Data access (DA) i OPC Alarm & Events serverskih i klijentskih aplikacija. Paket uključuje OPC konfiguracioni GUI, Tag browser, Modbus OPC server kao primjer, dokumentaciju i objašnjenje OPC koncepata i OPC test za klijenta.

Pošto postoji na softwareskom tržištu vrlo veliki broj uređaja i servera, potrebno je za OEMs, krajnje korisnike i sistem integratore da ponekad napišu svoje vlastite OPC kompatibilne serverske i klijentske aplikacije. OPCToolWorX implementira napredne programske koncepte od OPC Specifikacija.

OPC toolWorX sadrži dva integrisana alata:

- OPC server alat (toolkit)
- OPC klijent alat

Ovi toolkitovi sadrže izvorni kod aplikacije uzorka, dokumentaciju, dva primjera servera, klijent test aplikaciju, i Visual Basic test aplikaciju.

OPC ToolWorX osobine su :

- nudi dva potpuno nova OPC ToolWorx proizvoda
 - punu izvornu verziju
 - jeftinu verziju bez izvornog koda (source code) – LIB verziju
- slobodno končani model

- Microsoft ActiveX i COM programiranje
- OLE automation DLL
- Multi-končana (multi-threading) softwareska arhitektura
- OPC konfiguracioni explorer
- OPC toolkit izvorni kod
- univerzalni Tag browser
- kreira OPC in-process servere za Windows CE
- koristi samo MS Visual C++
- Podrška za Windows CE procesore : MIPS, Hitachi S3, S4, Intel, Strong Arm
- kreira OPC servere kao NT servise
- podržava MS SQL Server, MS Access, i druge konfiguracije baza podataka
- sadrži wizarde koji kreiraju servere u kratkom vremenu sa OPC data access i OPC Alarms & Events
- novi moćni distribuirani udaljeni pristup bazi podataka putem ADO/OLEDB protokola
- univerzalni tag browser sa podrškom za alarme i događaje (A&E) i (HDA)
- moćni wizarde za brzi razvoj (rapid development)
- podrška za pakovane bite (packed bits capability)
- podrška za OPC 2.0 kastomizirani intefejs i OLE Automation interfejs
- podrška za najnovije metode OPC mrežnog browsera
- mnoge nove optimizacije u runtimeu

ActiveX ToolworX

ActiveX ToolWorX je prvi industrijski OPC bazirani ActiveX toolkit. Omogućava korisniku da kreira svoje vlastite ActiveX kontrole koji su istinski OPC klijenti. Zatim može da spoji ActiveX kontrolni element na bilo koji OPC server, i uroni ga u neki od popularnih ActiveX kontejnera kao što su : Visual Basic, Web bazirane HTML stranice, Microsoft Word, MS Excel, MS Powerpoint i mnoge druge ActiveX kontejnere drugih vendora. Zatim može da kreira biblioteke ponovno iskoristivih (reusable) komponenti na bazi popularne Microsoft ActiveX tehnologije. Koristeći standardni OPC plug-and-play, može ih spojiti na uređaje u postrojenju kao što su PLC i digitalni regulatori, I/O uređaji itd.

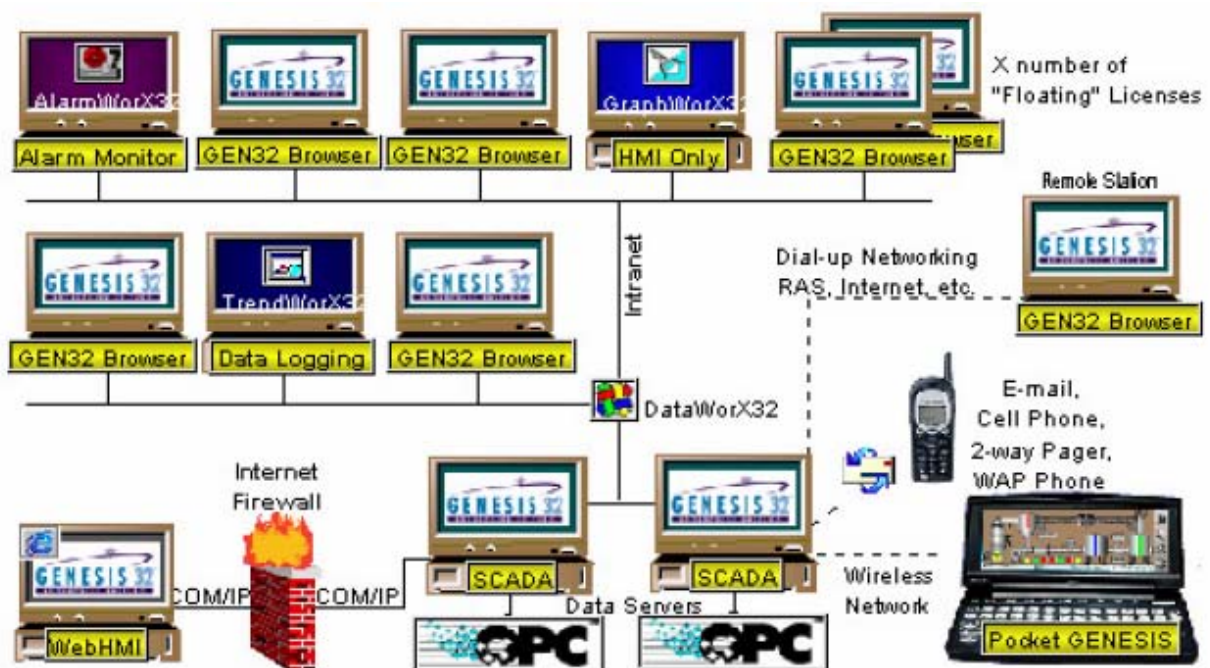
Osnovne karakteristike su:

- izvršava se na Win95, Win98, WinNT, Win2K
- prvi industrijski OPC bazirani ActiveX alat
- kreira OPC omogućene HMI ActiveX kontrole
- uranja ih u bilo koji ActiveX kontejner
- uključuje i moćni GraphWorX32 kontejner
- kreira realističan izgled ActiveX kontrola
- moćne animacione mogućnosti
- uključuje i ActiveX Gauge izvorni kod
- brza dinamička animacija
- više-konični (multi-threading) dizajn
- uključuje OPC tag browser
- skalabilni ActiveX kontrolni elementi
- dinamička ažuriranja bez treperenja (flicker-free)

- plaguje se u Visual C++
- brzo kreiranje OPC omogućenih ActiveX kontrola koristeći ActiveX wizarde
- tag browser wizard za brzu generaciju koda iz tag browser intewrfejsa
- brzo pretvaranje GraphWorX32 kreiranih simbola u OPC omogućene ActiveX

Dizajn sistema

Kada dizajniramo sistem, moramo imati u vidu da različite aplikacije se mogu izvršavati kao samostalne ili kao ansambl na svakom od PC-jeva u mreži. Naprimjer, sistem na narednoj slici pokazuje nekoliko mješanih GENESIS32 sistema sa stanicama koje su posvećene funkcijama alarmiranja, druge su to za trendiranje, a na trećim se pokazuju samo grafički ekrani za nadzor i vođenje. GENESIS32 nudi fleksibilnost da se skalira i ekspanira , kako se mjenjaju potrebe sistema.



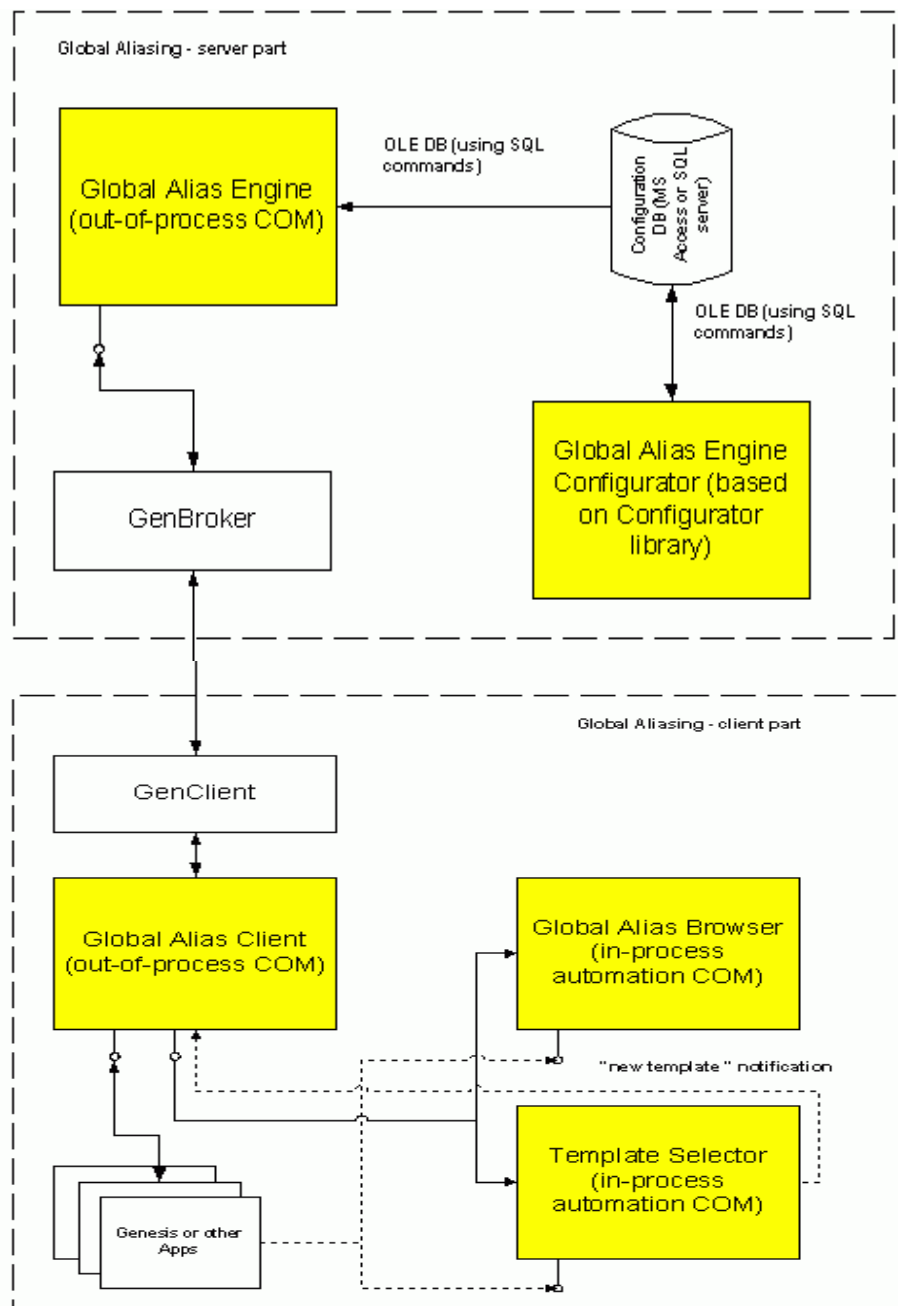
Primjer projektovanja sistema

GLOBALNI ALIASING UNUTAR GENESIS32 SOFTWARESKOG SISTEMA

Globalni aliasing (drugo ime , zamjensko ime), je novi snažni mehanizam implementiran u GENESIS32 klijente i servere, GraphWorX32, TrendWorX32 i AlarmWorX32. Globalni aliasi se pohranjuju u centralnu globalnu bazu podataka (MS SQL Server ili Access), i može im se pristupiti od strane raznih klijenata preko mreže. Da bi se obezbjedila zadovoljavajuća performansa obzirom na brzinu prenosa i razmjene podataka u HMI/SCADA sistemu koja ima naglašeni atribut realnog vremena, lokalni klijenti se ne konektiraju direktno na bazu podataka (globalni aliasing engine

server) , nego komuniciraju sa globalnim aliasing klijentom, koji će dobiti zahtjevanu informaciju za njih i lokalno je kaširati i staviti na raspolaganje lokalnom klijentu.

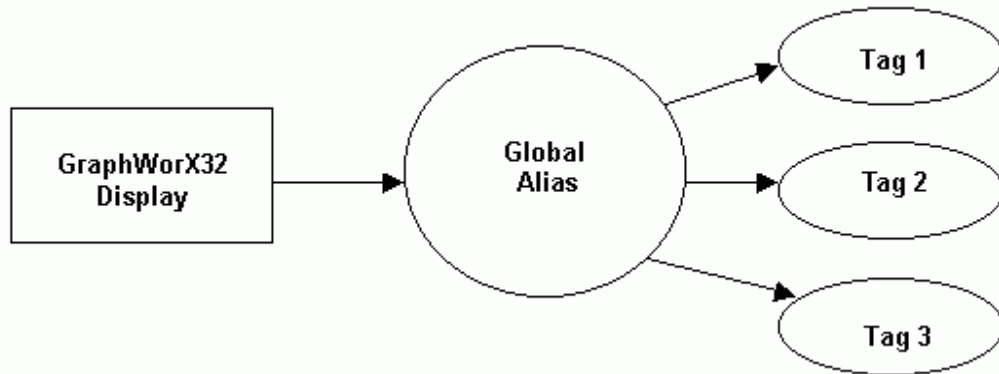
Naredna slika pokazuje globalnu aliasing arhitekturu sistema:



Globalna aliasing arhitektura sistema

Globalni alias je string podatka koji omogućava korisniku da referencira višestruke izvore podataka (naprimjer pero za zapisivanje analognog signala procesnog taga u TrendWorX32 paketu i procesnu tačku (ppt) u GraphWorX32) sa jednim jedinstvenim imenom. Pošto možemo referencirati višestruke izvore podataka iz jedne lokacije, globalni aliasing može reducirati ukupan broj individualnih displej fajlova koje treba da kreiramo kod razvoja aplikacije.

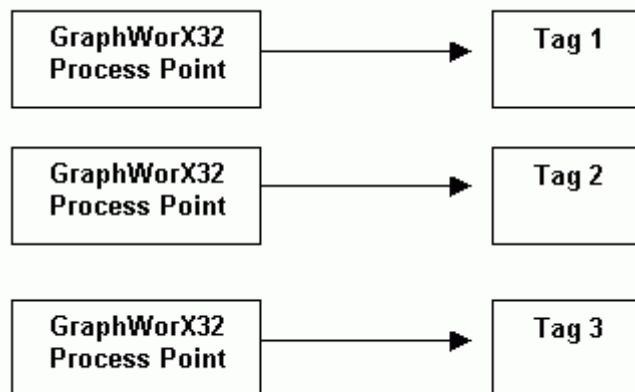
Globalni aliasing sistem obezbjeđuje globalno pohranjivanje za stringove aliasa i da ih pretražuje i razlučuje u runtimeu. Aliasi se mogu koristiti u OPC tagovima, elementima u menijima, imenima fajlova, itd. Naredna slika naprimjer pokazuje kako globalni alias se povezuje sa više OPC tagova u GraphWorX32 displeju. Vrijednosti podataka sa kojima su udruženi globalni aliasi se zovu vrijednosti aliasa. Svi globalni aliasi i alias vrijednosti se kreiraju i konfiguriraju u globalnom alias konfiguratoru.



Jedan alias vezan na više izvora podataka

Alias rezolucije i teme

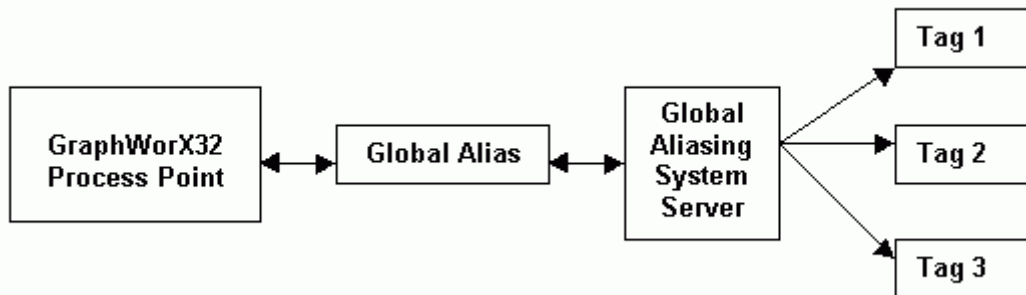
Rezolucija aliasa je proces doznačavanja specifične tag vrijednosti tom aliasu. Korisnik može doznačiti jedan ili više tagova aliasima (napr. Tag 1), kako je pokazano na narednoj slici:



Aliasing višestrukih tagova

Svaki globalni alias se može pridružiti sa temom. Tema povezuje globalni alias sa jednom ili više vrijednosti podataka. (tj. tagova). Globalna aliasing baza podataka dozvoljava korisniku da doznači aliasima različite vrijednosti na trenutno izabranu temu. Teme se mogu dinamički mjenjati, bilo programski ili kroz automatizirane (automation) API , ili putem GraphWorX32 pick (izbor) akcije. Naredna slika pokazuje kako generički jednostruki alias se može koristiti, naprimjer da poveže višestruke tagove, na jednu procesnu tačku, u GraphWorX32. Globalni aliasing engine server razlučuje alias

vrijednosti i elemente tema , koje su udružene sa vrijednostima podataka (tj.. sa tagovima), kako je to pokazano na narednoj slici:

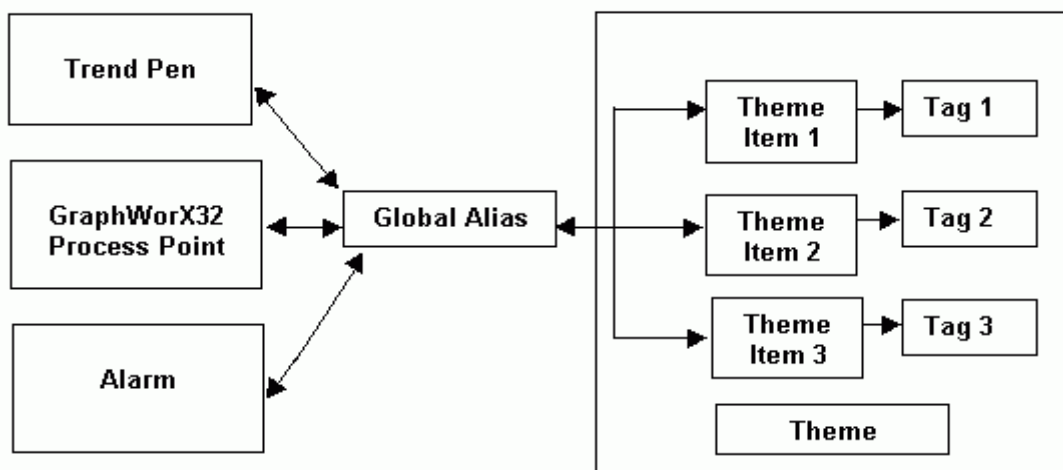


Alias rezolucija

Svaka tema može imati jedan ili više elemenata teme udruženih sa njom, i svaka vrijednost aliasa može biti pridružena sa elementom teme. Element teme omogućava korisniku da preključuje izmedju vrijednosti podataka koji su povezani sa temom, kao što je to pokazano na narednoj slici. Naprimjer, ako imamo jednu temu sa tri elementa teme, i svaki element teme je pridružen sa tagom, možemo preključivati izmedju elemenata teme i time i izmedju tagova.

Dakle, imamo jednu procesnu tačku koja je udružena sa tri različite vrijednosti, kako se vidi i na narednoj slici. Sve teme i elementi tema se kreiraju i konfiguriraju u globalnom aliasing konfiguratoru.

Nadalje, jedan alias može biti očitavan od strane mnogih aplikacija (napr. GraphWorX32, TrendWorX32, AlarmaWorX32), kako je to pokazano na narednoj slici. Možemo takodjer pridružiti višestruke aliase sa jednom temom.



Korištenje elemenata teme da biramo izmedju različitih vrijednosti podataka

Obuhvat (scope) tema

Rezolucija aliasa je kontrolirana sa obuhvatom (scope) teme, tj. na kojem nivou je tema upotrebljena u aplikaciji. Globalni aliasing sistem podržava tri različita nivoa aliasa, kako je to pokazano na narednoj slici:

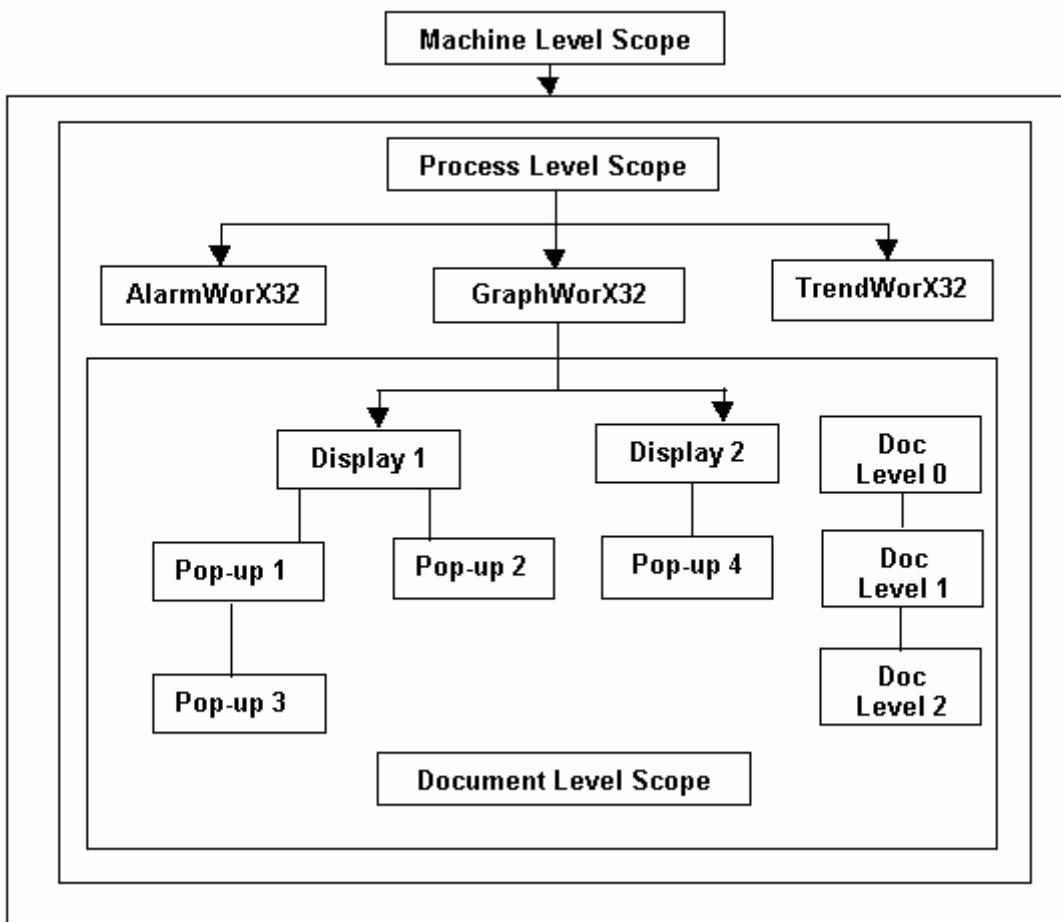
- **mašinski nivo** (machine-level) : rasprostire se na cjelokupnu mašinu (tj. računar)
- **procesni nivo** (process-level): rasprostire se na tekući proces (tj. GraphWorX32 sa pop-up prozorima, TrendWorX32 pero, ili AlarmWorX32 viewer za prikaz alarma u hronologiji događanja)
- **nivo dokumenta** (document-level): rasprostire se na tekući dokument ili prikaz i na njegove unjete (embedded) kontrolne komponente (napr. GraphWorX32 pop-up prozor).

Opaska : Aliasing na nivou dokumenta je raspoloživ samo u GraphWorX32. Ne primjenjuje se kod ActiveX kontrolnih elemenata u TrendWorX32 i AlarmWorX32 ekranima.

Alias se doznajuju datoj grupi na bazi slijedećih delimitera (razdjelnika)

- <!MACHINE>
- <@PROCESS@>
- <#DOCUMENT#>

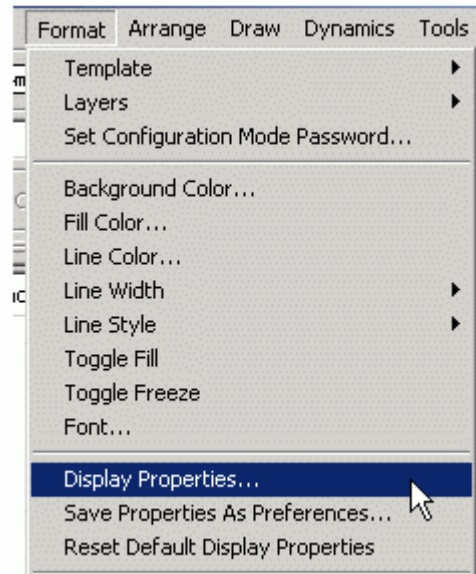
U GraphWorX32 , korisnik može imati višestruke nivoe dokumenata, ili pop-up prozore, kako je to pokazano na narednoj slici:



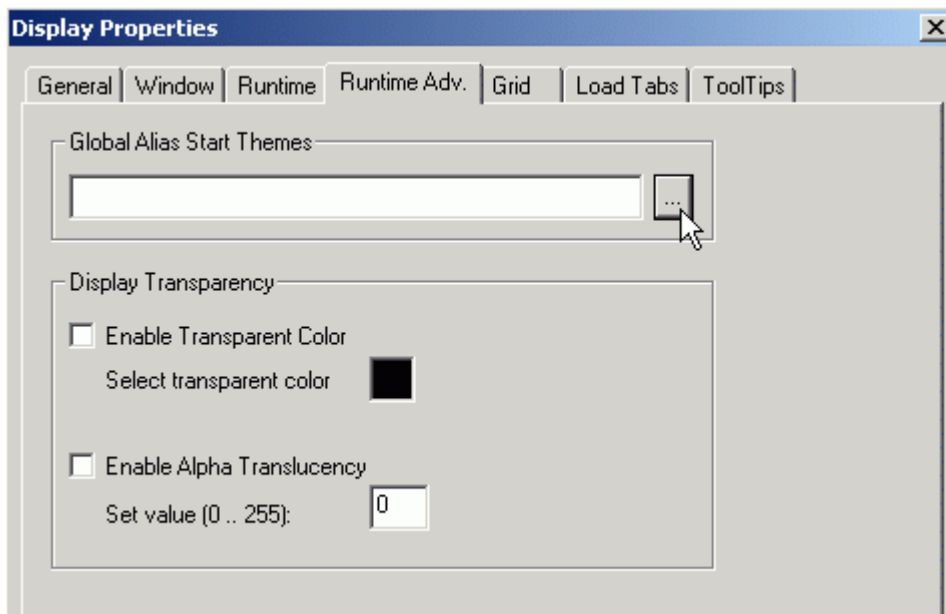
Prikaz obuhvata (scope) za teme

Obuhvat teme na nivou dokumenta je definiran u okviru GraphWorX32 osobina prikaza (display properties). Da bi se definirao default scope teme u okviru GraphWorX32 treba uraditi slijedeće:

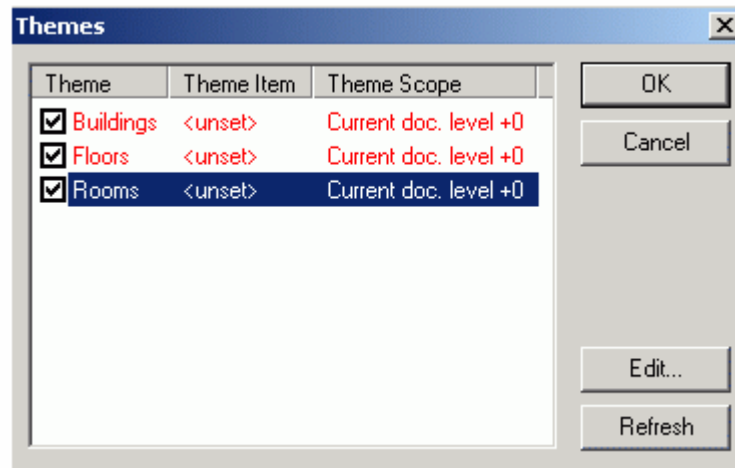
1. Izabrati **Display properties** iz **Format** menija, kao što je pokazano na narednoj slici:



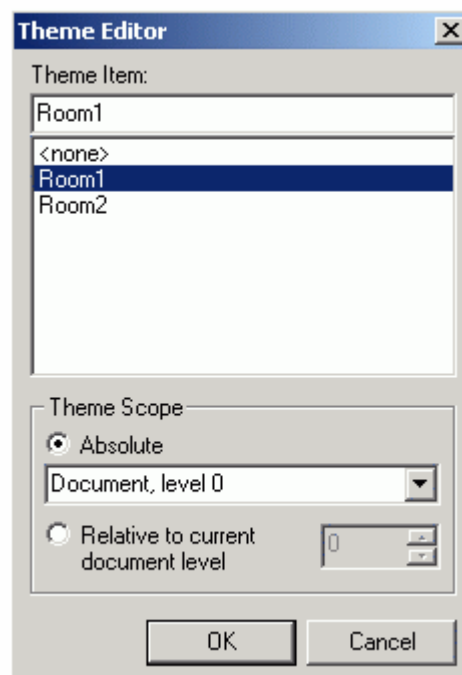
2. Ovo će otvoriti Display Properties dijalog boks. Kliknuti na Runtime Advanced tab, kako je to pokazano na narednoj slici:



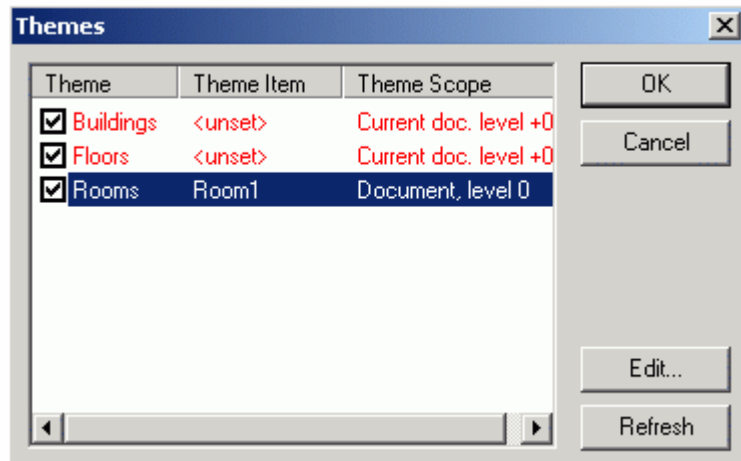
3. Da se izabere startna tema globalnog aliasa, treba ili unjeti ime teme u polje teksta Global Alias Start Themes ili kliknuti ... taster da se izabere tema iz Themes dijalog boksa , kao na narednoj slici:



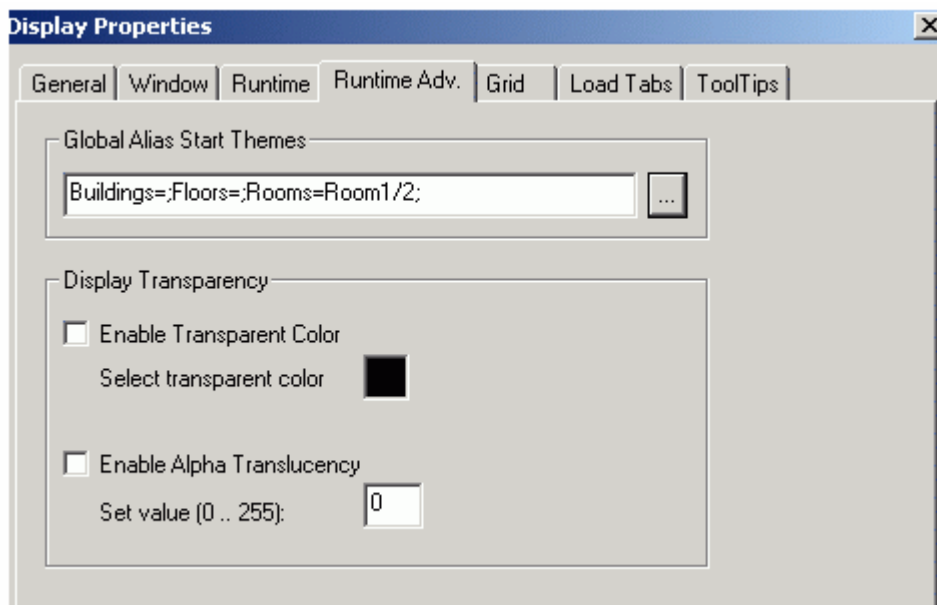
3. Da se definiira obuhvat teme (theme scope), treba naznačiti (highlight) temu i zatim kliknuti na **Edit** taster. To će otvoriti **Theme Editor** dijalog boks, koji je pokazan na narednoj slici. Treba izabrati Theme Item (napr. "Room1"), da se pridruži startnoj temi. U okviru **Theme Scope** , možemo specificirati **Absolute** obuhvat teme (tj. sve nivoe od mašinskog, preko procesnog do nivoa dokumenta). Alternativno, možemo specificirati obuhvat za temu koja je relativna sa tekućim nivoom dokumenta (**relative to current document level**), napr. glavni prikaz , pop-up prozor 1, pop-up prozor 2, itd.



4. Izabrani **Theme Item** i **Theme Scope** su sada prikazani u **Themes** dijalogu, na narednoj slici. Kliknuti **OK**.



5. Izabrane teme se sada pojavljuju kao **Global Alias Start Themes** u **Display Properties** dijalog boksu, kako je to pokazano na narednoj slici:



Implementacija global aliasisa u GraphWorX32 paketu

GraphWorX32 podržava globalni aliasing povezujući se sa ICONICS globalnim aliasing serverom. Globalni aliasing je integriran u slijedeće GraphWorX32 karakteristike:

- *podrška za vidljive objekte (pravougaonike, elipse, linije, itd.)*
- opis
- kustomizirane komande
- globalni alias browser za opštu stranicu

- *podrška za dinamiku objekata (veličinu, lokaciju, rotaciju, itd.)*
 - OPC tag, gornji i donji opseg taga
 - opis
 - kastomizirane komande
 - globalni aliasing browser za izraze i konekcije
- podrška za tekst objekte (tekst, taster, itd.)
 - labele
 - globalni alias browser za labele i opise
- podrška za tastere za displej
 - displej fajla (automatski podržan naslijedjivanjem od PICK)
 - labela (automatski podržana naslijedjivanjem od teksta)
 - globalni alias browser za labele i opise
- podrška za polja stanja (state fields)
 - vrijednost stanja (StateValue)
 - string stanja (Statestring)
 - default vrijednost
- Pristup globalnom alias browseru iz opisa i jezika edit boksova za Analogni selektor, Animator, boju (color), analognu boju (ColorAnalog) i Digitalni selektor
- Flash , hide (sakrij), lokacija (location), izaberi (pick), procesnu tačku (process point), rotaciju , veličinu i, vrijeme/datum
- Polja stanja i spajanje na kod koji je vezan za stranicu
- dodatni browse tasteri za lokalne varijable kao : InitVal (početna vrijednost), LoRange (donja granica opsega), HiRange(gornja granica opsega), displej tabovi, osobine prozora.
- osobine displeja
 - caption
 - downloadovana vrijednost 2x
 - toggle vrijednost 3x
- Nova tema pick akcije sa izborom GAS (global aliasing system)
- Podrška na nivou komandne linije: GraphWorX32 dozvoljava postavljanje inicijalnih globalnih alias tema koristeći argumente komandne linije, u obliku :
 - Themes="MACHINE"=<Buildings=Building1"

Sintaksa korespondira onoj kod pick akcije Select GAS (Global aliasing system) teme.

Implementacija u TrendWorX32

TrendWorX32 viewer podržava globalni aliasing povezujući se sa ICONICS globalnim aliasing serverom. Slijedeći stringovi se mogu aliasirati:

❖ Title

- Title

❖ Ranges

- Hi Tag, Lo Tag, Format, Format XY

❖ Details

- Format

❖ Pen

- Point Name, Hi Tag, Low Tag, Description, Units, Details Format, Ranges Format, Persistent Group name

❖ URL Path

- Konfigurirana u **General** tab polju na dijalog boksu **TrendWorX32 Viewer ActiveX properties**.

❖ Expressions

- U **Expression Editoru**, kliknuti na **Tags** taster i izabrati **Global Aliases** iz pop-up menija da se otvori **Global Alias Browser**.

Trenutačno (verzija 7.1) TrendWorX32 viewer podržava mašinski nivo i procesni nivo aliasinga. Sva polja, koja podržavaju aliasing, su povezana sa global aliasing browser dijalog, koji se poziva sa klikanjem na ... taster.

Korisnik se prvo poveže sa realtime, stalnim tagom, ili historijskim tagom a zatim ima opciju da aliasira string svih pera ili dio njih. Globalni aliasing je takodjer podržan za drag-and-drop operacije kao i za OLE automation metode.

Implementacija u AlarmWorX32

AlarmWorX32 Viewer ActiveX podržava globalni aliasing povezujući se sa globalnim aliasing serverom. U verziji 7.1 , AlarmWorX32 podržava mašinski nivo i procesni nivo aliasinga. Sva polja koja podržavaju aliasing su povezana sa global aliasing browser dijalogom, koji se poziva sa klikanjem na taster u konfiguraciji.

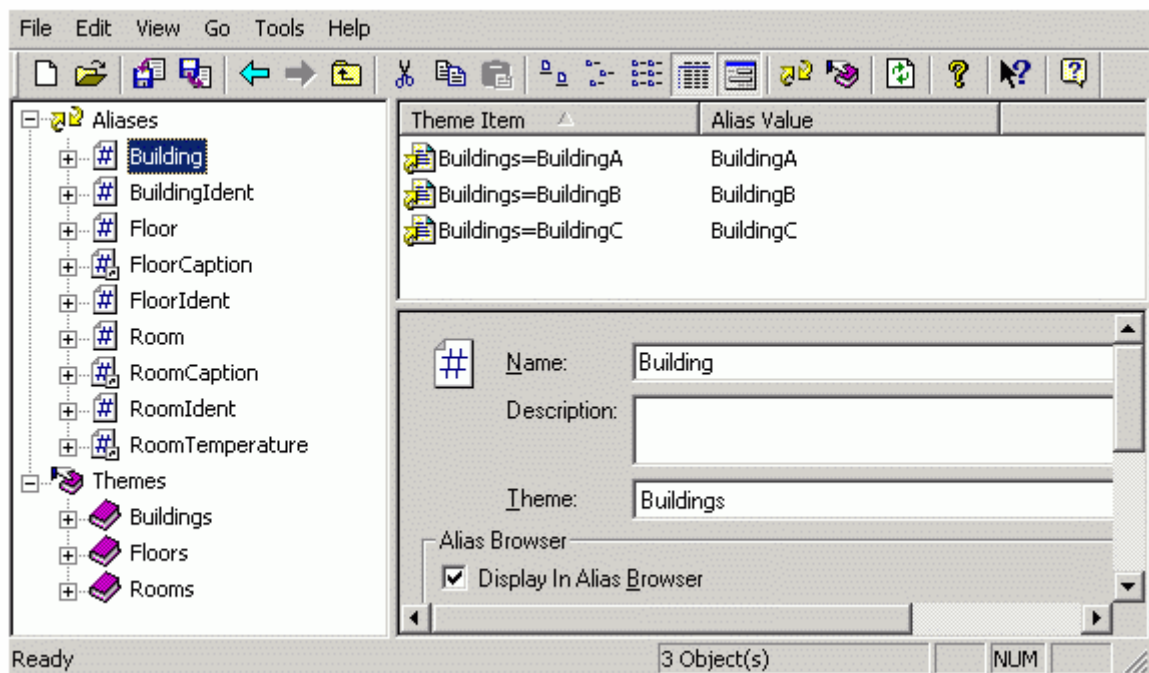
Nadalje, AlarmWorX32 Viewer ima i integrirani global aliasing browser.

Kada specificiramo globalni alias u **Alarm Viewer ActiveX Properties** dijalog boks, korisnik može također izabrati globalni alias iz **Global Alias Browser**, koji uključuje sve globalne aliase u globalnu alias sistem bazu podataka. Ovo eliminira potrebu ručnog unošenja imena aliasa. Klikanjem na ... taster, i izabirući **Global Alias Browser** iz pop-up meni opcija, otvara se Global Alias Browser kako slijedi:

- **Title, Configuration Fajl** , i **URL Path** sekcije u **Alarm Viewer ActiveX** i **Alarm Report ActiveX Properties** dijalog boksu **General** taba.
- Filtriranje alarma putem Expression editora
- **Event Subscriptions** dijalog boks.: **Server, Areas**, i **Sources**
- Filtriranje zapisa Alarm Report ActiveX

Rad sa Global Aliasing konfiguratorom

Global aliasing konfigurator je pokazan na narednoj slici:

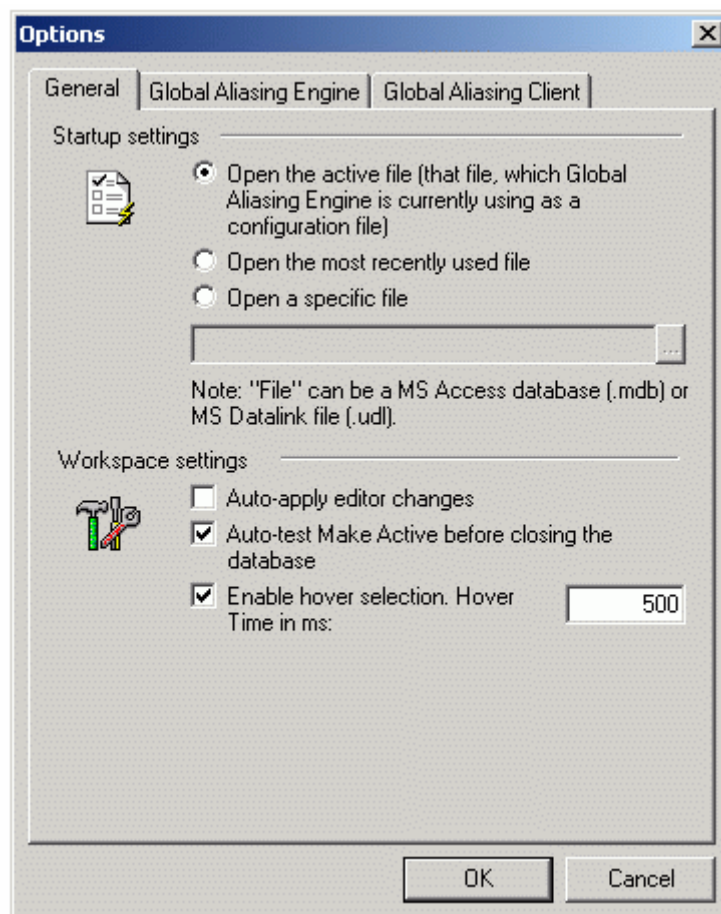


Konfigurator obezbjedjuje standardni format za konfiguriranje baze podataka, kao i uzorni (sample) default global aliasing konfiguracioni projekat. Konfigurator također uključuje i alatnu letvicu i menije sa mnogim komandnim funkcijama.

Alatne letvice uključuju komandne tastere u okviru dvije grupe :

- standardni toolbar
- data manipulation toolbar

U okviru Option dijalog boks imamo **General tab** , koji je prikazan na narednoj slici:



imamo slijedeće opcije:

Startup Settings

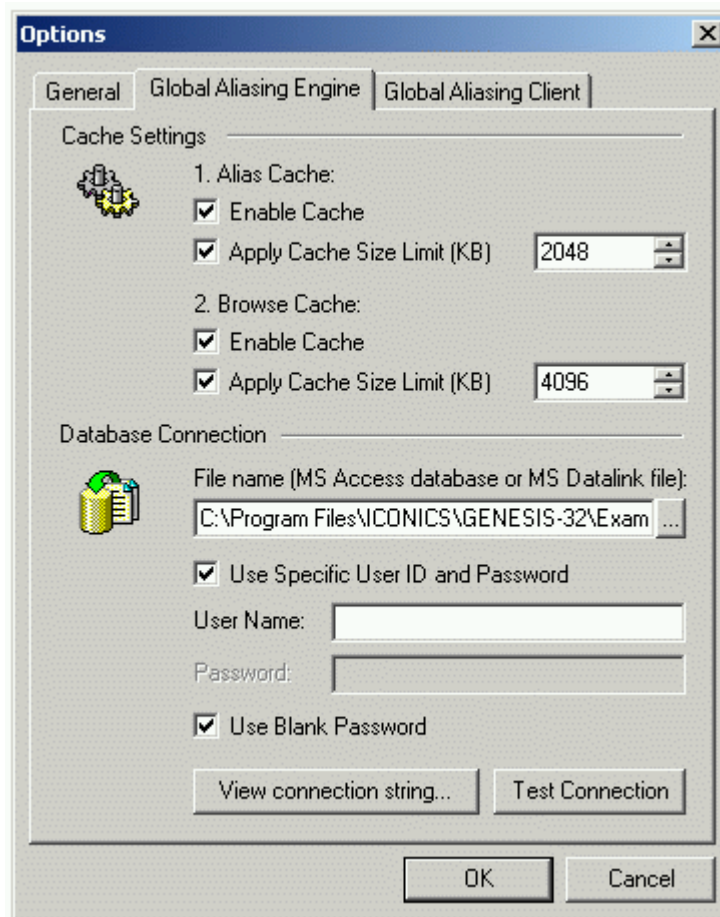
Ova opcija omogućava korisniku da pohrani regionalne setinge u registre OS , tako da se ona primjenjuju svaki put kada startamo globalni aliasing konfigurator. Ovo se primjenjuje i na setinge za jezik, kao i za setinge vremena i datuma.

Workspace settings

Slijedeći **Workspace settings** su na raspolaganju:

- **Auto-apply editor changes**
- **Auto-test Make Active before closing the database**
- **Enable hover selection**

U okviru **Global aliasing engine tab-a** , koji je prikazan na slijedećoj slici:



vidimo da je Global aliasing system engine server lociran na lokalnom čvoru po defaultu, ali može također biti lociran na udaljenom čvoru i njemu se može pristupiti putem intraneta ili interneta. Na ovom dijalogu prozoru se specificiraju slijedeći setinzi za **Global Aliasing engine server** kada je ovaj lociran na udaljenom čvoru:

Cache setinzi:

Global aliasing engine server ima ugrađen data cache koji privremeno pohranjuje globalne aliasing podatke za vrijeme runtimea. Cache setinzi specificiraju parametre za slijedeće cachee:

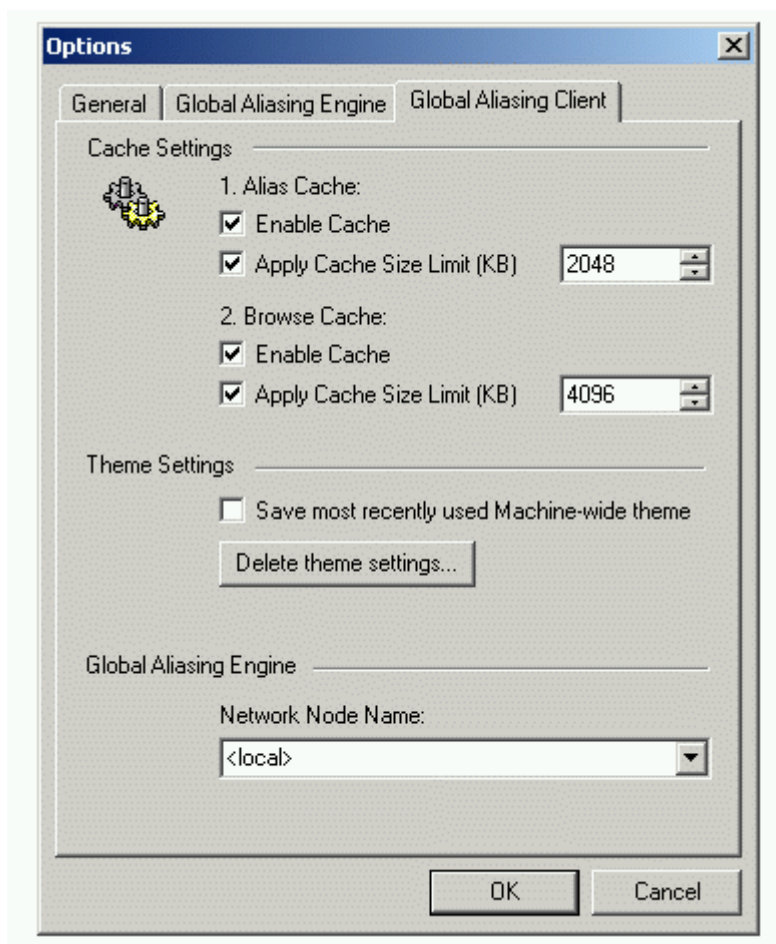
- **Alias cache** : cache koji se koristi da razluči globalne aliase za vrijeme runtimea
- **Browse cache** : cache koji se koristi za globalni alias browser.

Alias cache i browse cache su onemogućeni po defaultu, ali ih korisnik može individualno omogućiti. Kada je cache omogućen, može se također čekirati i **Apply cache Size limit** i specificirati njihova maksimalna veličina (u KB) za svaku od cache memorije.

Gledanje i testiranje konekcije sa bazom podataka

Setting za **Database connection** specificira tekuću MS Access (.mdb) ili MS Datalink (.udl) konfiguracionu bazu podataka za Globalni aliasing konfigurator.

Global aliasing client tab na **Options** dijalog boksu , koji je pokazan na narednoj slici, specificira slijedeće setinge za **Global aliasing client** (koji je lociran samo na lokalnoj mašini):



Globalni aliasing klijent ima ugradjen cache podataka koji privremeno pohranjuje globalne aliasing podatke za vrijeme runtimea. **Cache settings** specificira parametre za slijedeće cachee:

- **Alias cache:** cache koji se koristi da razluči globalne aliase za vrijeme runtimea
- **Browse cache:** cache koji se koristi za globalni alias browser.

Alias cache i browse cache su po defaultu onemogućeni, ali se može omogućiti svaki cache individualno, čekiranjem **Enable cache**, kako je pokazano na gornjoj slici.

Setovanje tema

Global aliasing teme su pohranjene u Global aliasing client cache. Kada se čekira **Save most recently used machine-wide theme** u **Theme Settings**, cache se pohranjuje u registre na mašinskim nivou (machine level).

Globalna aliasing mašina (global aliasing engine)

Global aliasing engine server je lociran po defaultu na lokalnom čvoru, ali može također biti lociran na udaljenom čvoru i da mu se pristupa preko intraneta ili interneta. Korisnik specificira lokaciju Global aliasing engine servera time što izabere čvor (node) iz otvarajuće liste pod **Network Node Name**.

Napomena: **Network node name** polje je raspoloživo samo onda kada **OPC Communication** se izabere u GenBroker konfiguratoru. Kada se izabere **GenBroker Communications** u GenBroker konfiguratoru, Global aliasing engine server koristi GenBroker konfiguracione setinge, i tada Network Node nam postaje nedostupan.

Karakteristike GraphWorrX32 paketa

GraphWorx32 je HMI (human – machine interface) softwareski paket za nadzor i vođenje procesa i integriranih funkcija procesno-poslovnih sistema. GraphWorx32 je u potpunosti saglasan sa OPC klijentom koji se bazira na ActiveX i OLE Automation tehnologijama.

Neke od karakteristika ovog softwareskog paketa su:

- podrška za Microsoft VBA 6.3
- Podrška za globalni aliasing
- Uključenje i integracija sa programom TraceWorx32 za debugiranje i dijagnostiku
- Priključivi (dockable) VBScript i Jscript editori
- Web publishing Wizard za izvoz i publikovanje displeja u HTML formatu
- Publikovanje višestrukih grafičkih stranica na WebServer
- Vrlo brzo (do 50 msec) ažuriranje ekrana za prednji plan i dinamičke linkove na procesnim ekranima
- Find and replace podrška za polja stanja (state fields)
- Find and replace podrška za lokalne aliases
- Jednostavni pristup ka tagovima simbola i aliasisa.
- Moćan editor za izraze
- AlarmWorx32 indikator ActiveX dodat na GraphWorx32 alatnu letvicu (toolbar).
- VBScript primjeri u biblioteci simbola
- Integrira podršku bazi podataka u GraphWorx32
- Browser (preglednik) za jezičke aliases
- Funkcije uvoza i izvoza ekspanidranih image fajlova.
- Kompresija slika
- Moćne funkcije zumiranja i paniranja u runtime ekranu.
- Globalno aliasiranje u konfiguracionom i runtime dijaloziima.
- VBA runtime događaji vezani za miša i tastaturu.
- Digitalni selektor, analogni selektor i animator
- Logiranje događaja (event log) za akcije i podatke
- Transparentnost i translucenost (providnost za boju)

- kompatibilnost sa OLEExpress 6.0
- Viewer za statistiku podataka u runtimu (GenStatistics)
- Podrška za višestruke kustom komande
- Visoki nivoi sigurnosti
- Slobodna rotacija objekata na ekranima
- Mogućnost konekcije sa izvorom podataka radi editiranja
- Podrška za izbor bilo kojeg interfejsnog jezika
- ActiveSync download u Windows CE i Pocket PC-jeve
- Pocket PC razvojni mod rada : podrška za Compaq , Casio i HP PDA
- Mogućnost real-time Thread prioriteta za threadove podataka i timera
- Mogućnosti doznačavanja Funkcionalnim tasterima i ostalim tasterima na tastaturi funkcija koje se pokreću tim tasterima
- Pop-up meniji koje definira korisnik
- Tipovi OPC request
- Konektivnost za OPC (OLE za upravljanje procesima)
- Moćni alati za kreiranje displeja
- Kompletan set crteža i animacionih alata u objektno orijentiranom okružaju. Displeji mogu biti skalabilni (automatski mjenjaju veličinu kada korisnik razviče ekran), ili sa fiksnom skalom (za kvalitetni prikaz slike sa pikselima , bez obzira koja je veličina prozora).
- GraphWorx32 je kontejner za ActiveX kontrolne elemente i OLE objekte.
- Mogućnost unošenja Active X kontrola iz ICONICS ili ActiveX kontrolnih elemenata drugih kreatora kao i OLE objekata , direktno u displeje.
- Server ActiveX dokumenata. GraphWorx32 je ActiveX dokument server, što znači da njegovi displeji se mogu izvršavati i u aplikacijama kao što je MS IE.
- Sveobuhvatni OLE Automation interfejs.
- Moćni set OLE Automation metoda i osobina za programsko manipuliranje GraphWorx32 displejima.
- Brza dinamička animacija
- GraphWorx32 ActiveX kontrola. GraphWorx32 uključuje ActiveX kontrolu ("GWXview32.OCX") koja je u stanju da izvršava GraphWorx32 prikaze. Ova komponenta ima sve runtime mogućnosti od "GraphWorx32.EXE", i može biti lako uronjena u ActiveX jezike, kao što su Visual Basic i HTML stranice.
- Napredna biblioteka simbola. GraphWorx32 uključuje utilitije za loading, pohranjivanje (storing) i organiziranje GraphWorx32 simbola. Jednostavno se može povući i spustiti (drag and drop) simbol u i iz biblioteka simbola.
- Saglasnost sa OPC 1.0 i OPC 2.0 DA (Data Access) standardima
- Podrška za više od 70 tipova animacija.
- Razvojna i konfiguraciona podrška za Windows CE
- Moćni slojevi prikaza kao kod AUTOCAD paketa sa clutter/declutter mogućnostima
- Gradijentno punjenje (gradient fill) kao kod PowerPointa
- Polja stanja (state fields) koja prikazuju čitljivi tekst koji predstavlja stanja mašine.
- Mogućnosti rotacije teksta u inkrementima od 90 stepeni.
- Web bazirani izgled i osjećaj (look and feel) sličan onome kao kod MS IE.
- Profesionalno crtanje 2D i 3D gradijenata u biblioteci simbola.
- Runtime window properties mode za konfiguriranje WYSIWYG.

- Moćne i napredne mogućnosti editiranja višestruke linije (polyline).
- Mogućnosti postavljanja passworda na displej fajlovima iustom simbolima koje je kreirao korisnik .
- Smart ActiveX alatna letvica (toolbar) koja dozvoljava korisniku da doda ili otkloni bilo koji ActiveX kontrolni element na ActiveX toolbaru.
- Kastomizirani toolbari.

Moćne funkcija rada sa VBA (Visual basic for Applications) , koje uključuju:

- Help Wizard. Daje detaljne informacije o korištenju VBA wizarda.
- OPC data access wizard (za pristup podacima). Setuje i dobiva OPC podatke
- MS Excel wizard. Integrira podatke u i iz Excela.
- Report wizard. Kreira jednostavne i korisne izvjestaje.
- Recipe wizard. Kreira recepture.
- MS Word wizard. Integriše podatke u MS Word.
- Email i Outlook wizard. Šalje email baziran na VBA događajima.
- Wizard OPC kalkulatora. Spaja kalkulator sa četiri funkcije sa tagom na bilo kojem displeju.
- OPC keypad wizard. Spaja tastaturu (Querty keypad) sa tagom na bilo kojem displeju.
- Runtime aliasing wizard. Mjenja aliases za vrijeme runtime moda.
- Analog chart wizard. Kreira prikaze sa analognim OPC tagovima
- Digital chart wizard. Kreira prikaze sa diskretnim OPC tagovima.

Dinamičke konekcije

Objekti na ekranima se animiraju i kontrolišu sa vrijednostima različitih tačaka podataka. Ove dinamičke konekcije sa objektima na displejima dolaze iz različitih OPC data servera kao izvora podataka. Korisnik može napraviti višestruke dinamičke konekcije sa jednim objektom na ekranu. Na primjer, može kreirati objekat čija se boja mjenja sa vrijednošću jedne konektovane tačke podatka a veličina sa drugom konektovanom tačkom.

Korisnik kreira dinamičke konekcije sa funkcijama koje su sadržane u meniju **Dynamics**. Tipovi dinamičkih konekcija uključuju flash, boju, procesnu tačku (PPT), unos podatka, veličinu, lokaciju, analogni i digitalni selektor, animator, sakrivanje objekta i njegovu rotaciju.

Konekcije sa podacima

Analiziraćemo interfejs izmedju GraphWorx32 objekata i ulazno/izlaznih podataka iz procesa. Razmotrićemo slijedeće:

- Kako napraviti konekcije sa procesnim podacima koristeći Tag Browser.
- Izraze

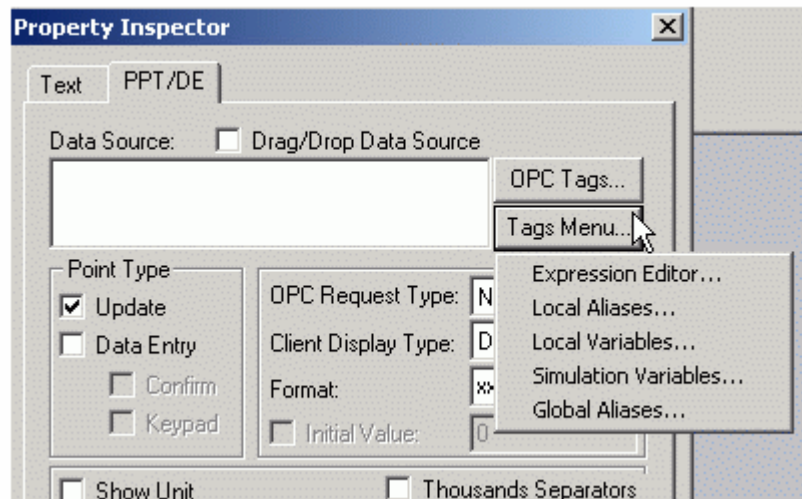
- Konstante i lokalne varijable
- Kreiranje generički višestruko iskoristivih prikaza i simbola sa aliasingom

Dinamička animacija se postiže sa transformisanjem vidljivih objekata na bazi specificiranih konekcija sa podacima. Svaki dinamički objekat dozvoljava jednu ili više konekcija za njegovu primarnu vrijednost(i). Neke dinamike takodjer dozvoljavaju i nadjahivanje opsega (range override), ovo su tada takodjer konekcije sa podacima. Konekcija sa podatkom je navjerovatnije tag koji predstavlja neku vrijednost u sistemu. Nadalje, konekcija sa podatkom može da bude i konstantna vrijednost, lokalna GraphWorx32 varijabla, ili matematski izraz (koji može uključiti tagove, konstante i lokalne varijable).

Konekcija sa podacima se može direktno ukucati kod editiranja izvora podatka u konfiguracionom dijalogu za različite dinamike. Tekst stringovi se takodjer mogu prevući i ispustiti (drag and drop) u ovo editorsko polje iz bilo kojeg drag izvora, koji čini drag and drop podatke raspoloživim u Windows tekst clipboard formatu. Postoje takodjer i dva tastera:

OPC Tags taster – koji otvara **Tag Browser**

Tags Menu - koji prikazuje meni sa slijedećim opcijama:
Expression Editor, Aliases, Local Variables, Simulation Variables, i Global Aliases, kako je prikazano na slijedećoj slici:

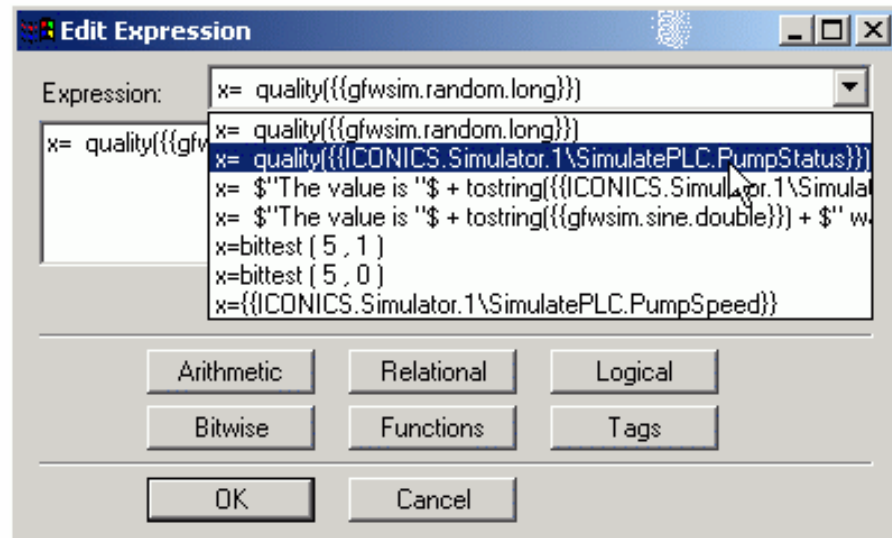


OPC tagovi i Tag meni tasteri u Property Inspectoru za objekat

Izabirući Expression Editor , iz Tags meni otvara Edit Expression dijalog boks, koji je pokazan na narednoj slici. Kategorije funkcionalnosti koje su na raspolaganju za izraze uključuju:

- aritmetičke
- relacione
- logičke

- bit orijentirane
- funkcije



Dijalog boks za Edit Expression

Expression je string koji definira i evaluira konekciju sa podatkom izmedju klijenta i OPC servera. Za vrijeme runtime moda, OPC serveri razlučuju vrijednost podatka za izraz. Da se indicira da je konekcija sa podatkom izraz, treba početi string sa " x=" tokenom, kao što je pokazano u narednom tekstu :

`x={{ICONICS.Simulator.1\SimulatePLC.PumpSpeed}}`

Korisnik može da ili direktno ukucava izraze u tekst boks u okviru dijalog boksa **Edit Expression**, ili može koristiti simbole i funkcije koji su na raspolaganju i koji mu pomažu da koristi propisanu sintaksu stringova kada bude unosio izraze.

Stringovi u izrazima

Izrazi omogućavaju da se izračunavanja izvršavaju na dolazećim podacima. OPC server može obezbjediti podatke u jednom ili više tipova podataka, kao što su:

"float", "long", "integer", "string", itd.

Ako numerički podatci dolaze sa servera kao stringovi, oni se porede kao stringovi u izrazima. Ovo je uradjeno na bazi alfabetskog redoslijeda slova. Zbog toga, izraz evaluiran kao TRUE "20" > "100" , je korektan. Naravno, ako želimo da provedemo numeričku komparaciju, 20< 100 i gornji izraz će izgledati nekorektan, mada to nije slučaj.

Ako želimo da se provede numerička komparacija, onda treba dodati 0 svakom od tagova, da bi logički operatori radili korektno. Na primjer:

`x={{JC.N1OPC.1.0\HDQTRS\sys2\ad-3.Present Value}}+0) > ({{JC.N1OPC.1.0\HDQTRS\sys2\ad-4.Present Value}}+0)`

Alternativni način bi bio da promjenimo OPC server tako da on šalje stringove sa fiksnim brojem digita sa vodećim nulama, ili da koristimo GraphWorx32 registre za konverziju iz stringa u broj.

Sintaksa sa proširenom tačkom (PES)

Point extension syntax (PES) – sintaksa sa proširenom tačkom, omogućava da dobijemo dodatne informacije koje su vezane sa OPC tagovima, kao što je kvalitet (quality) i vrijeme i datum očitavanja (timestamp). Slijedeća sintaksa se koristi u izrazima za validan PES zahtjev:

- tag:ICONICS.Simulator\SimulatePLC.Ramp#timestamp
- tag:ICONICS.Simulator\SimulatePLC.Ramp#quality

iz lokalne mašine

i

- tag:\\pc1\ICONICS.Simulator\SimulatePLC.Ramp#timestamp
- tag:\\pc1\ICONICS.Simulator\SimulatePLC.Ramp#quality

iz noda pc1 u mreži.

Ponekada može biti potrebno da prisilimo "request data type", za specifični tip podatka, kao što je "string", da bi se prikazala ova informacija u procesnoj tački (PPT).

OPC tagovi

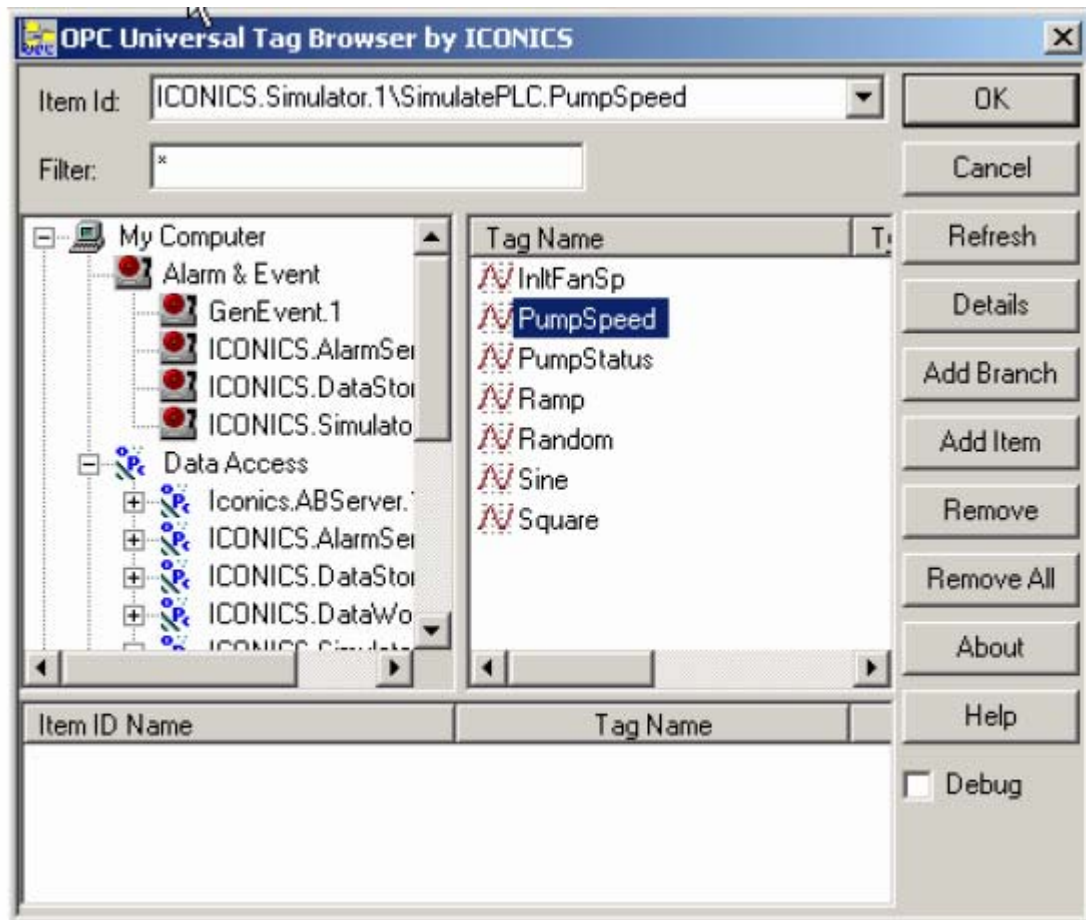
OPC Tag ili tačka podatka (data point) , je konekcija sa podatkom izmedju klijenta i OPC servera. OPC tagovi se mogu koristiti u izrazima kada je tag uronjen (embedded) izmedju duplih zagrada , kao u slijedećem primjeru :

{{tag_name}}

naprimjer :

x={{ICONICS.Simulator.1\SimulatePLC.PumpSpeed}}

Korisnik može koristiti Tag Browser, pokazan na narednoj slici, da izabere OPC Alarm and Event – alarm i dogadjaj (AE), Data Access – pristup podatku (DA), ili Historical Data Access – pristup historijskom podatku (HDA), tip taga, da uključi u njegove izraze:



Aliasi

Alias je string koji predstavlja ili opisuje objekat ili tačku podatka na displeju. U izrazima se mogu koristiti i lokalni i globalni aliasi.

Lokalni aliasi

Za lokalni alias unutar izraza, koristi se slijedeća sintaksa:

<<local_alias_name>>

Naprimjer:

x= <<TankLevel>>

Globalni aliasi

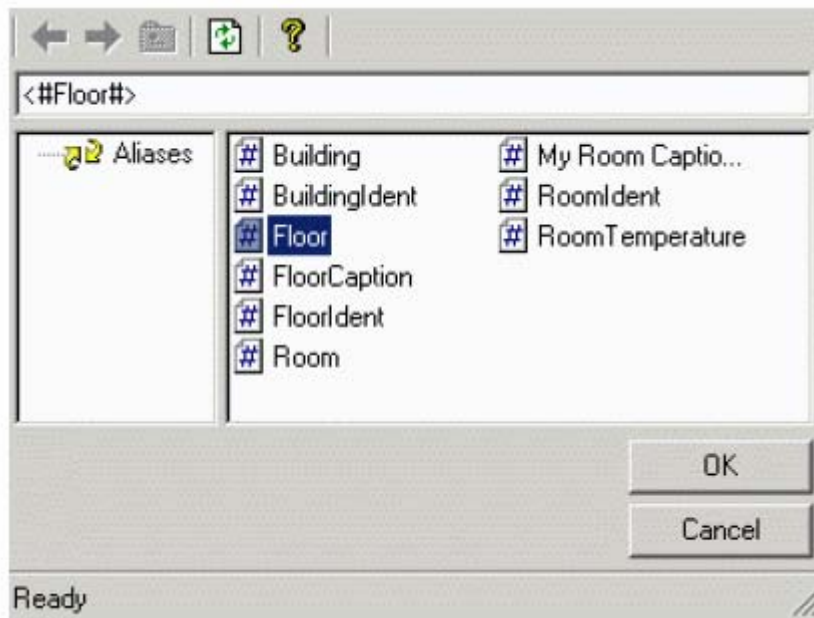
Za globalni alias unutar izraza , koristiti slijedeću sintaksu:

<#global_alias_name#>

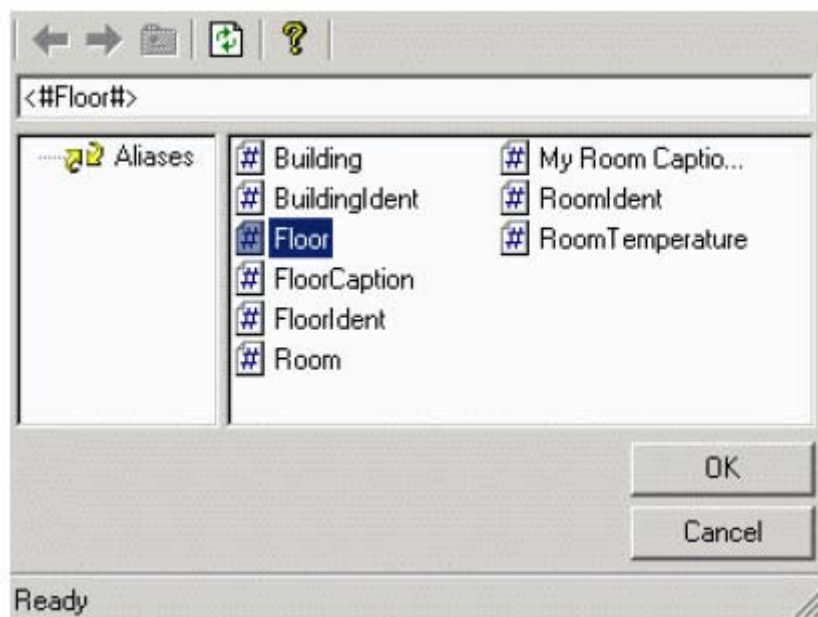
Naprimjer:

x = <#RoomTemperature#>

Ako izaberemo **Global Alias Browser** otvoriće se taj browser, kako je pokazano na narednoj slici.



Izabrati global alias iz ovog browsera, koji uključuje sve globalne aliase u globalnoj alias bazi podataka. Ovo eliminira potrebu da ručno upisujemo ime aliasa. Svi globalni aliasi koji su konfigurirani u Global alias engine konfiguratoru su rapoloživi da se mogu izabrati unutar browsera. Kontrola stabla global alias engine konfiguratora je oponašana i u upravljanju stablom u globalnom alias browseru. Izabrati global alias dvaputa kliknuvši ime aliasa (napr. "Floor" na narednoj slici). Alias ime će se pojaviti na vrhu browsera, koji automatski dodaje <# i #> rezdjelivače (delimiters) na alias ime.



Izbor aliasa iz global alias browsera

Jezički aliasi

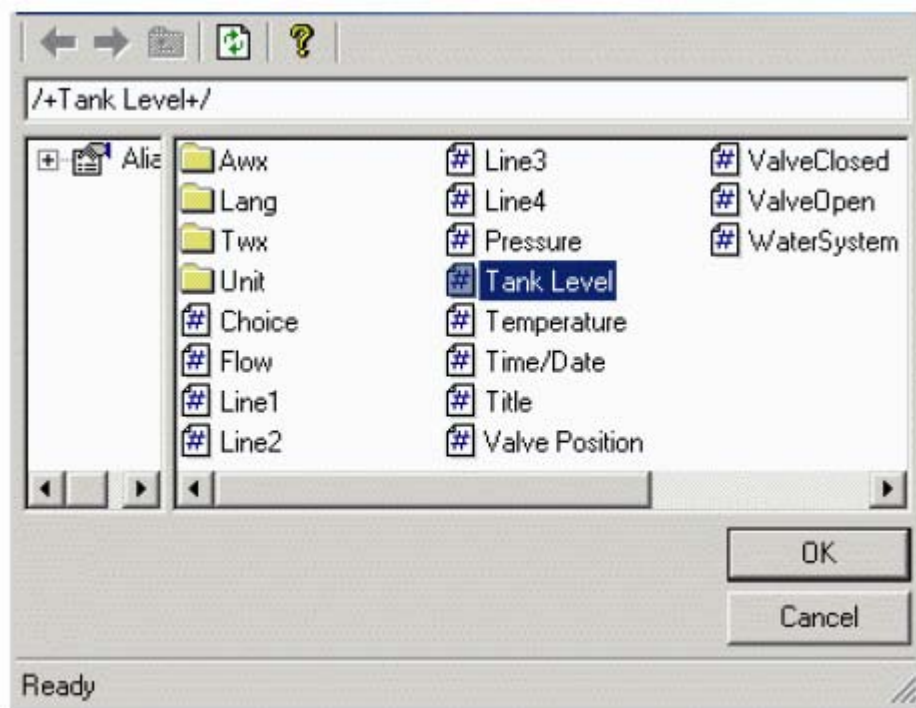
Za jezičke aliase unutar izraza, treba koristiti slijedeću sintaksu:

`/+language_alias:name+/`

Naprimjer:

`x= /+WaterSystem+/`

Izbor **Language Alias Browser**-a iz pop-up menija otvara browser aliasa za jezik (interfejsni jezik između Operatora i HMI sistema), kao što je pokazano na narednoj slici. Browser uključuje sve jezičke aliase u bazi podataka interfejsnih jezika koji stoje na raspolaganju. Svi jezički aliasi koji su konfigurirani u Language Configurator (jezičkom konfiguratoru), su na raspolaganju da se mogu izabrati unutar browsera. Kontrola stabla jezičkog konfiguratora je oponašana u kontroli stabla jezičkog alias browsera. Izabrati jezički alias, kliknuvši dva puta na ime aliasa. Ime aliasa se pojavljuje na vrhu browsera, koji automatski dodaje `/+` i `/+` razdjelivače na ime aliasa.



Izbor alisa iz browsera jezičkih aliasa

Varijable

Varijable se mogu koristiti u izrazima. Kako varijabla treba da bude referencirana zavisi o tipu varijable. Lokalna varijabla se može koristiti u izrazima kada je varijabla umetnuta između dvostrukog znaka `~` (tilde, napr: `~~variable~~`).

Lokalne varijable

Za lokalne varijable unutar izraza , treba koristiti slijedeću sintaksu:

~~local_variable_name~~

Primjer:

x= ~~Setpoint~~

Simulacione varijable

Za simulacione varijable unutar izraza, koristiti slijedeću sintaksu:

}}gfwsim.random.long}}

Aritmetički izrazi

Simboli '+', '-', '*' i '%', koriste slijedeći format:

expression::parameter **symbol** parameter

gdje:

Parameter	Lokalna varijabla, OPC tag, ili neki drugi izraz
Symbol	+ ili - ili / ili %

Rezultat

Izraz rezultira u broju bilo kojeg tipa (float, long, byte, itd.)

Primjeri

Simbol	Opis	Primjer	Rezultat
+	Sabiranje	~~var1~~ + ~~var2~~	8 + 3 =11
-	Oduzimanje	~~var1~~ - ~~var2~~	8-3=5
*	množenje	~~var1~~ * ~~var2~~	8 *3 =24
/	djeljenje	~~var1~~ / ~~var2~~	8/3=2.66667
%	Izračunava ostatak djeljenja	~~var1~~ %~~var2~~	8%3=2
(and)	Daje prednost u računanju djelova izraza	~~var1~~ /(~~var2~~ + ~~var3~~)	8/(3+2)=1.6

Relacioni izrazi

Simboli '<', '>', '<=', '>=', '==' i '!=' koriste slijedeći format:

expression::parameter **symbol** parameter

gdje:

Parameter	Lokalna varijabla, OPC tag, ili neki drugi izraz
Symbol	< ili > ili <= ili >= ili == ili !=

Rezultat

Izraz rezultira u Boolovoj vrijednosti (0 ili 1).

Primjeri

Simbol	Opis	Primjer	Rezultat
<	Manje od	~~var1~~ < ~~var2~~	8 < 3 =0
>	Veće od	~~var1~~ > ~~var2~~	8>3=1
<=	Manje ili jednako	~~var1~~ <= ~~var2~~	8 <=3 =0
>=	Veće ili jednako	~~var1~~ >=~~var2~~	8>=3=1
==	Jednako sa	~~var1~~ ==~~var2~~	8==3=0
!=	Nije jednako sa	~~var1~~ !=~~var2~~	8!=3=1

Logički izrazi

Simboli '&&' i '||' koriste slijedeći format:

expression::parameter **symbol** parameter

Simbol '!' koristi slijedeći format:

expression:: **symbol** parameter

gdje:

Parameter	Lokalna varijabla, OPC tag, ili neki drugi izraz
Symbol	<&& ili ili !

Rezultat

Izraz rezultira u Boolovoj vrijednosti (0 ili 1)

Tabela istine

~~var1~~	0		not 0	
	0	not 0	0	not 0
~~var1~~ && ~~var2~~	0	0	0	1
~~var1~~ ~~var2~~	0	1	1	1
!~~var1~~	1	1	0	0

Primjeri

Simbol	Opis	Primjer	Rezultat
&&	I (and)	~~var1~~ && ~~var2~~	8 && 3 =1
	Ili (or)	~~var1~~ ~~var2~~	8 3=1
!	not	~~var1~~ <= ~~var2~~	8 <=3 =0

Bit orijentirani operatori (bitwise)

Simboli '&', '|' i '^' iz grupe bit orijentiranih operatora koriste slijedeći format:

expression::parameter **symbol** parameter

Simbol '~' pak koristi slijedeći format:

expression:: **symbol** parameter

Simboli 'shl' i 'shr' iz ove bit orijentirane grupe, koriste slijedeći format:

expression:: **symbol** (value,shift by)

gdje

Parameter	Lokalna varijabla, OPC tag, ili neki drugi izraz
Symbol	&& ili ili ^ ili shl ili shr ili ~

Izraz rezultira u broju ako parametri koji se koriste sadrže brojeve

Bit tabela

	Binary (Decimal)	Binary (Decimal)
~~var1~~	0000.0000.0000.1000 (8)	0000.0000.0110.0000 (96)
~~var2~~	0000.0000.0000.1010 (10)	0000.0000.0000.1000 (8)
~~var1~~ & ~~var2~~	0000.0000.0000.1000 (8)	0000.0000.0000.0000 (0)
~~var1~~ ~~var2~~	0000.0000.0000.1010 (10)	0000.0000.0110.1000 (104)
~~var1~~ ^ ~~var2~~	0000.0000.0000.0010 (2)	0000.0000.0110.1000 (104)
shl (~~var1~~,3)	0000.0000.0100.0000 (64)	0000.0011.0000.0000 (768)
shr (~~var1~~,3)	0000.0000.0000.0001 (1)	0000.0000.0000.1100 (12)
~(~~var1~~)	1111.1111.1111.0111 (-9)	1111.1100.1111.1111 (-97)
bittest (~~var1~~,3)	0000.0000.0000.0001 (1)	0000.0000.0000.0000 (0)

Primjeri

Simbol	Opis	Primjer	Rezultat
&	Bit and	~~var1~~ & ~~var2~~	8 & 3 = 0
	Bit Or	~~var1~~ ~~var2~~	8 3 = 11
^	Bit exclusive Or	~~var1~~ ^ ~~var2~~	8 ^ 3 = 11
shl	Bit shift lijevo	shl(~~var1~~,3)	8 << 3 = 64
shr	Bit shift desno	shr(~~var1~~,3)	8 >> 3 = 1
~	Not (dva kompliment)	~ (~~var1~~)	!8 = -9
bittest	Test bita		

Opaska:

Bittest funkcija zahtjeva da specificiramo poziciju bita koji će biti testiran. Moramo naznačiti da starta od 0. Drugim riječima, pozicija bita "0" indicira 'najmanje značajni bit.

Funkcije

Simboli 'sin', 'asin', 'cos', 'acos', 'tan', 'atan', 'log', 'ln', 'exp', 'sqrt', 'abs', 'ceil', i 'floor' koriste slijedeći format:

expression:: **symbol** (parameter)

Simboli 'pow', 'min', i 'max', koriste slijedeći format:

expression:: **symbol**(parameter, parameter)

Simbol 'if' koristi slijedeći format:

expression:: **symbol**(parameter, parameter, parameter)

gdje:

Parameter	Lokalna varijabla, OPC tag, ili neki drugi izraz
Symbol	Sin, asin, cos, acos, tan, atan, log, ln, exp, sqrt, abs, ceil, floor, min, max, pow, ili if

Rezultat

Izraz rezultira u broju:

Primjeri

Simbol	Opis	Primjer	Rezultat
sin	Sinus ugla u radijanima	sin(~var1~)	sin(0.785)=0.71
cos	cosinus ugla u radijanima	cos(~var1~)	cos(0.785)=0.71
tan	tangens ugla u radijanima	tan(~var1~)	tan(0.785)=0.52
asin	arc sinus vraća ugao u radijanima	asin(~var1~)	asin(0.5)=0.52
acos	cosinus vraća ugao u radijanima	acos(~var1~)	acos(0.5)=1.05
atan	tangent vraća ugao u radijanima	atan (~var1~)	atan(1)=1.05
sqrt	vraća kvadratni korijen	sqrt(~var1~)	sqrt(100)=10
pow	vraća vrijednost 1 na potenciju od vrijedosti2	pow(~var1~,~var2~)	pow(100,1.5)=1000
log	log na bazi 10	log(~var1~)	log(100)=2

ln	logaritam na bazu e	ln(~~var1~~)	ln(7.389)=2
exp	exponent	exp(~~var1~~)	exp(2)=7.389
abs	absolutna vrijednost	abs(~~var1~~)	abs(-1) =1
ceil	integer ceiling	ceil(~~var1~~)	ceil(7.39)=8
floor	floor integer	floor(~~var1~~)	floor(7.39)=7
min	manja vrijednost od dvije	min(~~var1~~,~~var2~~)	min(10,5)=5
max	veća vrijednost od dvije	max(~~var1~~,~~var2~~)	max(10,5)=10
if	uslovni iskaz	if(~~var1~~<~~var2~~,~~var1~~,~~var2~~)	if(5<8,5,8)=5
like	poredjenje stringa sa wildcard	like(string,pattern, case sensitive)	
quality	kvalitet taga ili izraza	vidjeti dalje u tekstu	vidjeti dalje u tekstu
tostring	konverzija tipa	vidjeti dalje u tekstu	vidjeti dalje u tekstu
0x	hexadecimalna konstanta	x=0x11	17 (decimalno)
0t	Oktalna konstanta	x=0t11	9
0b	binarna konstanta	x=0b11	3

Opaska:

Za **like** operator : "string" je jednak stringu za traženje , "pattern" je jednak stringu za kojim se traga (može uključiti i wildcards): ne-nula za traženje koje je osjetljivo na velika i mala slova, nula za traženje koje nije osjetljivo na mala i velika slova. Sintaksa stringa je "\$"string"\$"

Kvalitet

Opcija **quality** u **Functions** meniju **Expression Editor**a se koristi da se evaulira kvalitet OPC taga ili jednog izraza.

Koristi se slijedeća opšta sintaksa za izraze za kvalitet:

x= quality (expression)

Opaska:

Expression , može takodjer biti jednostavan izraz sastavljen od samo jednog taga

Funkcija quality vraća kvalitet OPC taga kao string između zagrada , i to kao jedan od sljedećih rezultata:

- 192 : kvalitet je dobar (GOOD)
- 64 : kvalitet je nesiguran (UNCERTAIN)
- 0 : kvalitet je loš (BAD)

Primjer izraza za kvalitet

Izraz	Rezultat
x=quality({{ICONICS.Simulator.1\SimulatePLC.PumpStatus}})	192 (kvalitet GOOD)

Kvalitet izraza je određen kroz evaluaciju svakog pojedinačnog taga u izrazu. Dakle, ako imamo višestruke tagove u izrazu, (i svaki tag ima različit kvalitet), rezultat izraza (tj. 192 GOOD , 64 (BAD) ili 0 (UNCERTAIN), odgovara kvalitetu taga sa najmanjim kvalitetom. Ako izraz sadrži uslovni iskaz (napr. if, then, ili else), tada rezultat izraza je samo afektiran sa kvalitetom grane koja se izvršava.

Posmatrajmo sljedeći izraz:

x= if (quality({{Tag1}}) == 192, {{Tag1}}, {{Tag2}})

Izraz se može iščitati kao:

“Ako je kvalitet Tag1 GOOD (tj. 192) tada izraz rezultata x je vrijednost Tag1. U svim drugim slučajevima (tj. ako kvalitet Tag1 je UNCERTAIN ili BAD), rezultat izraza x je vrijednost Tag2”.

Mo možemo izračunati rezultate za ovaj izraz koristeći različite kvalitete za Tag1 i tag2, kao što je pokazano u sljedećoj tabeli:

Case	Tag1 quality	Tag2 quality	Result	Result quality
1	GOOD	GOOD	Tag1	192 (GOOD)
2	GOOD	UNCERTAIN	Tag1	192 (GOOD)
3	GOOD	BAD	Tag1	192 (GOOD)
4	UNCERTAIN	GOOD	Tag2	192 (GOOD)
5	UNCERTAIN	UNCERTAIN	Tag2	64 (UNCERTAIN)
6	UNCERTAIN	BAD	Tag2	0 (BAD)
7	BAD	GOOD	Tag2	192 (GOOD)
8	BAD	UNCERTAIN	Tag2	64 (UNCERTAIN)
9	BAD	BAD	Tag2	0 (BAD)

U slučajevima 1-3 u gornjoj tabeli, kvalitet Tag1 je GOOD, i zbog toga rezultat izraza je GOOD. Dakle, rezultat izraza nije afektiran sa kvalitetom Tag2, (u “else” grani izraza), koji se ignorira.

U slučajevima 4-6, kvalitet Tag1 je UNCERTAIN, i zbog toga rezultat izraza je kvalitet Tag2.

U slučajevima 7-9, kvalitet Tag1 je BAD, i zbog toga rezultat izraza je kvalitet Tag2.

Opaska: Funkcija "quality()" vraća vrijednost koja predstavlja kvalitet izraza unutar zagrada ali je uvijek GOOD_QUALITY. Na primjer, ako Tag1 je BAD_QUALITY tada izraz "x=quality ({{Tag1}})" će vratiti 0 sa GOOD_QUALITY.

Rezultat izraza je minimum kvaliteta evaluiranog taga u izrazu i afektiran je samo sa kvalitetom uslovne (if, then, ili else) grane koja se izvršava.

Posmatrajmo slijedeći jednostavan primjer:

x= if ({{TAG_01}}>0,{{TAG_02}},{{TAG_03}})

Ovaj izraz se može interpretirati kako slijedi:

" Ako vrijednost TAG_01 je veća od 0, tada rezultat x izraza je TAG_02. Ako je vrijednost TAG_01 manja ili jednaka 0, tada rezultat izraza x je TAG_03.

Predpostavimo da su slijedeće vrijednosti i kvaliteta za ove tagove:

TAG_01=5 sa kvalitetom GOOD
TAG_02=6 sa kvalitetom UNCERTAIN
TAG_03=7 sa kvalitetom BAD

Pošto je vrijednost TAG_01 jednaka 5 (tj. veća od 0), rezultat izraza je TAG_02. Dakle, konačni rezultat izraza je 6, i kvalitet finalnog izraza je UNCERTAIN.

Konverzija tipa varijable

Opcija **tostring** na **Functions** meniju u **Expression Editoru** uzima vrijednost bilo kojeg parametra u zagradama i konvertuje ga u string kako slijedi:

Vrijednost je +(value)+ unit

Može se koristiti da konvertuje iz broja u string, i može biti vrlo korisno za konkatenciji (sastavljanju) stringova.

Korektna sintaksa za opciju **tostring** je :

x="\$The value is "\$ + tostring(value) + "\$ unit"\$

Opaska:

U gornjem izrazu, riječ "unit" je tekst za držanje mjesta za mjernu jedinicu koju specificira korisnik, od mjerne veličine ili varijable (napr. Watt, mm, m, itd)

Primjer izraza za konverziju

Izraz	Rezultat
x="\$The value is "\$ + tostring({{gfwsim.ramp.float}}) + "\$ Watt"\$	Vrijednost je 543.2345152

Konstante

Functions meni od **Expressions Editora** podržava konstantne vrijednosti, uključujući hexadecimalne, oktalne, i binarne formate

Primjeri izraza koji koriste konstante

Expression	Result
x=0x11	17
x=0t11	9
x=0b11	3

Expression Editor pogodno umeće 0x i 0t i 0b prefikse sam , tako da korisnik ne mora da ih pamti i unosi.

Interpretacija i prevodjenje konstanti

Slijedeći primjer pokazuje kako se izračunavaju vrijednosti za svaki tip konstante:

- **hexadecimalna** : $0x20A = 2 * (16^2) + 0 * (16^1) + 10 * (16^0) = 2 * 256 + 0 * 16 + 10 * 1 = 512 + 0 + 10 = 522$
- **Octalna** : $0t36 = 3 * (7^1) + 6 * (7^0) = 3 * 7 + 6 * 1 = 21 + 6 = 27$
- **Binarna**: $0b110 = 1 * (2^2) + 1 * (2^1) + 0 * (2^0) = 1 * 4 + 1 * 2 + 0 * 1 = 4 + 2 + 0 = 6$

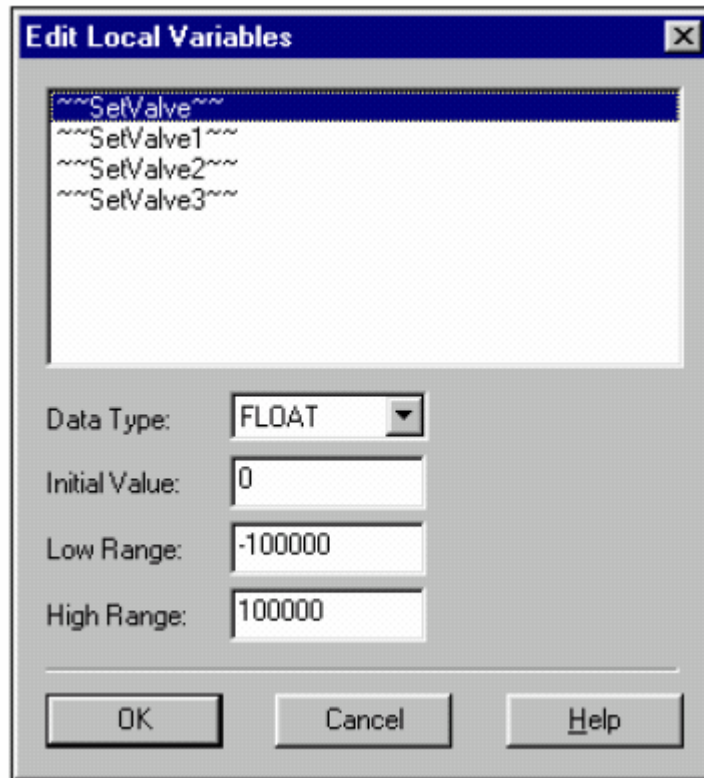
Lokalne GraphWorX32 varijable

Da indiciramo da je konekcija sa podatkom lokalna varijabla, treba koristiti slijedeću sintaksu:

~~local_variable_name~~

Izabirući **Edit Local Variables** iz **Dynamics** menija otvara **Edit Local Variables** dijalog boks koji je pokazan na slijedećoj slici:

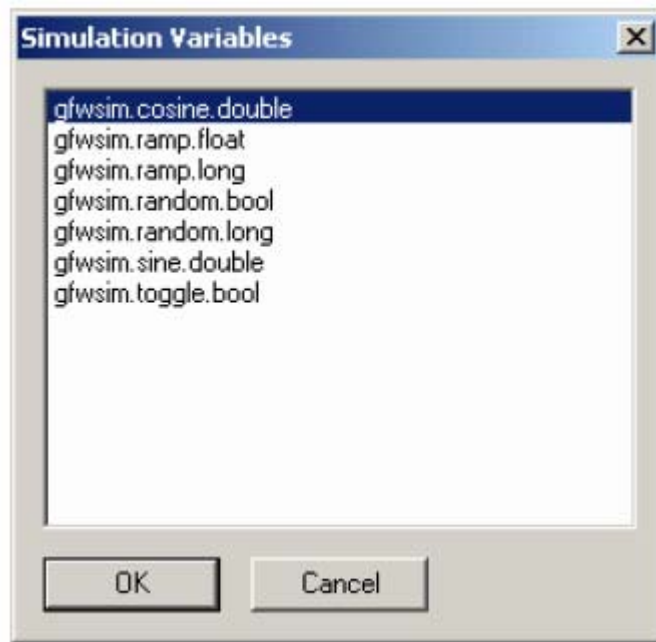
Na njemu možemo promjeniti default setinge za lokalne varijable:



Editiranje parametara lokalnih varijabli

parametar	Opis
Tip podatka (data type)	Specificira tip podatka lokalne varijable
Početna vrijednost (initial value)	Postavlja početnu vrijednost lokalne varijable
Gornja i donja vrijednost opsega (high range and low range)	Postavlja opseg vrijednosti za lokalnu varijablu

Opseg lokalnih GraphWorX32 varijabli je ograničen na displej u kojem su definirane. Tipično, lokalne varijable bi se koristile kao vrijednosti za animacione efekte (tj. simulirane vrijednosti koje ne trebaju da budu definirane na svim nivoima sistema). GraphWorX32 uključuje nekoliko prethodno definiranih lokalnih simulacionih varijabli. Korisnik može pristupiti listi ovih varijabli klikanjem na **Tags Menu** i izabirući **Simulation Variables** dok je u Property Inspectoru. Ovo otvara **Simulation Variables Dialog** , pokazan na narednoj slici, koji izlistava sve raspoložive simulacione varijable. Ove simulacione varijable su korisne za testiranje displej animacija kada OPC server nije raspoloživ.



GraphWorX32 simulacione varijable

Konstantne vrijednosti

GraphWorX32 automatski prepoznaje ako je uneseni string broj, i interpretira taj podatak konekcije kao konstantnu vrijednost. Ako želimo da unesemo konstantnu string vrijednost (tj. ne numeričku konstantu), treba koristiti slijedeću sintaksu:

`$"constant_string_value"$`

Ova sintaksa spriječava da string bude detektovan kao ime taga.

Aliasing konekcije sa podacima

Aliasing je mogućnost da definiramo string (obično neko kratko ime) da predstavlja neki drugi string (obično dio ili puno ime taga). Korisnik može pristupiti komandi **Edit Aliases** na meniju **Dynamics**. Možemo također pristupiti tome i desno kliknuvši na objekat u displeju i izabirući **Edit Aliases** iz pop-up menija.

Kada uključujemo alias ime u konekciju sa podatkom, alias ime treba da bude okruženo sa specijalnim razdjelnicima "<<" i ">>". Ovi razdjelnici se koriste da identificiraju početak i kraj alias imena.

Naprimjer:

`[opcserver].<<tank>>.out.`

U ovom slučaju "tank" je alias ime.

Aliasing na drugom nivou dozvoljava da se alias ime unese kao alias definicija. Ovaj drugi alias treba da bude razriješen prije nego što se izvrši set-aliases komanda.

Aliasing na objektnom nivou

Objekti koji koriste konekcije sa podacima mogu održavati svoje vlastite alias tabele. Svaki dinamički objekat može imati jednu alias tabelu. Ova alias tabela se automatski kreira kada unesemo string izvora podatka koji sadrži alias imena (tabela se automatski popunjava sa ulazima za svako alias ime koje se koristi u konekciji sa podacima dinamičkog objekta). Po defaultu, alias zamjenski string se postavlja da je jednak alias imenu. Ako je alias ime jednako alias zamjenskom stringu, tada alias neće biti razriješen. Korisniku je dozvoljeno da promjeni zamjenske stringove u alias tabelama na objektnom nivou. Alias imena u tabeli se kontroliraju sa alias imenima koja se koriste u stringovima izvora podataka.

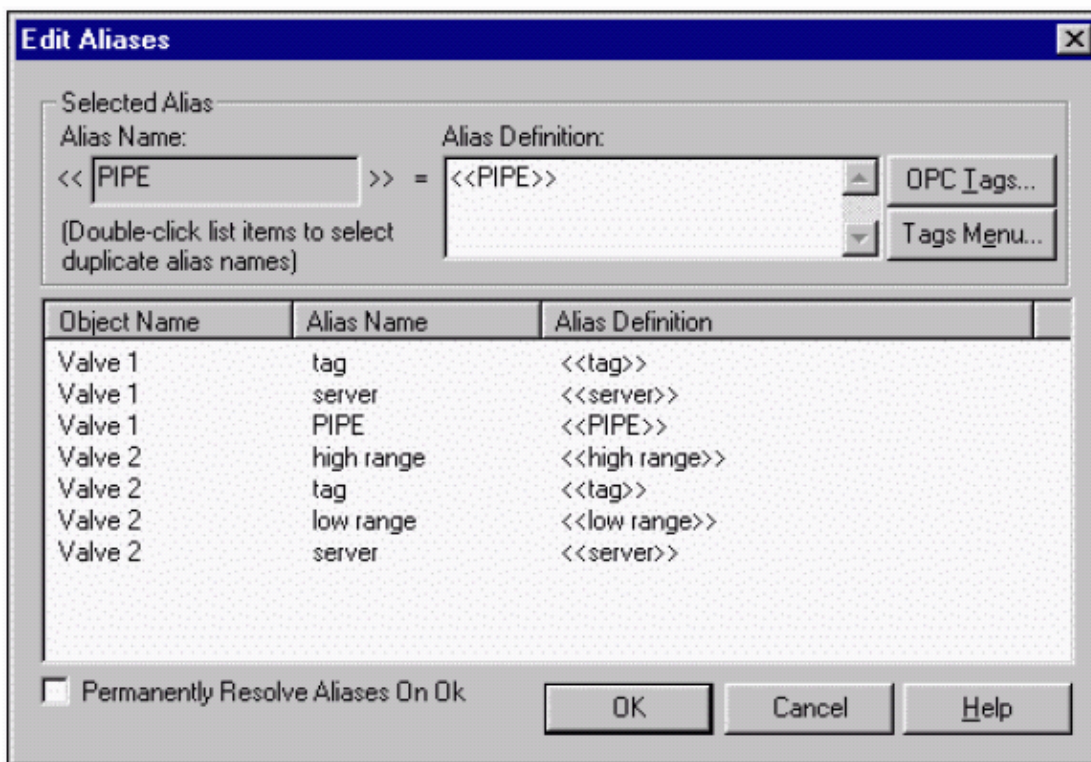
Da bi se editirale aliase objektnog nivoa u GraphWorX32, treba izabrati jedan ili više vidljivih objekata i zatim izabrati **Edit Aliases** iz menija **Dynamics**. Ovo otvara **Edit Aliases** dijalog boks. Sve alias tabele za sve dinamike priključene na sve selektirane objekte se objedinjuju i prikazuju **Edit Aliases** dijalog boks. Ako selektirani objekti imaju definirana imena objekata, ova imena se koriste da identificiraju koja alias imena pripadaju kojim objektima. Ako objekti nisu izabrani kada korisnik izabere da editira aliase, svi aliasi na cjelokupnom dispelju se pojavljuju u **Edit Aliases** dijalog boks.

Editiranje aliasa

Da bi se otvorio **Edit Aliases** dijalog boks, koji je pokazan na narednoj slici, treba izabrati Edit Aliases iz **Dynamics** menija. Ovo omogućava korisniku da modificira tabele aliasa objektnog nivoa i da promjeni alias definicije za svaki ulaz alias imena.

Višestruki detalji u kontrolnoj listi se mogu izabrati simultano da omogući višestruke ulaze da se mogu odjedanput svi modifikovati. Dvostruko kliknuvši na bilo koje alias ime automatski izabire sva dupla alias imena u listi.

Da se sortira lista aliasa po odredjenoj kategoriji, treba jednostavno kliknuti na naslov kolone i lista će se urediti u skladu sa prioritetima u toj koloni.



Dijalog boks za editiranje aliasa

Parametri editora aliasa

Parametar	Opis
Alias definition	Mjenja definicije za sve izabrana alias imena
OPC Tags taster	Lansira OPC Universal Tag Browser
Tags Menu taster	Prikazuje meni sa slijedećim opcijama: Expression Editor, Local Variables, Simulation Variables, i Global Aliases
Permanently Resolve Aliases on OK – ček boks	Alias su normalno razriješeni kada ulazimo u runtime mod (alias imena se restauriraju kada ponovo se vraćamo u konfiguracioni mod rada). Čekiranjem ovog boksa razriješavamo aliase u konfiguracionom modu (i time trajno zamjenjujemo originalne aliase)

Opaska: Stringovi za zamjenu aliasa se mogu modificirati za vrijeme runtime moda putem OLE Automation-a.

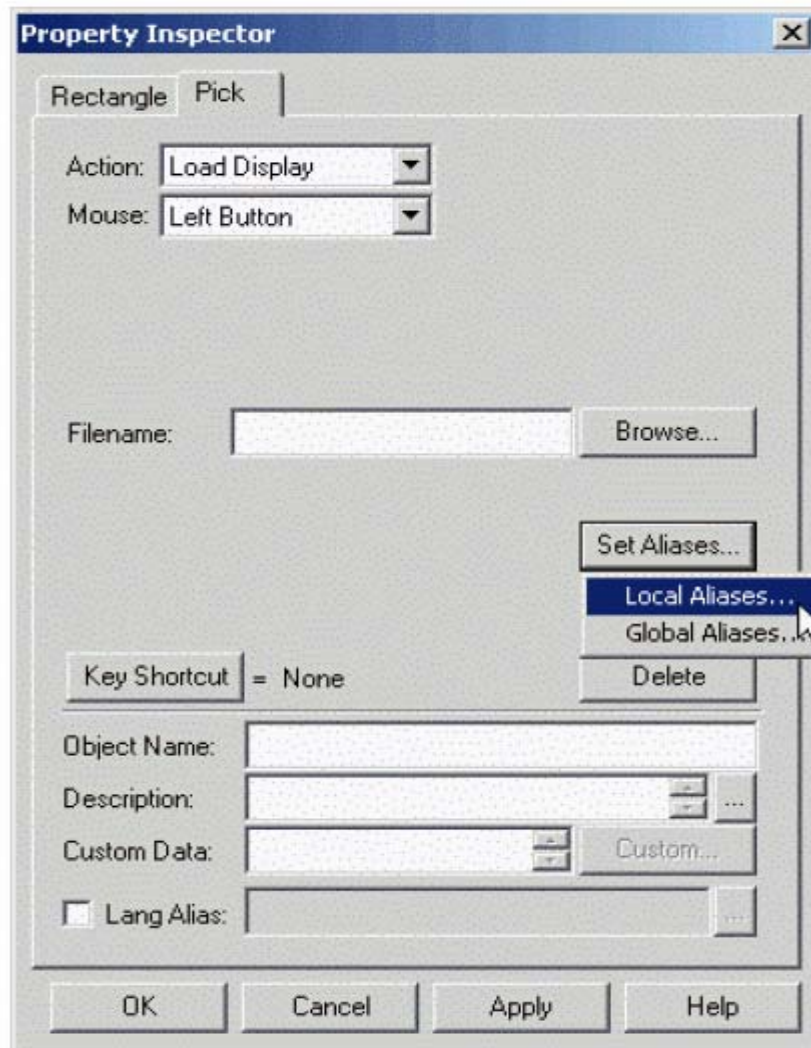
Runtime aliasing

Runtime aliasing obuhvata niz načina promjene aliasa na displeju za vrijeme runtimea. Korisnik može specificirati kolekcije aliasa da se postave kao dio date komande. Na primjer, kada koristimo pick akciju da napunimo displej na ekran, možemo specificirati aliase da se setuju u displeju koji treba upravo da se napuni na ekran (load). Ovi aliasi koji će biti setovani mogu biti specificirani direktno u GraphWorX32 ili mogu doći iz nekog vanjskog tekst fajla sa tab delimiterima (ovakvi fajlovi se mogu izvesti iz Excela, biti kreirani u Notepadu, ili kreirani koristeći GraphWorX32 Alias Fajl Editor).

Komande koje se mogu pridružiti aliasima uključuju:

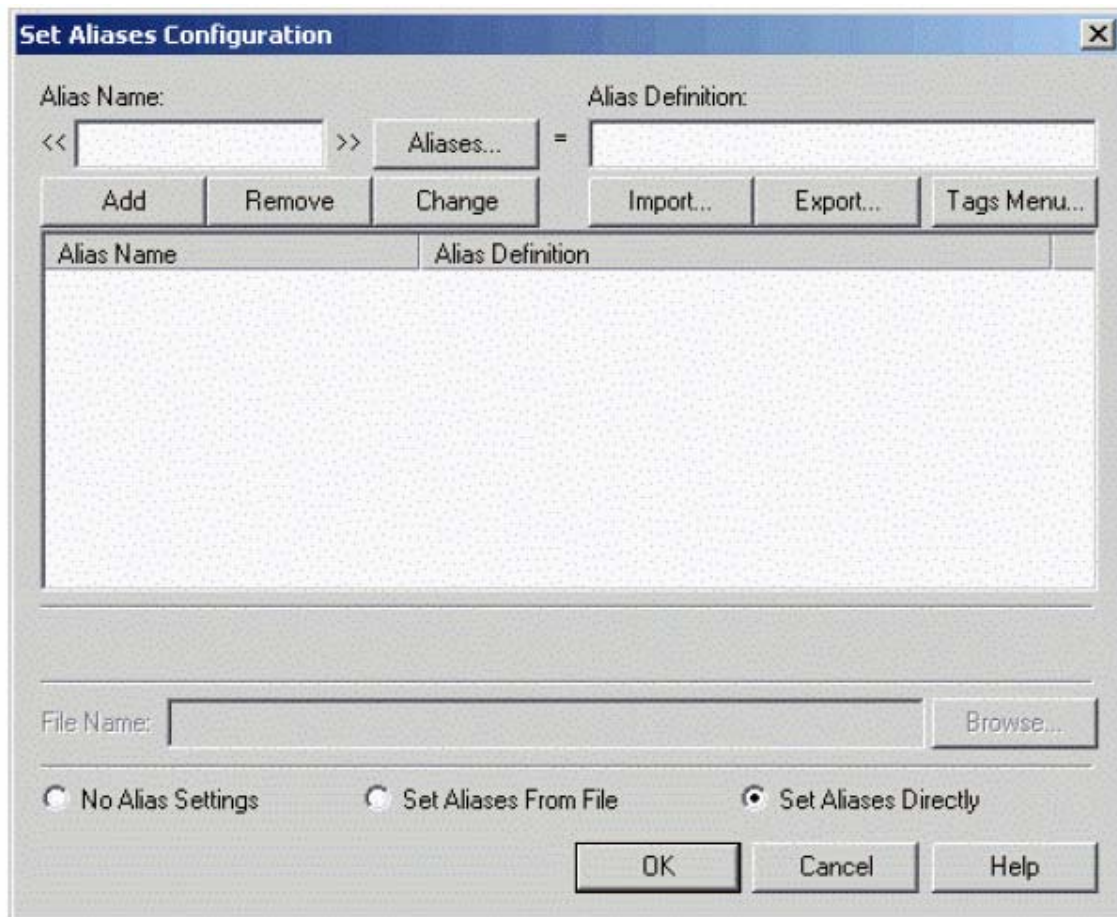
- **Pick action-Load display:** setuje aliase u novo napunjeni displej
- **Pick action – Drag/drop load display:** setuje aliase u novi drop-loadovani displej.
- **Pick action-popup window:** setuje aliase u novi pop-up prozor.
- **pick action-embedded window:** setuje aliase u novi uronjeni prozor
- **Pick action-Set aliases:** setuje specificirane aliase u tekući otvoreni displej, može se primjeniti na cijeli dispelj ili specificirani objekat ili simbol
- **Pick action-Alias dialog:** predstavlja Operatoru dijalog, i setuje aliase u tekućem displeju na bazi selekcija koje napravi operator u ovom dijalogu; može se primjeniti na cijeli displej ili specificirani objekat ili simbol.
- **Tab load Display:** svaki tab može uključivati set aliasa
- **Launching GraphWorX32 from the command line :** Korisnik može specificirati alias fajl kao jedan od parametara na komandnoj liniji.

Da se setuju lokalne aliase , treba kliknuti na **Set Aliases** taster na odgovarajućem pick action tabu u Property Inspectoru i izabrati **Local Aliases** iz pop-up menija, kao što je pokazano na narednoj slici:



Setovanje aliasa iz Pick akcije

Ovo će otvoriti Set Aliases Configuration dijalog boks, kako je pokazano na narednoj slici. Ovaj intefejs je sličan za sve akcije, ali može malo varirati zavisno od akcije koja se konfigurira.



Konfiguracioni dijalog boks za setovanje aliasa

Parametri setovanja aliasa

Parametar	Opis
Alias Definition	mjenja definicije za sve izabrana imena aliasa
Tags Menu taster	Displejira meni sa slijedećim opcijama: OPC Tags, Expression Editor, Local Variables, Simulation Variables, i Global Variables
No Alias settings	Ako se ova opcija izabere, ne mogu se specificirati aliasi
Set aliases from fajl	Aliasii mogu doći iz vanjskog tab delimitiranog tekst fajla. Ovi fajlovi se mogu izvesti iz Excela, kreirati u Notepadu, ili kreirati koristeći GraphWorX32 Alias File Editor. Treba kliknuti Browse taster da se izabere fajl .
Set Aliases directly	Aliasii koji treba da se setuju se mogu specificirati direktno u GraphWorX32 koristeći Set Aliases Configuration dijalog boks.

Postoje takodjer mnoge koje korespondiraju mnogim od gore pobrojanih akcija.

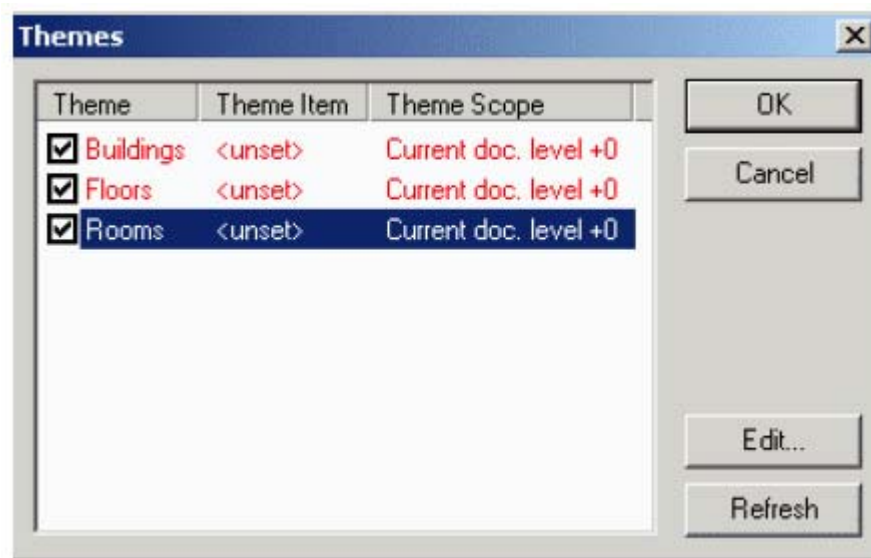
Opaska: Alias zamjenski stringovi se mogu modificirati za vrijeme runtimea putem OLE Automation-a.

Korisnik može također specificirati globalne aliase klikanjem na **Set Aliases** taster na odgovarajućem pick action tabu u Property Inspector-u i izabrati **Global Aliases** iz pop-up menija.

Izabrati temu iz **Themes** dijalog boksa, kako je pokazano na narednoj slici. Kliknuti zatim na **OK**.

Opaska:

Da se editiraju osobine teme, izabrati temu i kliknuti na **Edit** taster. Ovo otvara **Theme Editor**, gdje korisnik daje detalje tema i opseg tema (theme scope).



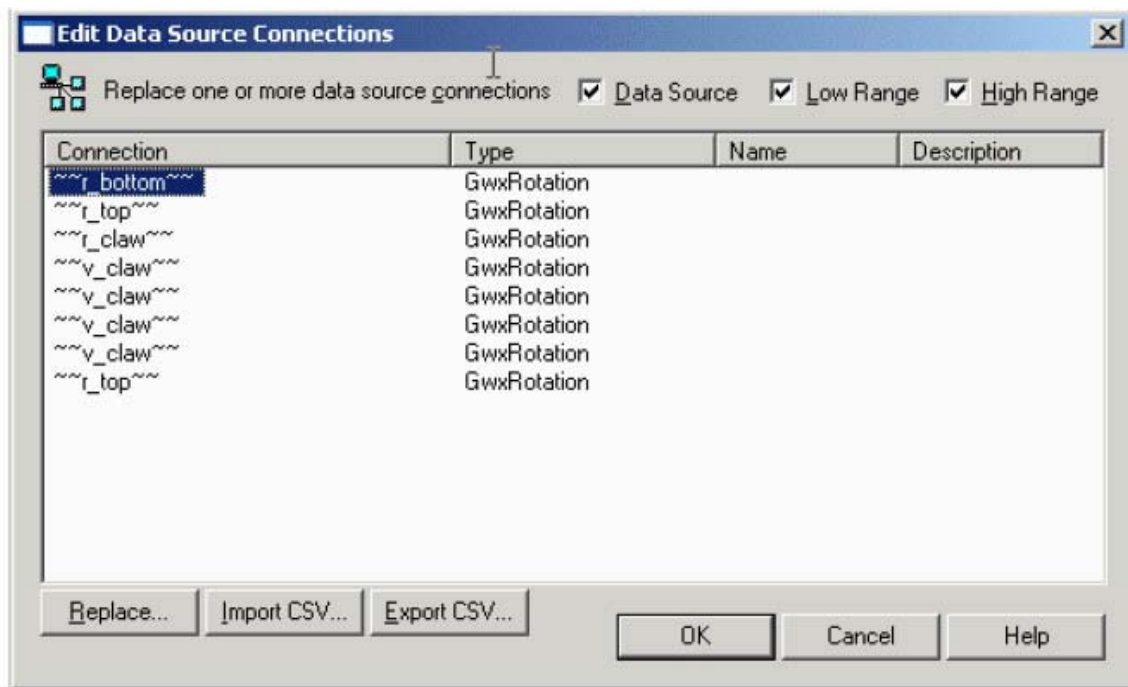
Izabiranje globalnih alias tema

Aliasing drugog nivoa (second-level aliasing)

Aliasing drugog nivoa dozvoljava alias ime da se unese kao alias definicija. Ovaj drugi alias će biti rezriješen prije nego što se izvrši **set – aliases** komanda.

Editiranje konekcija sa izvorima podataka

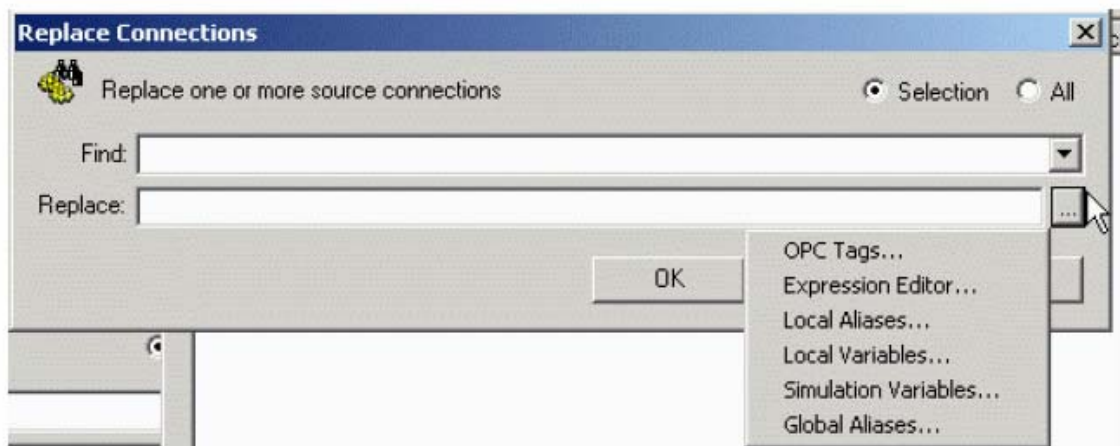
Izabiranje **Edit Connections** iz **Dynamics** menija otvara **Edit Data Source Connection** dijalog boks, koji je pokazan na narednoj slici, koji pojednostavljuje konekciju sa GraphWorX32 simbola na OPC podatke.



Editiranje konekcija sa izvorima podataka

Zamjenjivanje konekcija

Klikanjem na **Replace** taster otvara **Replace Connections** dijalog boks, kao što je pokazano na narednoj slici. Ovo omogućava da se provede search and replace operacija za jednu ili više konekcija, na jednom ili više simbola na tekućem displeju. Mogućnost zamjene podržava i wildcard zamjenu, slično kao i kod Find/Replace komande.



Dijalog boks za zamjenu konekcije

Import i export

Dodatno, tagovi mogu biti automatski konektirani na simbole uvodjenjem *.csv fajla i njegovim korištenjem za auto konekcije. Kliknuti na Import CSV taster da se browsiraju svi .csv fajlovi.

Dinamičke konekcije

Pokazaćemo kako treba realizovati dinamičke konekcije od objekata na displejima do tačaka koje dolaze iz raznih OPC data servera. Ovi objekti su animirani i kontrolirani sa vrijednostima ovih tačaka podataka. Korisnik može realizovati višestruke dinamičke konekcije na jedan objekat. Na primjer, korisnik može kreirati objekat koji se mjenja u boji i veličini na bazi vrijednosti iz dvije različite konekcione tačke.

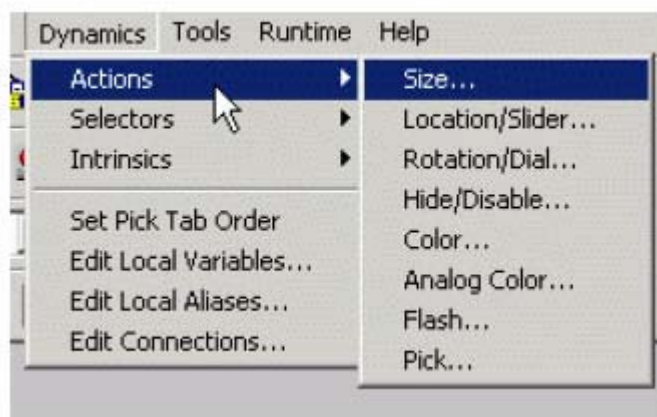
Korisnik kreira dinamičke konekcije sa funkcijama koje su sadržane u Dynamics meniju. Tipovi dinamičkih konekcija uključuju flash, boju procesnu tačku (PPT), unos podatka, veličinu, lokaciju, analogni i digitalni selektor, animator, rotaciju i sakrivanje.

Izvor podatka

U svakom dinamičkom dijalogu postoji izvor podatka,

Dinamičke akcije

Konekcije za dinamičke akcije primjenjuju specifične akcije na objekte na displeju na bazi vrijednosti tačaka podataka. Korisnik može kreirati ove konekcije putem **Actions** podmenija na Dynamics meniju, kako je to pokazano na narednoj slici, ili klikanjem na odgovarajući taster na Dynamics toolbaru:



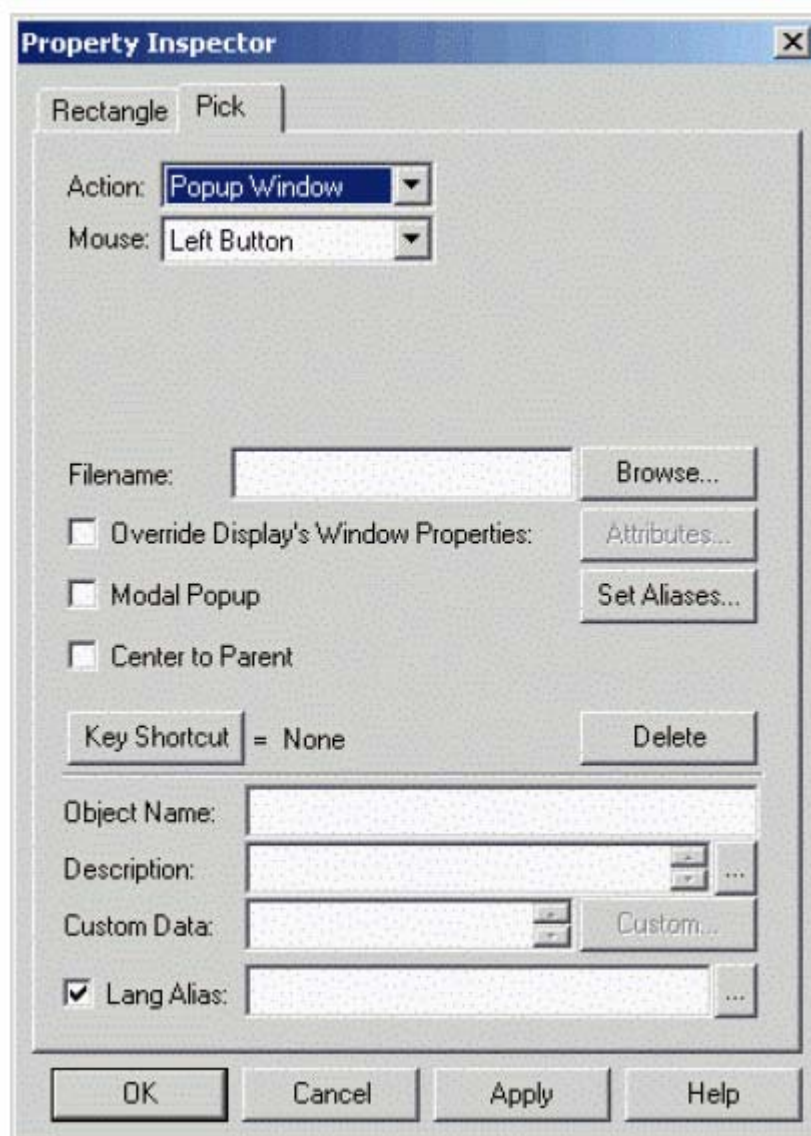
Pik (pick) akcije

Funkcija **Pick Action** omogućava korisniku da konfigurira objekat koji izvršava akciju kada objekat se klikuje u runtime modu rada. Takve akcije uključuju loadovanje prikaza, pop-up (iskakanje) prozora , lansiranje aplikacija, i download vrijednosti.

Da bi se izgradila pick action konekcija treba:

1. Izabrati objekat na displeju
2. Izabrati **Actions>Pick** iz **Dynamics** menija , ili kliknuti na **Pick** taster na **Dynamics** altnoj letvici.
3. **Pick** tab na Property Inspector za taj objekat se otvara, kako je pokazano na narednoj slici.
4. Izabrati željene parametre iz kombo boksa u Action polju.

Opaska: izgled Pick taba varira zavisno od tga koja je akcija izabrana



Dijalog boks Property Inspector: Tab za Pick akciju

Klikanjem na **Key Shortcut** taster, otvara se **Define Key Shortcut** dijalog boks, koji je pokazan na narednoj slici. Unjeti taster ili izabrati modifikator, i zatim kliknuti OK. Kada se displej promjeni na runtime mode, Operator može izvršiti pick akciju pritišćući definisane tastere za shortcut:



Akcija Run Script

Run Script pick akcija u Pick tabu na Proprety Inspectoru starta tri različita tipa skripti: VBA, VBScript i Jscript.

Da bi se to realizovalo , treba:

1. Izabrati tip iz Script Type drop-down liste, kao što je pokazano na narednoj slici:
2. Kada se izabere programski jezik, odgovarajući editor se može koristiti da se napiše kod za pick akciju. Ako izaberemo VBScript ili Jscript, novi skript će biti generiran u Script Editoru.

Run Script parametri

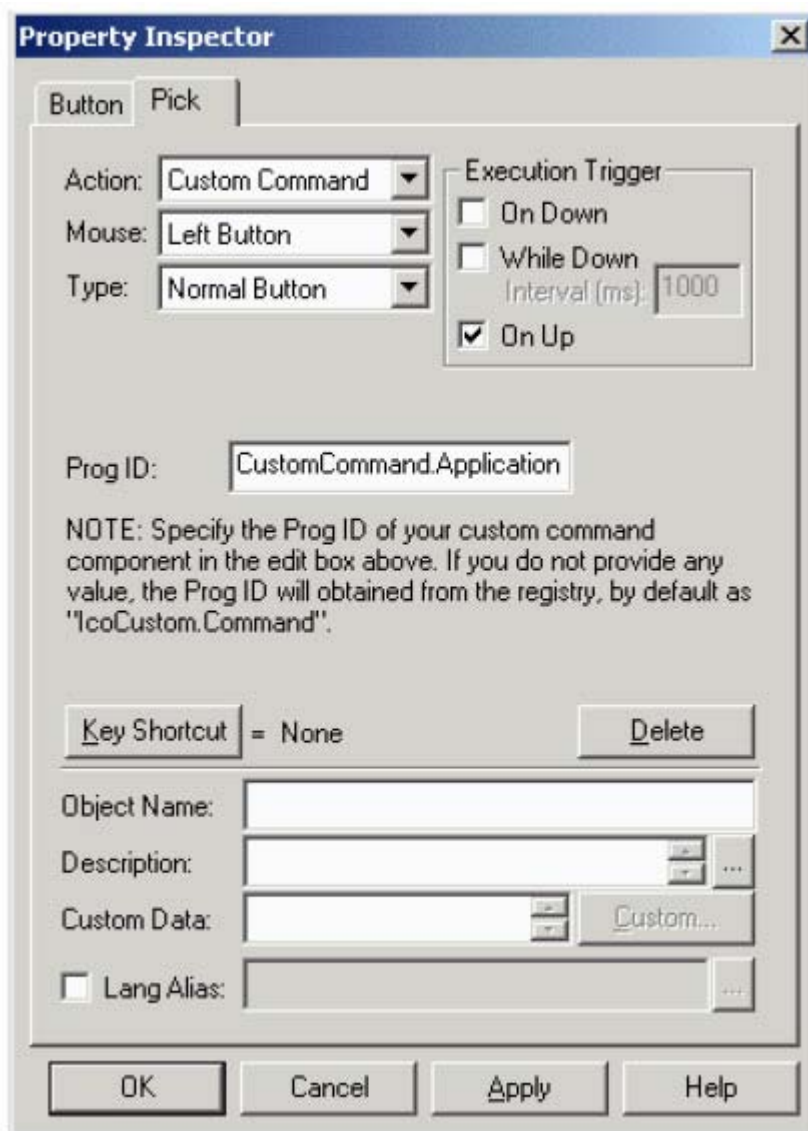
Parametar	Opis
Script Name	Ime VBA makroa koji će biti izvršen. Ovo ime je pohranjeno u GraphWorX32 fajl .
Script type	Izabira VBA Script, VBScript ili JScript
Create	Otvora VBA Script Wizard , koji omogućava da kreiramo macro ime.
Edit	Otvora odgovarajući skript editor , koji se može koristiti da se napiše kod za pick akciju.



Akcija kastomizirana komanda

Custom Command pick akcija u Pick tabu na Proprety Inspektoru se može koristiti da izvršava kastomiziranu funkciju. Ova funkcija može biti izvršivi fajl (.exe), ili *.dll fajl .

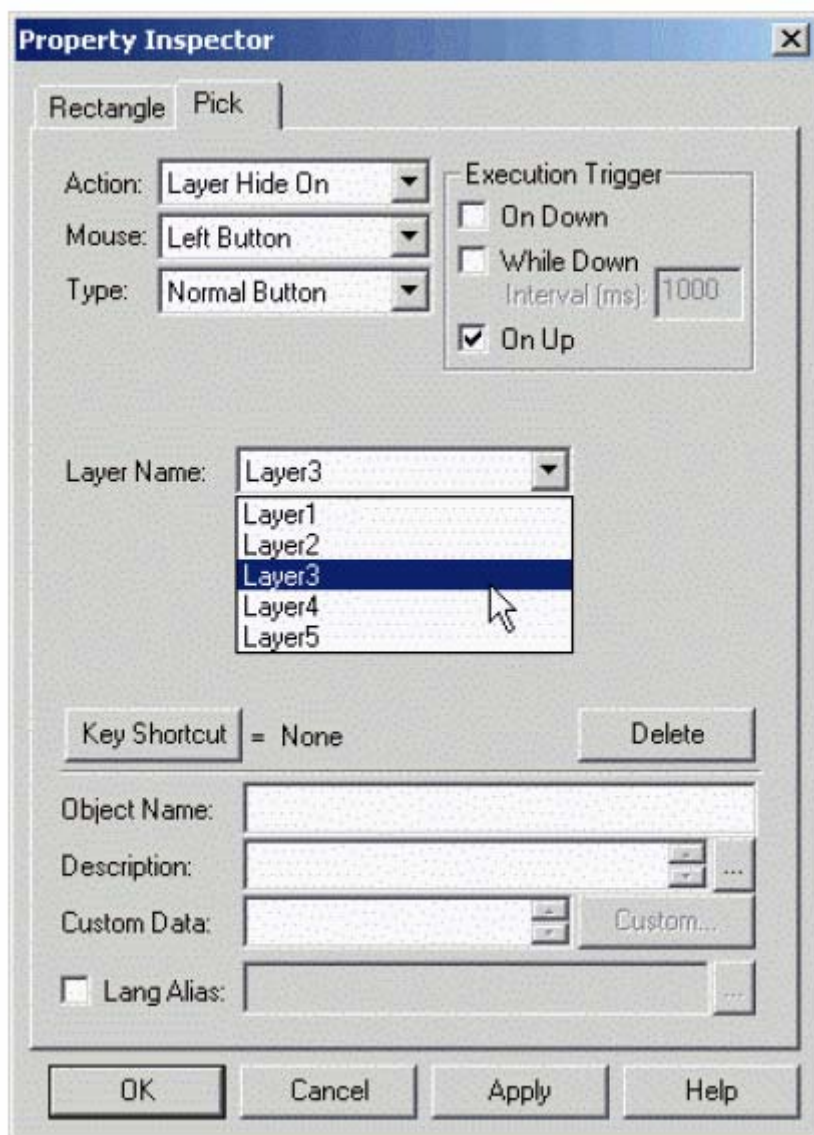
GraphWorX32 podržava višestruke DLL-ove kastomiziranih komandi. U polju **Prog ID** , možemo specificirati Prog ID od kastomizirane komandne komponente. Ako korisnik ne obezbjedi ovu vrijednost, Prog ID će biti dobijena iz registara po defaultu kao "IcoCustomCommand".



Akcija – Slojevi (layers)

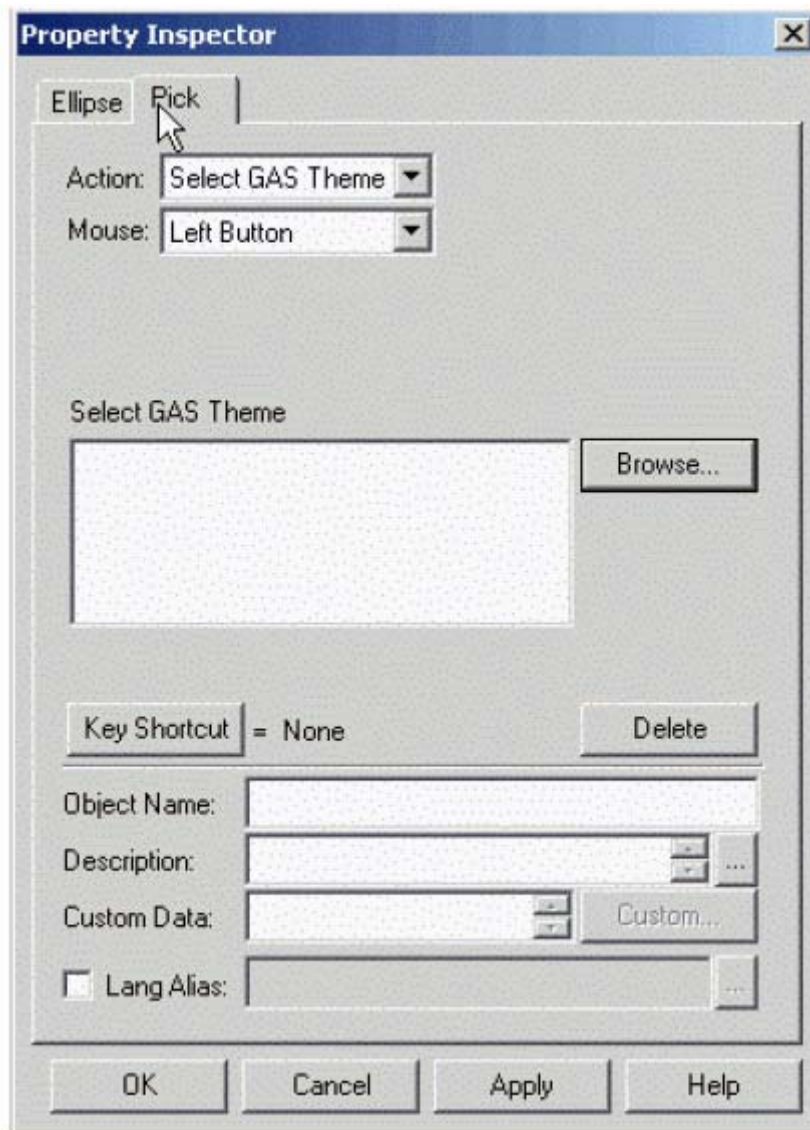
Korisnik može sakriti, pokazati, i preklopiti slojeve u prikazima koristeći pick akcije **Layer Hide On**, **Layer Hide Off**, i **Layer Hide Toggle**, u **Pick** tabu u **Property Inspector** dijalog boksu, kako je to pokazano na narednoj slici.

Treba izabrati odgovarajuću pick akciju, i zatim izabrati sloj iz Layer Name drop-down liste, kao što se to vidi sa slike:

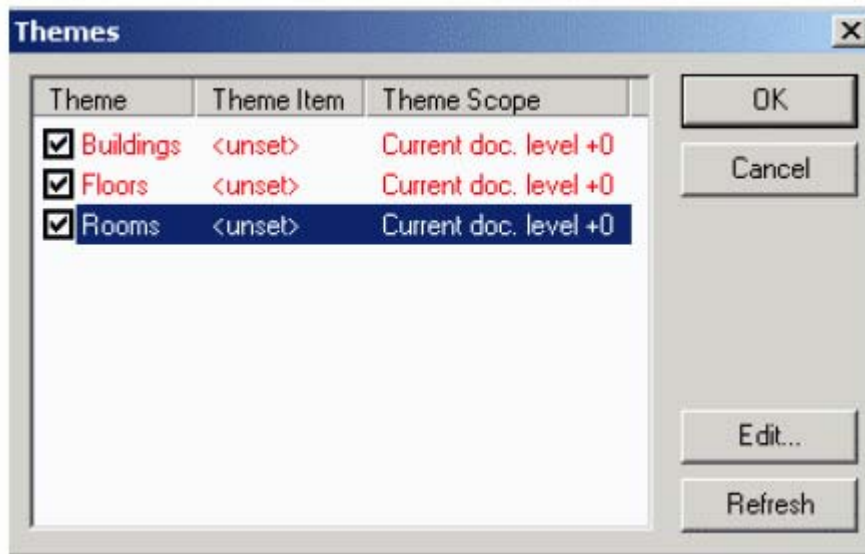


Akcija – Selekcija GAS teme

Pick akcija **Select GAS theme** , u Pick tabu Property Inspector-a omogućava korisniku da izabere globalne alias (GAS) teme, kao što je to pokazano na narednoj slici:

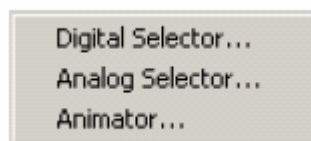


Opaska: Da bi se editirale osobine teme, treba izabati temu i kliknuti na **Edit** taster. Ove će otvoriti **Theme Editor**, gdje korisnik može specificirati elemente teme i njen opseg (scope).



Dinamika selektora

Na narednoj slici je pokazan meni za selektore koje korisnik može izabrati :



Selektori prikazuju specifične objekte na bazi vrijednosti data konekcije. Korisnik može pristupiti selektorima izabirući **Selectors** iz **Dynamics** menija, kao što je pokazano na gornjoj slici, ili iz **Dynamics** alatne letvice. Postoje tri tipa selektora:

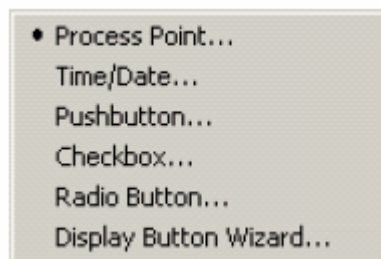
- **Digital Selector:** on prikazuje jedan od niza objekata na bazi stanja digitalnih signala. Po jedan signal je spojen na svaki objekat. Signal najvećeg prioriteta (odredjen pozicijom u listi), prikazuje odgovarajući objekat kada je njegova vrijednost true (ili u opciji može biti false).
- **Analog Selector:** prikazuje jedan od niza objekata na bazi vrijednosti analognog signala. Ovaj objekat se prikazuje kada signal upada unutar opsega vrijednosti specificiranog za taj objekat.
- **Animator** : Izabira grupu objekata da ih sekvencijalno prikaže u zavisnosti od stanja digitalne varijable sa kojim su oni povezani.

Opaska: Digitalni selektor, analogni selektor, i animator se mogu modificirati nakon kreiranja koristeći podsimbolsko (subsymbol) editiranje. Izabrati selektor i zatim bilo sa : **SHIFT + klik desni taster** ili **Klik desni taster + Edit Symbol** ući u editiranje podsimbola. U njemu korisnik može dodati, modificirati, i izbrisati objekte u digitalnom

selektoru. Pritisnuti na **Escape** taster ili dvostruko kliknuti van objekta da se zatvori editiranje podsimbola. Unesene promjene će se odraziti na selektoru. Koristiti Property Inspector da se spoje novo dodati objekti, ako je potrebno.

Intrinsic dinamika

Intrinsic dynamics omogućava korisniku da kreira operatorske kontrole u displej fielovima. On može napraviti dinamičke konekcije da kontrolira objekte koji prikazuju real-time podatke za vrijeme runtime moda, kao što su objekti za unos podataka, slajderi i tasteri (push buttons). Može se pristupiti kontrolnim funkcijama iz **Intrinsic** submenija od **Dynamics** menija, kao što je pokazano dole, ili iz Dynamics alatne letvice (toolbar).



Procesne tačke i unos podataka

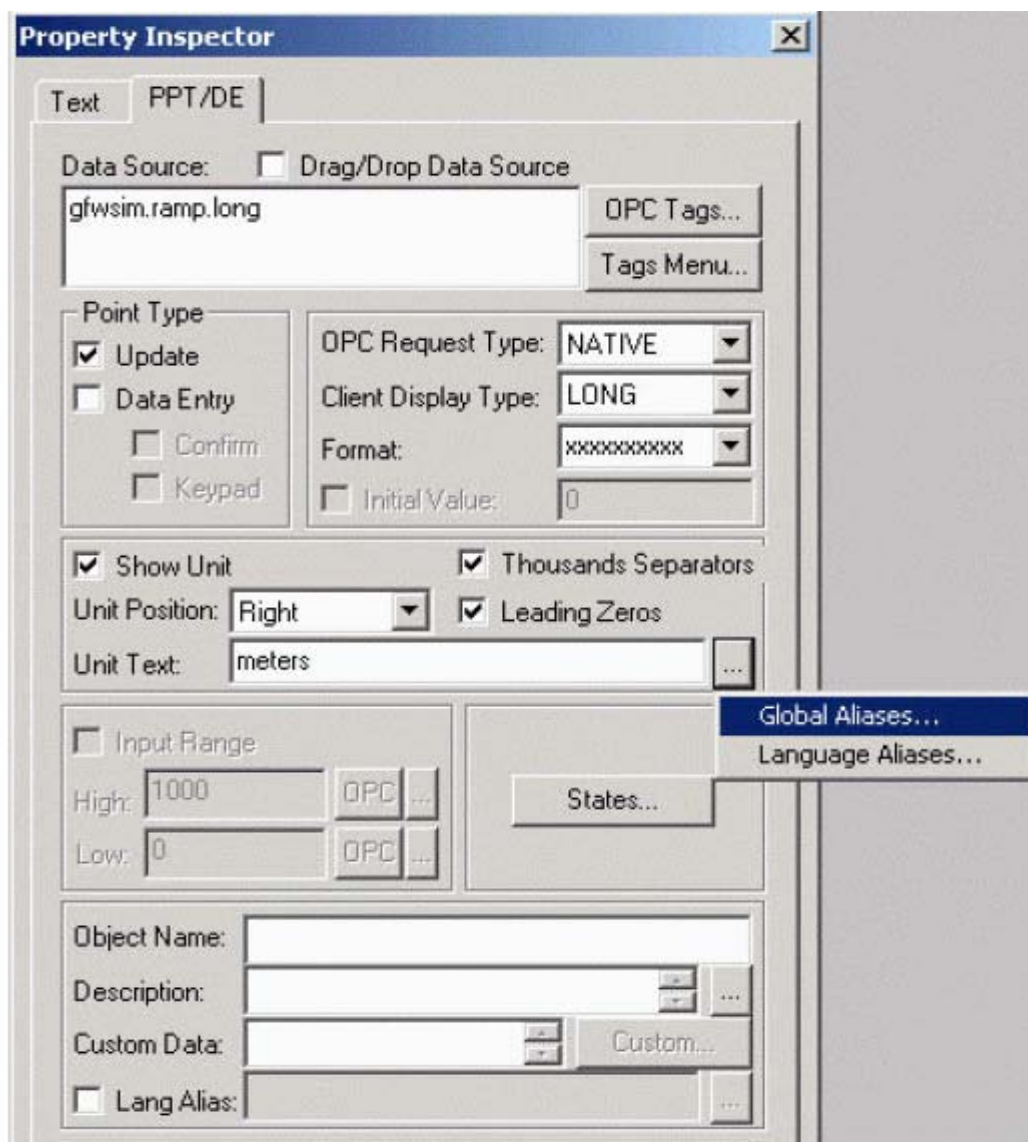


Funkcija Process Point/Data Entry kreira objekat koji se koristi da prikaže i unese podatak u alfanumeričkom formatu. **Process point** je objekat koji prikazuje real-time vrijednost spojene tačke podatka za vrijeme runtime-a. Ova vrijednost se ažurira kada server obavjesti GraphWorX32 o promjeni u vrijednosti tačke podatka.

Unos podatka (Data entry) se ponaša na isti način kao i procesna tačka, ali unos podatka takodjer dozvoljava Operatoru da ručno unese nove vrijednosti koje treba downloadovati u sistem.

Da bi se kreirao process point/data entry (PPT/DE) objekat , uraditi:

1. Izabrati **Intrinsic>Process Point** iz **Dynamics** menija, ili kliknuti na **Process point** taster na **Dynamics** alatnoj letvici.
2. Kliknuti unutar radne zone gdje želimo da smjestimo procesnu tačku. Ovo će otvoriti Process Point/Data Entry dijalog boks, pokazan na narednoj slici:
3. Izabrati parametre izlistane u narednoj tabeli:
4. Dodatne osobine koje su vezane sa vizuelnim izgledom PPT/DE mogu se dobiti izborom Text taba. Tekst osobine procesnih tačaka su slične onima koje se koriste za editiranje statičkih tekst objekata.
5. Kliknuti OK. Boks sa upitnicima koji predstavljaju PPT/DE format će se pojaviti u radnoj zoni.



Dijalog boks Property Inspector . tab za procesnu tačku

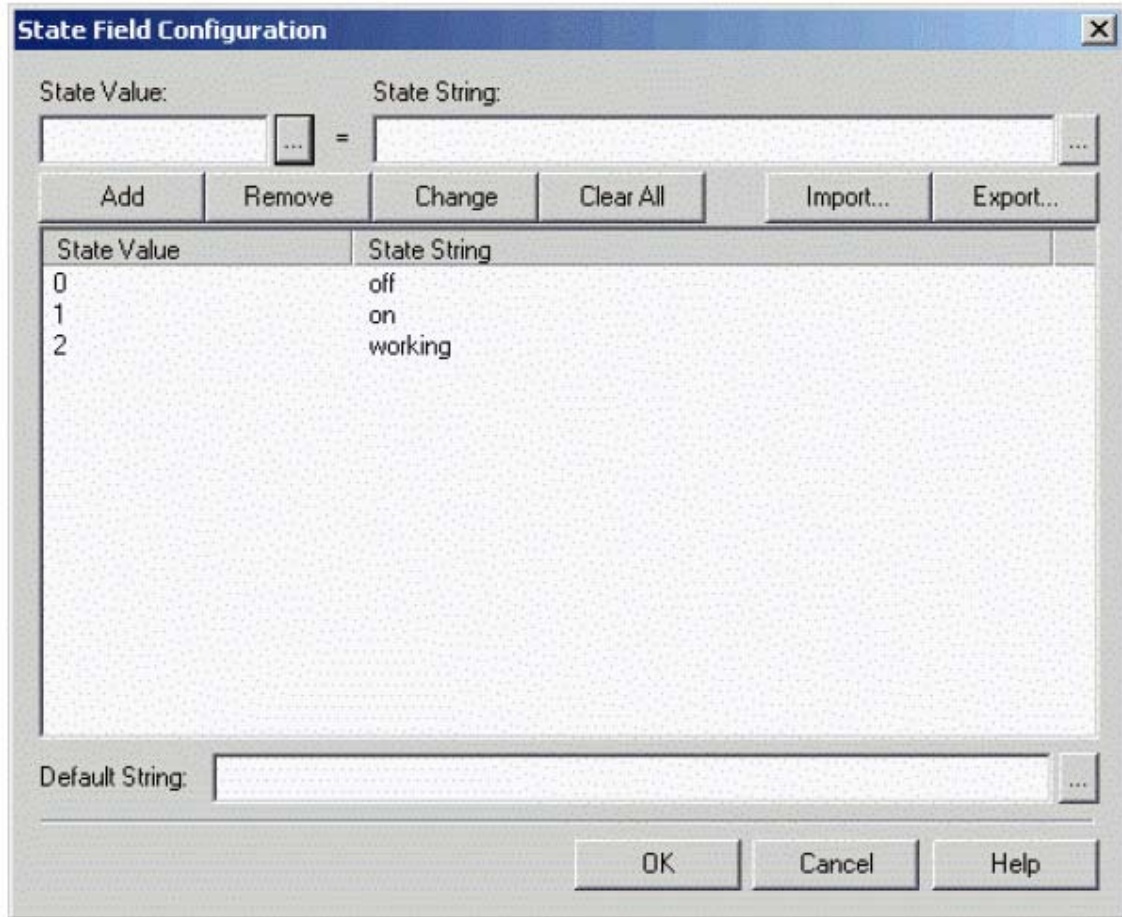
Parametri za unošenje PPT/DE

Parametar	Opis
Data source	Svaki dinamički dijalog ima izvor podatka. Konekcije sa podacima se mogu unjeti direktno u polje Data Source u konfiguracionom dijalogu različitih dinamika. Klikanjem na OPC Tags taster otvara se OPC Universal Tag Browser , iz kojeg možemo izabrati tagove u sistemu. Možemo takodjer koristiti opcije pod Tag menu da napravimo konekcije sa podacima.
Update	Definira tip tačke koju sistem ažurira u realnom vremenu.

Data Entry	Mjenja vrijednost spojene tačke podatka u runtime modu.
Keypad	Prikazuje tastaturu ili numerički pad da omogući da se unese vrijednost kada se klikne na procesnu tačku za vrijeme runtime-a.
OPC request type	OPC Request type se koristi u komunikaciji sa OPC podatkom između GraphWorX32 (kao OPC klijenta) i OPC servera, i između OPC Servera i I/O uređaja sa kojima je OPC Server spojen
Client Display type	Ovo polje specificira kako OPC podatci iz OPC servera se pokazuju u GraphWorX32 prikazu
Format	Specificira broj decimalnih mjesta za konektovane tačke podataka u runtime-u.
Float	Decimalna tačka sa pokretnim zarezom. Može se definirati naučna notacija kao napr. "xx.xxe"
.....	ostali formati podataka
Initial Value	definira početnu (inicijalnu – startnu) vrijednost na koju se inicijalizira objekat za unos podatka.
show unit	treba čekirati ovaj boks ako se želi da se prikaže mjerna jedinica u runtime-u
.....
states	Otvora State Field Configuration dijalog boks, koji dozvoljava da se unese značenje binarne vrijednosti.

Polja stanja (state fields)

Klikanjem na **States** taster u **PPT/DE** tabu Property Inspectora , otvara se State Field Configuration dijalog boks, koji je pokazan na narednoj slici, koji omogućava da se unese značenje binarne vrijednosti. Polja stanja (state fields) se odnose na stanje varijable. Ona su obično binarne vrijednosti , koje su prikazane na string, koji ima neko značenje.

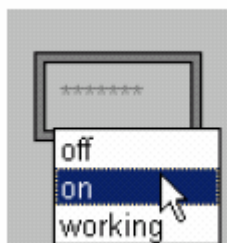


Konfiguracioni dijalog boks za polje stanja

Konfiguriranje polja stanja je vrlo slično onom kod aliasa. Korisnik može unjeti vrijednost stanja ili jednostavno izabrati stanje koje već postoji, unjeti ili promjeniti string stanja, i onda kliknuti odgovarajući taster akcije (Add, Remove, Clear ili Clear All).

Import i **Export** tasteri otvaraju **Open** i **Save AS** dijaloge. Default string se odnosi na string koji će se pojaviti kada stanje nema neki unaprijed konfigurirani string priključen tom stanju.

Ako procesna tačka je setovana na Data Entry, drop-down meni svih mogućih stanja bit će na raspolaganju , kao u narednom primjeru:



Data entry procesna tačka u runtime-u

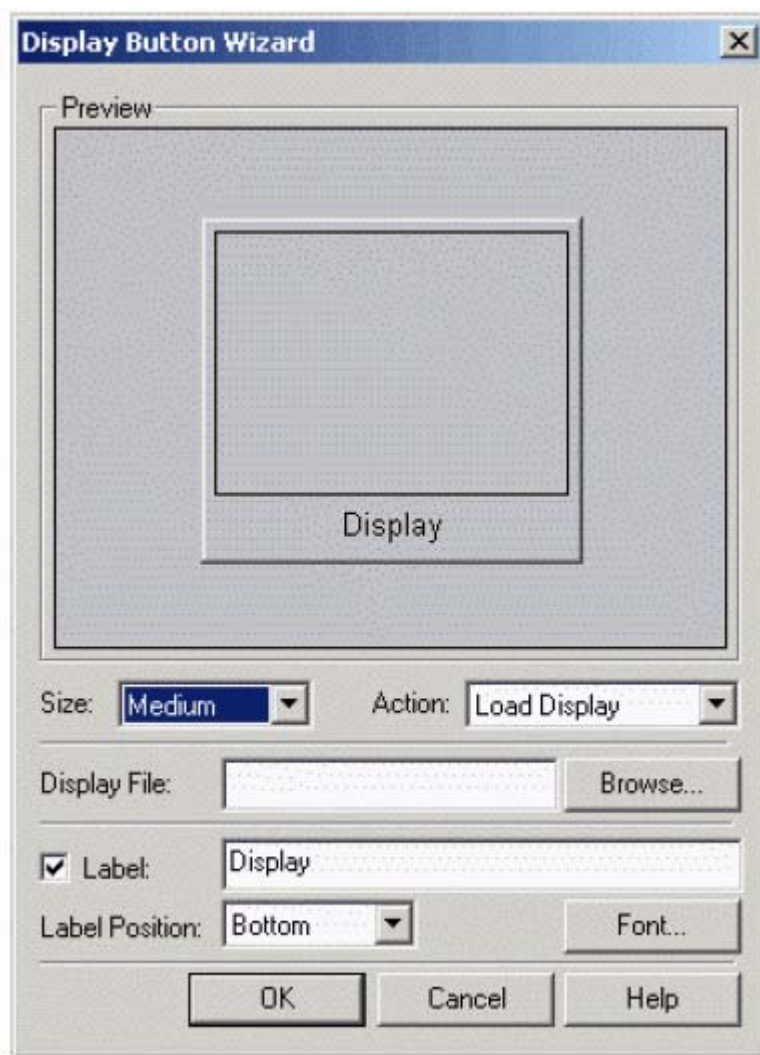
Wizard Displej tastera

GraphWorX32 sadrži i opciju koja se zove **display button**. Ovaj displej taster , kada se konfigurira pomoću wizarda , može da uradi jednu od slijedećih akcija:

- loaduje novi displej
- otvori displej u uronjenom GraphWorX32 prozoru
- prikaže pop-up prozor
- služi kao drag-and-drop objekat koji može biti postavljen u GraphWorX32 ActiveX kontejner da loaduje displej.

Ova mogućnost smanjuje gomilanje (clutter) u prikazima pošto uronjeni (embedded) prozori i pop-up prozori se pokazuju samo onda kada su potrebni, pritiskom na taster.

Da bi se konfigurirao displej taster, izabrati **Intrinsic> Display Button Wizard** iz **Dynamics** menija, ili kliknuti na **Display Button** na **Dynamics** alatnoj letvici. Ovo će otvoriti **Dysiplay Button Wizard** kao na narednoj slici:



Preview polje pokazuje kako će izgledati površina tastera kada se konfigurirše.

Display taster wizard – akcija

Ovaj kombo boks određuje kako će se taster ponašati u runtime-u. Svaka od četiri raspoložive akcije je izlistana i koristi displej koji je unesen u polju **Display File** :

- **Load display** : zatvara tekući displej i otvara novi displej
- **embedded window**: otvara novi displej u uronjenom GraphWorX32 prozoru, i nije ograničen na unutrašnjost GraphWorX32 prikaza nego se može maksimizirati ili minimizirati od strane korisnika.
- **popup window**: otvara popup prozor koji, za razliku od uronjenog prozora nije ograničen na unutrašnjost GraphWorX32 displeja i može se maksimizirati ili minimizirati od strane korisnika.
- **drag/drop load** : kada se izabere ova opcija, taster se više ne ponaša kao standardni. Sada će se on ponašati kao drag/drop objekat. Ako postoje GraphWorX32 kontejneri u displeju, taster se može prevući i ispustiti u kontejner , prouzrokujući da se specificirani displej loaduje.

Kustomizacija konfiguracija

Svaki tab u GraphWorX32 Proprety Inspectoru ima **Custom** taster koji je lociran pored **Custom Data** polja za editiranje. Ovaj taster se može povezati sa kustomizacionom konfiguracionom utility, koju treba da programira korisnik.

Klikanjem na ovaj taster kreira se Automation objekat koji ima neki program ID (ProgID). Po defaultu taj ProgID je **IcoCustom.Configure**. Ovaj ProgID se može prepisati sa setovanjem alternativnog ProgID u Windows registru:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Gwx32\OEM  
Information\Custom Configure ProgID = "YourProgIDHere"
```

Korisnik može željeti da ovaj Automation objekat se eksponira iz DLL: U tom slučaju GraphWorX32 koristi dispeč intefejs ovog objekta , da poziva metod:

```
boolean CustomConfigure(long hWndParent, IDispatch* ObjectOne,  
BSTR ObjectOneType, IDispatch* ObjectTwo, BSTR ObjectTwoType);
```

gdje su:

hWndParent- je window handle od GraphWorX32 proprety inspektora.

ObjectOne- dispeč pointer od GraphWorX32 objekta koji je pridružen sa Gwx32 proprety stranicom koja se pokazuje (t.j. ako je "pick" strana trenutno aktivna, ObjectOne je "GwxPick" objekat za tu stranu).

ObjectOneType- ime stringa tipa objekta za ObjectOne (napr. "GwxPick").

ObjectTwo- dispeč pointer od GraphWorX32 displeja kojem ObjectOne pripada.

ObjectTwoType- ime stringa tipa objekta za ObjectTwo (t.j. "GwxDisplay").

Nakon što se metod povraća, GraphWorX32 oslobadja dispeč pointer za "IcoCustom.Configure", automation objekat.

Konci (threads) za podatke i tajmere

Data thread i **Timer thread** se mogu kustomizirati za setinge registara. Ako postoji potreba da se poveća prioritet threada od GraphWorX32 , moguće je konfigurirati ove registarske ključeve u registru:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Gwx32\Runtime
Settings\DataThread\Priority
```

```
HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Gwx32\Runtime
Settings\TimerThread\Priority
```

Vrijednosti su stringovi, i slijedeći stringovi su validni:

-15	THREAD_PRIORITY_IDLE
-2	THREAD_PRIORITY_LOWEST
-1	THREAD_PRIORITY_BELOW_NORMAL
0	THREAD_PRIORITY_NORMAL [default]
1	THREAD_PRIORITY_ABOVE_NORMAL
2	THREAD_PRIORITY_HIGHEST
15	THREAD_PRIORITY_TIME_CRITICAL

Izvršenje kustom komandi

Ovo je jedna od opcija za PICK dynamic akciju u GraphWorX32. Korisnik može izvršavati kustomizirane funkcije ili aplikacije koristeći opciju Custom Command. Kustom aplikacija treba da bude specificirana od strane korisnika. Aplikacija ili funkcija je ili izvršni fajl (.exe) ili DLL:

Prvi puta kada korisnik klikne na pick akciju, koja je **Custom Command**, GraphWorX32 kreira Automation objekat koji ima odredjeni programski ID (ProgID). Po defaultu, ProgID je: **IcoCustom.Command**. Ovaj ProgID može biti prepisan setovanjem alternativnog ProgID u registarskom ključu:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Gwx32\OEM
Information\Custom Command ProgID = "YourProgIDHere"
```

Korisnik će vjerovatno željeti da Automation objekat je eksponiran iz EXE. Tada EXE treba dozvoliti korisniku da kreira višestruke instance ovog automation objekta bez lansiranja višestrukih instanci aplikacije.

GraphWorX32 tada koristi dispeč interfejs ovog objekta da pozove metod:

```
boolean CustomCommand(long hWndParent, IDispatch* ObjectOne,
BSTR ObjectOneType, IDispatch* ObjectTwo, BSTR ObjectTwoType);
```

Parametri su isti kao i kod **CustomConfigure()** funkcije koja je opisana prije. Nakon što se metod povraća, GraphWorX32 ne oslobadja dispeč pointer za "IcoCustom.Command" automation objekat. GraphWorX32 drži dispeč pointer za slijedeći put kada se Custom Command izvrši (tako da GraphWorX32 ne treba da kreira ponobno objekat,) tako da je izvršenje brže. Ovaj dispeč pointer se ne oslobadja sve dok se ova instanca GraphWorX32 ne zatvori.

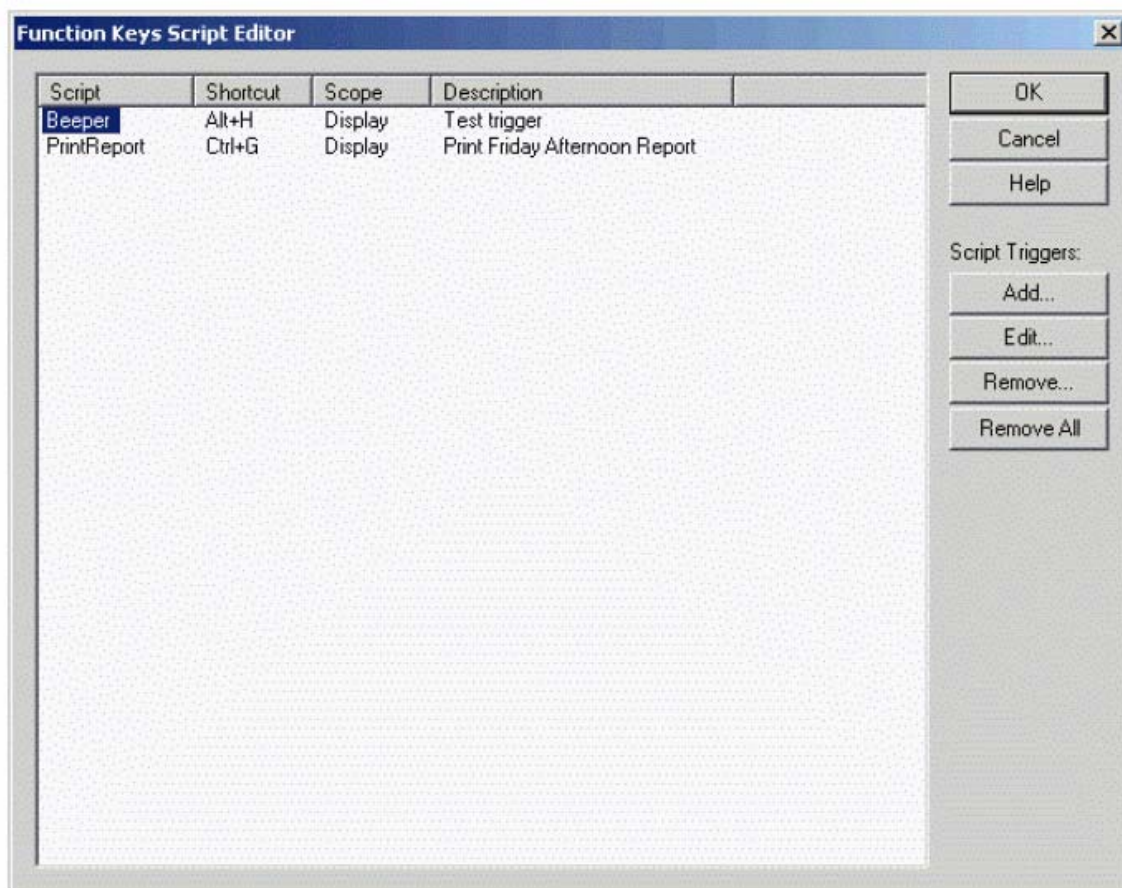
Meni alata (tools menu)

Meni Tools omogućava korisniku da pristupi mnogim alatima koji se koriste u manipulaciji sa GraphWorX32 podacima. Slijedeća lista opcija je na raspolaganju sa ove liste:

- Makroi (Create, Edit, Step , Run, Clean Unused VBA Modules (čišćenje ne korištenih VBA modula), i otvaranje Visual Basic Editora.
- Funkcionalni ključevi
- Postavljanje radnog direktorija
- Konfigurisanje sigurnosti
- Editor lokalnog fajla aliasa
- Konfigurisanje globalnih aliasa
- Konfigurisanje jezičkih aliasa
- 16 u 32 bitni prevodilac
- publikovanje u HTML-u
- Windows CE (konfigurisanje za Windows CE OS)

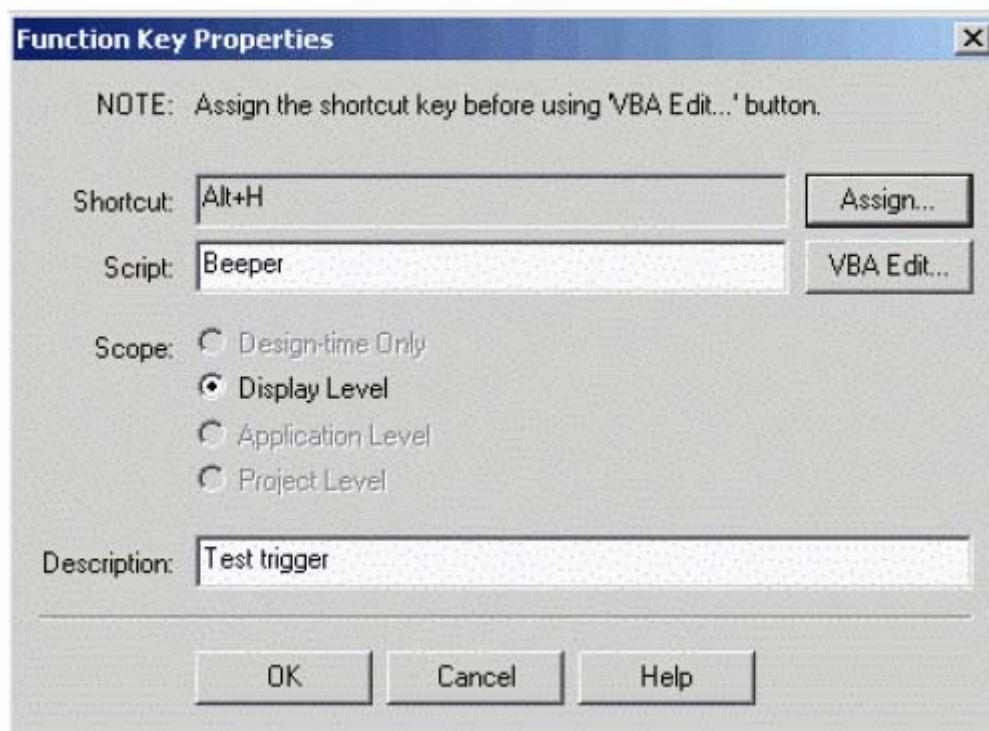
Funkcionalni ključevi

Mogućnosti **Functions keys** koje su na raspolaganju na **Tools** meniju obezbjeđuje jednostavnu manipulaciju sa funkcionalnim ključevima i hijerarhisku funkcionalnost. Izabiranjem **Function keys** iz **Tools** menija otvara **Functions keys Script Editor**, pokazan na narednoj slici, koji omogućava korisniku da dodaje, editira i otklanja trigere. **Trigger** je kombinacija ključeva shortcuta i VBA Script imena, koje skupa omogućuju otvaranje VBA skripta kada se pritisne shortcut ključ(eve).



Skript editor funkcionalnih ključeva

Klikanjem na **Add** taster ili **Edit** taster otvara se **Function Key Properties** dijalog boks, pokazan na narednoj slici, koji omogućava da specificiramo shortcut i skriptu.



Osobine funkcionalnih ključeva

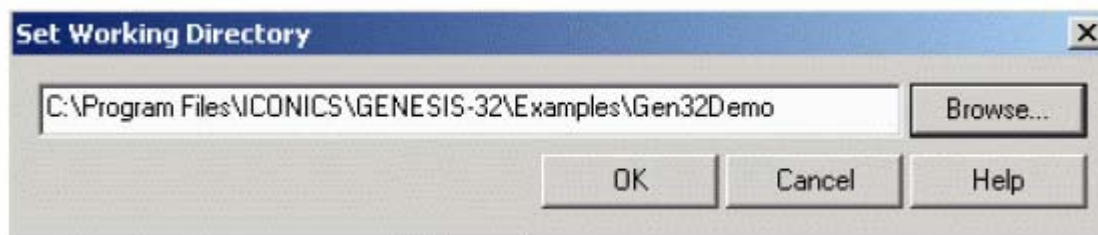
Možemo ili ukucati ime skripta ili kliknuti na **VBA Edit** taster, koji kreira skriptu ili skače na skript ako on već postoji. Korisnik sada može unjeti tijelo skripte u VBA editoru. Preporučuje se da se debugira skripta i verificira da se kompilira bez problema klikanjem na **Debug** i **Compile Project** u VBA Editoru.

Polje Scope specificira opseg skripte. U ovoj verziji 7.1 GraphWorX32 samo su na raspolaganju skripte za opseg nivoa displeja (**Display Level**). Funkcionalni ključevi nivoa displeja su prvi korak hijerarhije funkcionalnih ključeva. Druge opcije opsega su onemogućene.

Polje opisa (**Description**) je tekst polje koje se pojavljuje u posljednjoj koloni od **Function Keys Script Editora**.

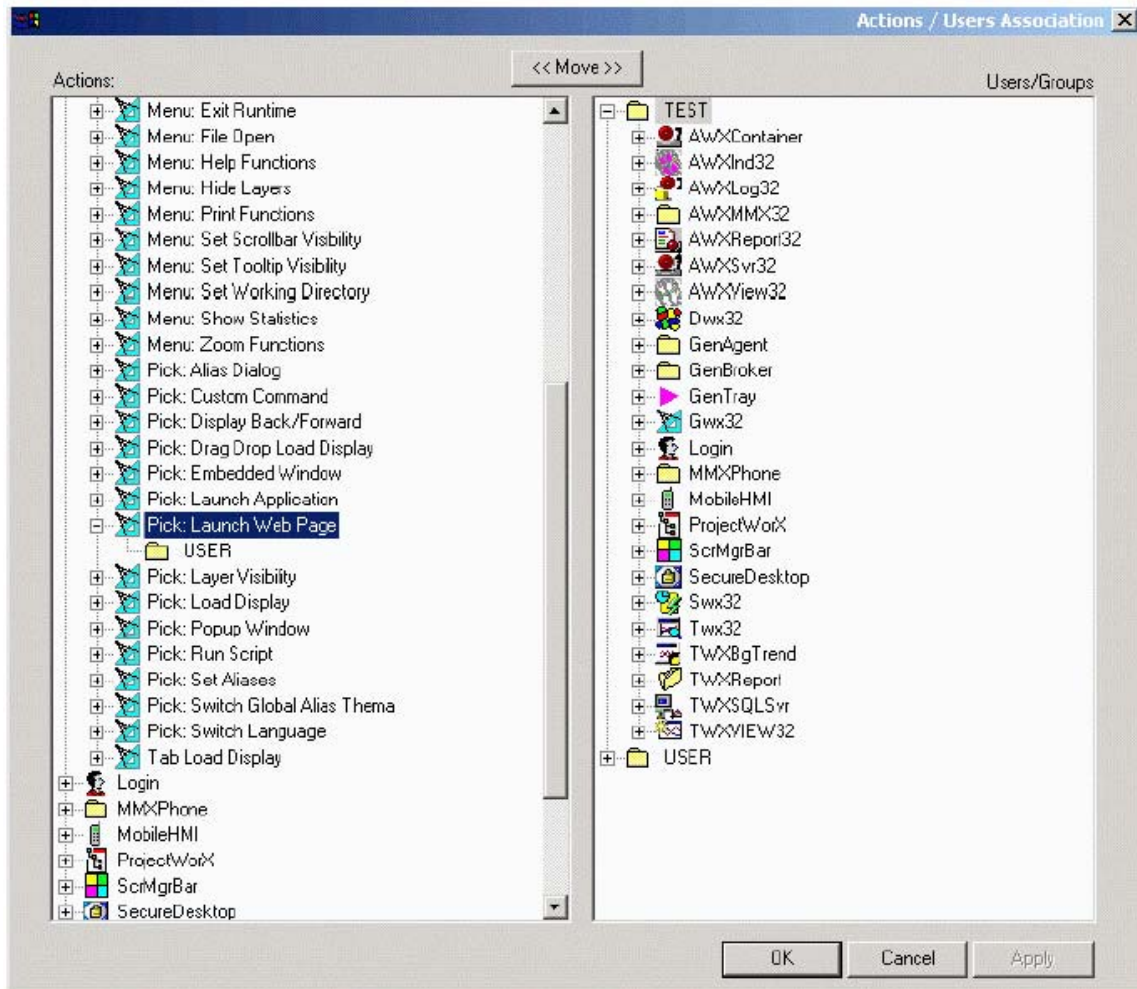
Setovanje radnog direktorija

Izabirući **Set Working Directory** iz **Tools** menija otvara **Set Working Directory** dijalog boks, pokazan na narednoj slici, koji omogućava konfigurisanje kastomiziranog direktorija u koji želimo da pohranimo sve konfiguracione fajlove.



Sigurnost

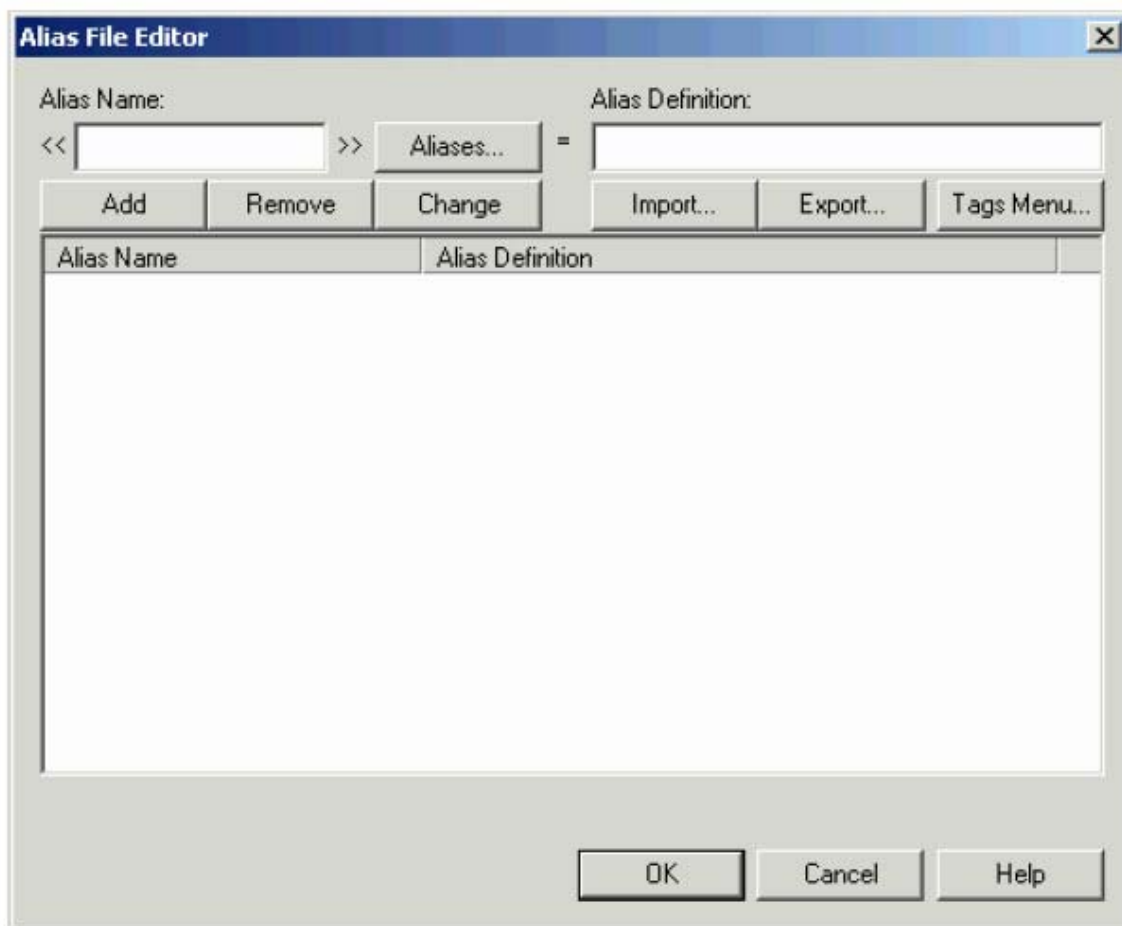
GraphWorX32 ima interfejs sa Iconics Security Serverom i u ovoj verziji (7.1) podržava siguronosne akcije prikazane u narednom boksu:



GraphWorX32 siguronosne akcije

Fajl editor lokalnih aliasa

Komanda **Alias File Editor** na **Tools** meniju otvara **Alias File Editor** dijalog boks, pokazan na narednoj slici, koji omogućava da dodamo, promjenimo ili uklonimo aliase. Klikanjem na **Aliases** taster pokazaće se svi aliasi za displej. Korisnik takodje može izabrati da uveze (**Import**) ili izveze (**Export**) jedan alias. U polju Alias Definition, korisnik može definirati aliase koristeći **Tags Menu** taster, koji dozvoljava da izaberemo tagove, izraze , lokalne varijable simulacione varijable i globalne aliase.



Dijalog boks fajl editora lokalnih aliasa

Publokovanje na HTML

GENESIS32 **Web Publishing Wizard** omogućava nam da "izvezemo" GraphWorX32 (*.gdf) displeje u HTML fajlove i/ili publikujemo HTML fajlove na Web Server (u okviru LANa ili Interneta). Pri publikovanju displeja na Web Server, WebHMI koristi HTML da referencira *. gdf fajlove u Internet-omogućenom formatu. Jedanput kada je GraphWorX32 displej "izvezen" u HTML fajl , i onda publikovan (tj. uplodiran) na Web server, klijent mašine ga mogu browsovati pomoću Internet browsera, kao što je MS IE ili Netscape Navigator. Svaki GraphWorX32 displej može se posmatrati kao Web stranica

Osnovni koraci u publikovanju GraphWorX32 displeja

Publikovanje GraphWorX32 displeja u HTML uključuje slijedeće bazične korake:

1. Kreiranje GraphWorX32 displej (*.gdf) fajla na razvojnoj radnoj stanici sa instaliranim GraphWorX32 softwareom.
2. Korištenje Web Publishing Wizarda da generira HTML fajl i publikuje fajl na Web Server (URL) adresu

3. Na Web Server PC (sa GENESIS32 instaliranim), startati GenBroker Server sa GenTray.
4. Na klijent PC-ju, otvoriti Web browser, kao napr. MS IE, i browsovati na URL adresu HTML fajla. Klijent PC downloadira HTML fajl sa Web servera. Web server isporučuje sve neophodne komponente (tj. Active X kontrole), na klijentov PC Web browser, koji onda izvršava GraphWorX32 displej direktno unutar svog browsera. Na ovaj način možemo gledati GraphWorX32 displej u realnom vremenu kao Web stranicu.

Izvorni (source) kod publikovanog HTML fajla sadrži reference na zahtjevane "plug-ins" koji su potrebni da isporuči GraphWorX32 Viewer ActiveX kontrole ka klijentu. GraphWorX32 Viewer ActiveX je potreban da se izvršava *.gdf fajl preko mreže. GraphWorX32 displej sa svoje strane nije ustvari "konvertovan" u HTML. Umjesto toga, displej (*.gdf) fajl je referenciran u HTML kodu od starne ActiveX plug-ina.

Da se predje na korak 2 iz gornje skale, naredni MS IE primjer pokazuje dio generiranog HTML izvornog koda koji referencira *.gdf fajl . **GWXview32.cab** fajl je referenciran u **codeBase** polju da isporuči "plug-in" za GraphWorX32 Viewer ActiveX. Ovo sa svoje strane otvara *.gdf displej. U narednom primjeru, *.gdf displej fajl se zove **Building1.gdf**, kako je pokazano u primjeru u **DisplayName** polju parametra.

```
<OBJECT classid=clsid:98A5DDE3-563B-11CF-A343-487C03C10000
codeBase="http://www.myserver.com/webhmi/cabs/GWXview32.cab"
id=GWXview321 style="HEIGHT: 420px; WIDTH: 620px" height=420
width=620>
<param name="_Version" value="65537">
<param name="_ExtentX" value="16404">
<param name="_ExtentY" value="11113">
<param name="_StockProps" value="160">
<param name="BorderStyle" value="1">
<param name="Appearance" value="1">
<param name="DisplayName"
value="http://www.myserver.com/webhmi/Building1.gdf">
<param name="UseAmbientBackColor" value="0">
<param name="AutoStartRuntime" value="-1">
<param name="OverrideScrollbarSettings" value="1">
<param name="VerticalScrollbar" value="0">
<param name="HorizontalScrollbar" value="0">
</OBJECT>
```

Izvoz *.gdf fajla u HTML fajl ne referencira samo **GWXview32.cab** fajl , kao što je pokazano u gornjem primjeru, nego takodjer referencira bilo koji drugi "plug-in" .cab fajl koji se zahtjeva za druge ActiveX komponente koje mogu biti uronjene unutar *.gdf fajla (napr. TrendWorX32 ili AlarmWorX32 Viewer ActiveX kontrole).

Medjutim, prije nego što *.gdf fajl može komunicirati sa živim OPC podacima **lcoSetServer.cab** fajl "plug-in" , koji je potreban za sigurnost i licenciranje, kao i **GenBroker configuration (*.gbc, ili *.gbx)** fajl , koji sadrže setinge za konfiguraciju mreže za OPC komunikacije, moraju takodjer biti isporučeni ka klijent PC-ju.

IcoSetServer.cab fajl i *.gbc ili *.gbx fajl su referencirani u HTML izvornom kodu. Na ovaj način sigurnost i informacija o licenciranju su na raspolaganju kad god je komponenta (kao napr. Alarm Viewer ActiveX) downloadovana. Uzorak (sample) HTML izvornog koda u nastavku pokazuje kako su ovi fajlovi referencirani u kodu.

```
<object id="SetServer2" classid="clsid:57802C16-9A15-11D4-B2A8-0090272E599B"
codeBase=http://www.myserver.com/WebHMI/cabs/IcoSetServer.cab
height=28 width=17>
  <PARAM NAME="CfgName"
  VALUE="http://www.myserver.com/WebHMI/Samples/Default.gbc">
</object>
```

GENESIS32 Web Publishing wizard u GraphWorX32 automatski preuzima brigu za sve neophodne HTML kod reference. MS Internet Explorer (IE) , koristi *.cab fajlove , dok Netscape Navigator koristi *.dpl fajlove.

Opaska:

Ako koristimo višestruke okvire (frames) za WebHMI strane (tj. glavni "navigacioni" okvir za browsovanje izmedju strana, i "content" okvir koji sadrži tijelo strana), treba obezbjediti da su IcoSetServer.cab fajl i *.gbc ili *.gbx fajl korektno referencirani u HTML izvornom kodu, za glavni okvir. Na ovaj način sigurnost i licenčna informacija su na raspolaganju kad god se komponenta downloaduje.

Isporučivanje neophodnih Web komponenata klijentu

WebHMI je dizajniran da radi sa **Zero Install** i filozofijom tankog klijenta (**Thin client**). Ovo znači da klijent PC nema ništa nego Windows, IE , ili Netscape Navigator. Dakle, sve neophodne Web komponente moraju se daljinski isporučiti i biti bezčujno instalirane. Tip i broj komponenti koje su potrebne za isporuku na klijent PC se određuje sa sadržajem GraphWorX32 displeja. Na primjer, displej sa uronjenim Trend Viewerom zahtjeva isporuku odgovarajućeg TrendWorX32 Viewer ActiveX Web komponente, ali ne i AlarmWorX32 Viewer ActiveX. Sa druge strane, displej sa uronjenim Alarm Viewerom zahtjeva odgovarajući AlarmWorX32 Viewer ActiveX ali ne i TrendWorX32 Viewer ActiveX,

Proces isporuke može biti vrlo polagan u terminima brzine mreže. Da se izbjegne rasipanje vremena, prvi zahtjev je da se isporuči samo zahtjevana komponenta. Proces isporuke je striktno spojen sa HTML kodom. Unutar HTML koda, postoje reference na *.dpl ili *.cab fajlove, što je bazni način da se isporuče komponente. Kao rezultat, izgradnja HTML strane je kritičan korak. Web Publishing wizard gleda unutar displeja da razumije koje su komponente potrebne sa strane klijenta i prema tome gradi odgovarajući HTML kod.

Zahtjevi na stanicu klijenta

Da bi mogao gledati publikovane GraphWorX32 displeje, klijent PC mora imati instaliran Web browser.

Zahtjevi na stanicu za razvoj web aplikacije

Razvojna stanica zahtjeva GraphWorX32 verzija 6.1 ili veća

Zahtjevi na stanicu Web servera

Web server PC mora imati slijedeće programe instalirane:

- WebHMI
- GENESIS32 (sa GenBrokerom)
- Za WinNT , Internet information Server (IIS) ili Personal Web server
- Za Win 2K i Win XP, Internet Information Server (IIS)

Podrška za višestruke displeje

Predpostavimo da imamo glavni GraphWorX32 displej (*.gdf) fajl , koji je linkovan na druge displej fajlove (t.j. svaki displej sadrži pick akcije, kao što je Load Display ili pop-up prozor, koji ukazuje na druge displej fajlove). Kada publikujemo glavni displej u HTML fajl , tada želimo da su svi linkovi i reference na druge zavisne displeje funkcionalni kada je HTML fajl downloadovan na browser klijenta. Web publishing wizard čini ovo mogućim detektujući sve međusobno vezane displej fajlove, stalno prolazeći ukруг (looping) kroz sve dinamičke akcije , i provjeravajući za pick akcije u kojima je ime fajla specificirano kao jedan od njegovih parametara. Slijedeće pick akcije su podržane za podršku višestrukim displejima GraphWorX32 Web publikovanja:

- load display
- uronjeni prozor (embedded window)
- iskačući prozor (popup window)
- drag/drop load
- setovanje aliasa
- alias dijalog (alias fajlovi specificirani putem **Set Aliases** dijalog boksa u GraphWorX32).

LoadTabs osobina displeja je takodjer podržana za publikovanje višestrukih GraphWorX32 displeja.

Podrška za uronjenu ActiveX kontrolu

Kod rada sa glavnim GraphWorX32 displej fajlom (*.gdf) koji je linkovan na druge displej fajlove (kao što je opisano u prethodnom pasusu), predpostavimo da smo uključili ActiveX kontrolu (napr. Alarm Viewer ActiveX ili Trend Viewer ActiveX) u svaki od zavisnih displej fajlova. Takodjer, predpostavimo da smo generisali i publikovali HTML fajl samo za glavni displej, ali želimo da ovaj HTML fajl trigeruje simultani download i instalaciju svih ActiveX kontrola uronjenih unutar svih međupovezanih GraphWorX32 displej fajlova.

Podrška Web publishing wizarda za uronjenu ActiveX kontrolu (embedded ActiveX control), čini ovo mogućim, dozvoljavajući nam da gledamo ActiveX kontrole u svim displejima iz klijentovog Web Browsersa.

Korištenje Web Publishing wizarda

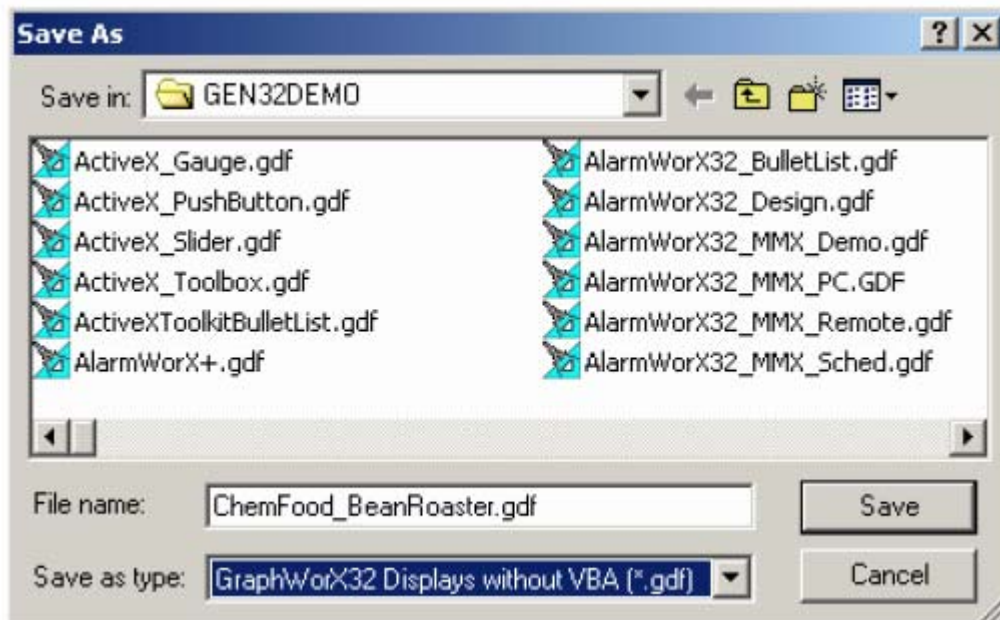
Wizard za publikovanje na Webu , izvršava dvije osnovne operacije:

1. Wizard kreira HTML fajl baziran na GraphWorX32 displej fajlu (*.gdf) kojeg je specificirao korisnik
2. Wizard nakon toga bilo "izvozi" (pohranjuje) HTML fajl u direktorij koji je specificirao korisnik na lokalnom drajvu, ili "publikuje" (upload) HTML fajl na URL adresu Web servera koju je specificirao korisnik (bilo preko intraneta ili interneta).

Startanje wizarda za publikovanje na Webu

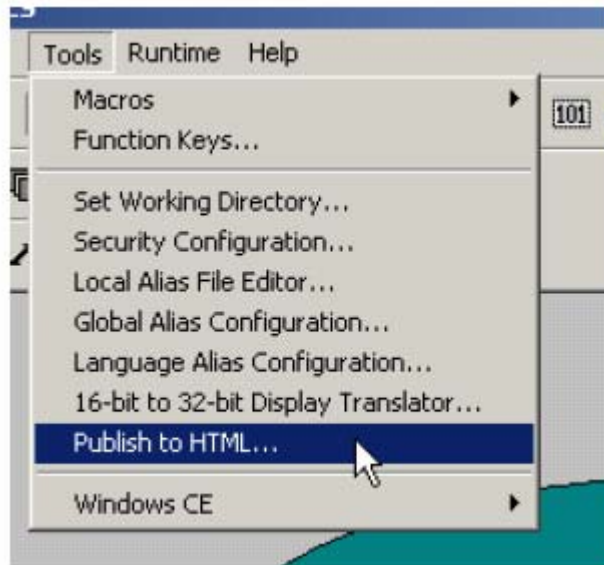
Da bi se lansirao Web Publishing wizard u GraphWorX32, uraditi:

1. Napuniti gotov ili kreirati GraphWorX32 displej fiel (*.gdf).
2. Nakon što smo ili kreirali novi displej fajl ili otvorili postojeći, izabrati **Save AS** iz **File** menija. Otvoriće se dijalog boks, u kojem možemo specificirati ime i lokaciju fajla kojeg ćemo pohraniti. Važno je kod pohranjivanja da izaberemo " GraphWorX32 Displays without VBA (*.gdf) , kao što je pokazano na narednoj slici, pošto Web Publishing wizard ne podržava MS Visual Basic za aplikacije (VBA).



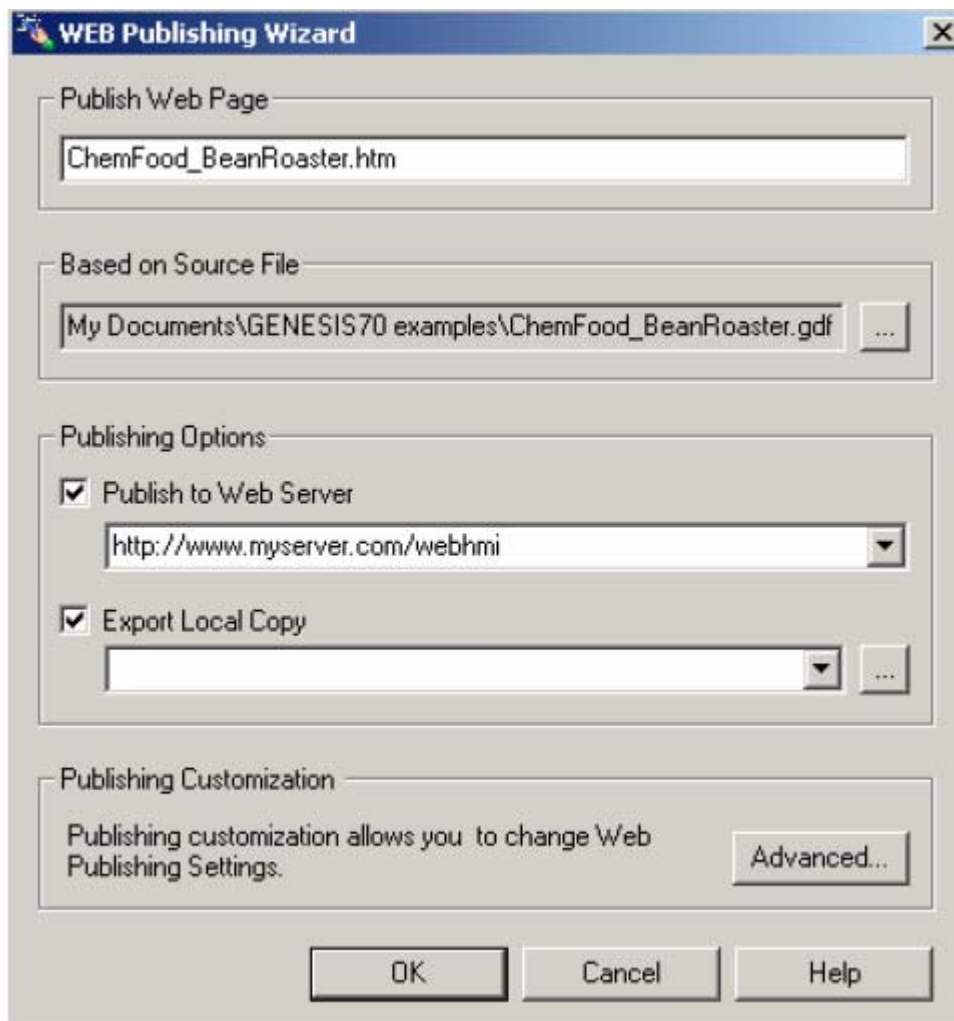
Pohranjivanje displeja kao Non-VBA fajla.

3. Izabrati **Publish to HTML** iz **Tools** menija, kao što je pokazano na narednoj slici:



Startanje Web publishing wizarada u GraphWorX32

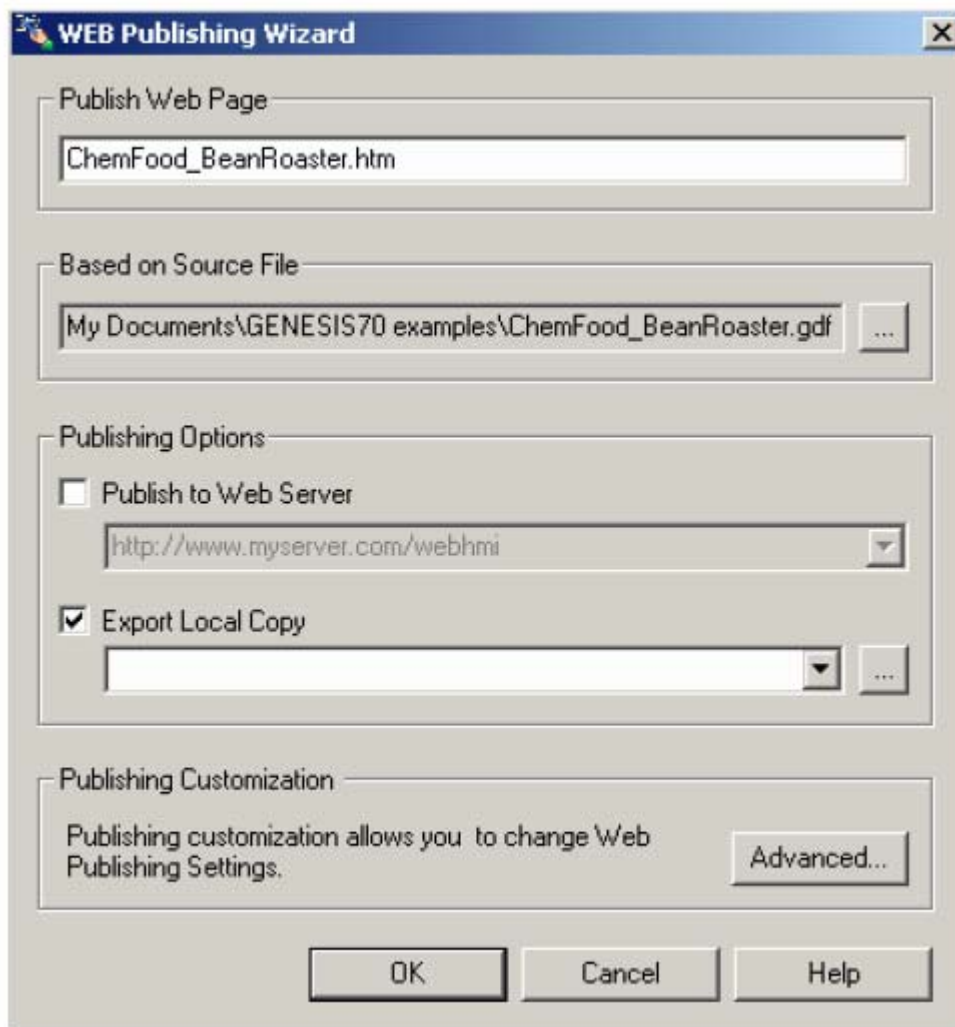
4. Ovo će startati **Web Publishing Wizard** konfiguracioni dijalog boks, kakav je pokazan na narednoj slici. Ovaj dijalog boks služi kao interfejs pomoću kojeg možemo publikovati (izvesti) GraphWorX32 displej fajlove na Web.



Lokalni izvoz displej fajla

Da bi se pohranio GraphWorX32 displej u direktorij na lokalnom drajvu, uraditi:

1. U polju **Based on Source File** Web Publishing wizarda, treba specificirati ime GraphWorX32 displeja, koji će se izvesti, kao je pokazano na narednoj slici.

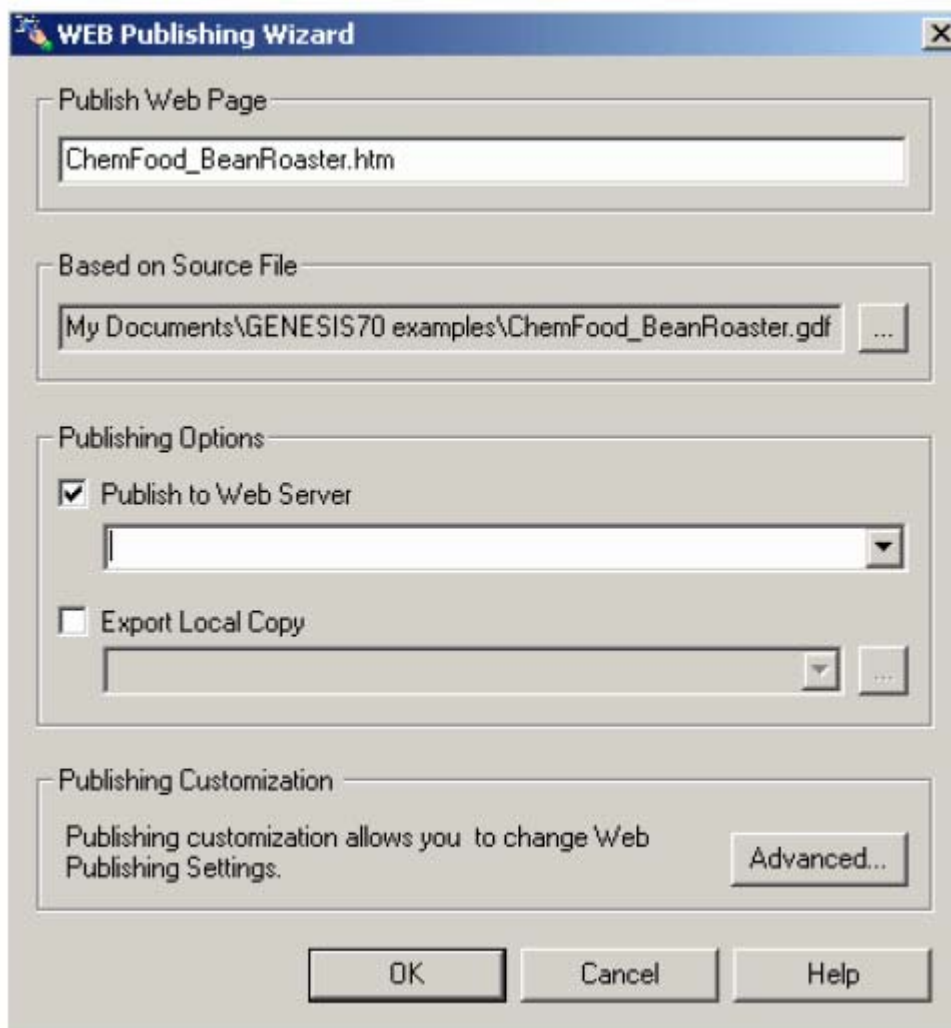


Opaska : Obadvije akcije tj. i **Publish to Web Server** i **Export Local Copy** se mogu izvršiti u isto vrijeme.

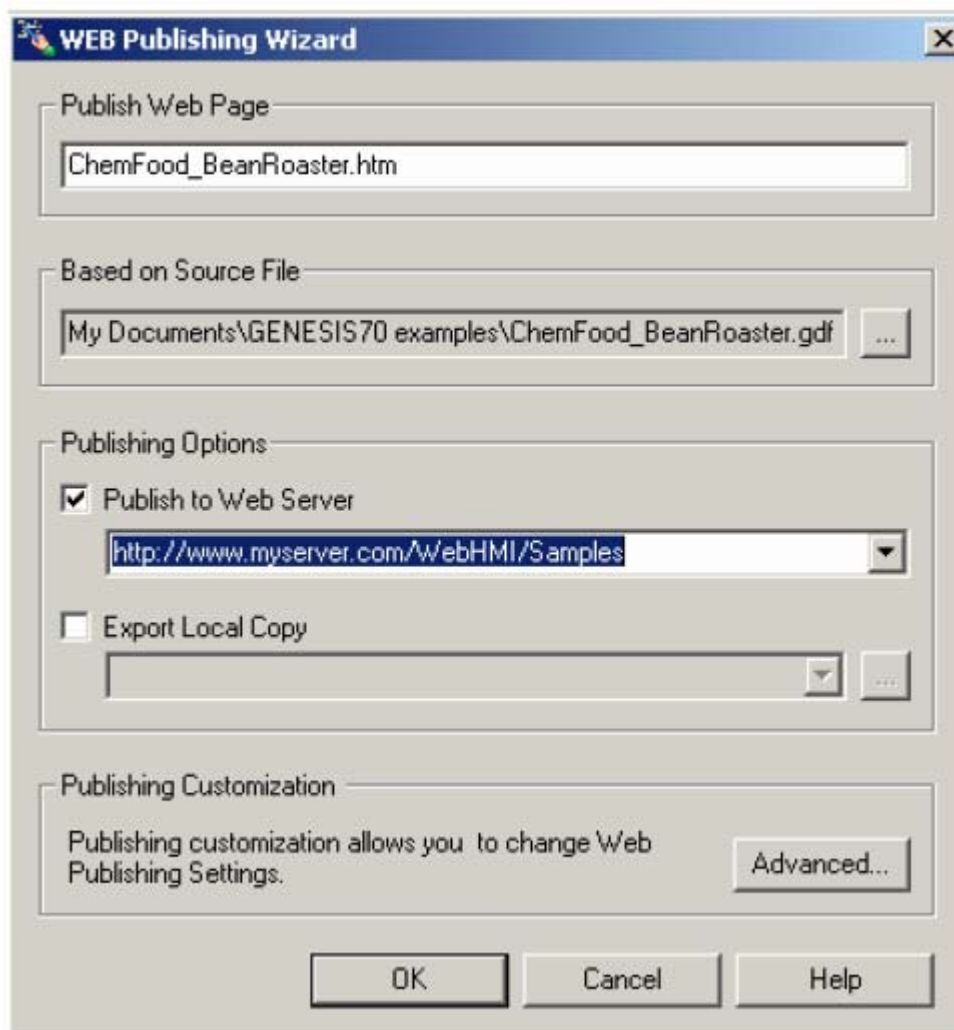
Publikovanje displej fajla na Web Server

Da bi se publikovao HTML fajl na direktorij na Web Serveru (t.j. putem Interneta ili intarneta), uraditi:

1. U polju **Based on Source File** u Web Publishing wizard dijalog boksu, treba specificirati ime GraphWorX32 displeja koji će se izvesti, kao što je pokazano na narednoj slici:



2. U polju **Publish Web Page** , specificirati ime HTML fajla koji će se kreirati.
3. U **Publishing Options** polju , čekirati **Publish to Web server** ček boks, i unjeti URL adresu Web servera sa kompletnim direktorijem , označavajući gdje tačno želimo da publikujemo HTML fajl , kao što je to pokazano na narednoj slici.



Kastomizacione opcije kod publikovanja

Web publishing wizard sadrži neke kastomizacione opcije za izvoz/publikovanje HTML fajlova. Klikanjem na **Advanced** taster , otvoriće se **Web Publishing Properties** dijalog boks, pokazan na narednoj slici, koji sadrži slijedeće tabove;

- General settings
- Server switching support
- screen resolution settings
- publishing options



Podrška za preključenje servera (server switching support)

Ovaj tab pokazan na narednoj slici, omogućava nam da omogućimo ili onemogućimo podršku GenBrokera za izvoz/publikovanje HTML fajla. Ovdje mi možemo specificirati koji GenBroker konfiguracioni fajl (*.gbc ili *.gbx) će biti aktiviran. GernBroker konfiguracioni fajl uspostavlja setinge za OPC komunikacije izmedju klijenata i Web servera.



Postoje tri moguće opcije za podršku GenBrokera:

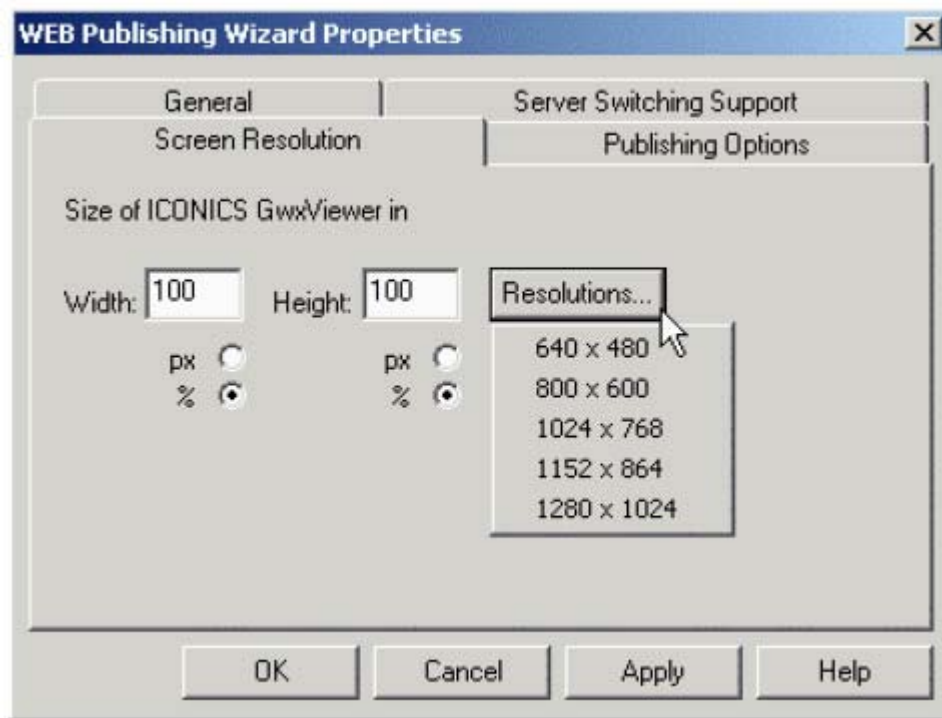
- **None** : GenBroker podrška nije aktivna
- **As Configured in WebHMI Settings**; Ovo podešenje koristi default URL adresu GenBroker konfiguracionog fajla (*.gbc ili *.gbx) kako je specificirano kod instalacije WebHMI softwarea.
- **User defined**; kada se izabere ova opcija, tada polje GenBroker Configuration Fajl URL postaje dostupno , omogućavajući nam da specificiramo URL adresu GenBroker konfiguracionog fajla (*.gbc ili *.gbx)

Klijenti mogu primiti podatke iz različitih WebHMI servera, ali se komunikacija može uspostaviti u svakom trenutku vremena samo sa jednim serverom. Server na koji će se klijent konektirati zavisi od tipa informacije koju klijent zahtjeva. Ako napr. Server A ne sadrži komponente koje su neohodne da bi mogao odgovoriti na zahtjev klijenta, zahtjev će biti prosljedjen do Servra B ili Servera C, itd.

Postavljanje rezolucije ekrana

Tab **Screen Resolution** u **Web Publishing Properties** dijalog boks, pokazan na narednoj slici, određuje rezoluciju ekrana i veličinu GraphWorX32 Viewer ActiveX kontrole, koja je referencirana u generisanom HTML fajlu i zatim downloadovana na klijent PC kada HTML fajl gleda u klijentovom Web Browseru. Korisnik može

specificirati veličinu ekrana u **Width** i **Heights** poljima u iznosima pixela (px) ili procentualno (%). Kliknuti na **Resolutions** taster da se izabere iz pop-up menija standardne rezolucije ekrana , kako je pokazano na narednoj slici:



Opcije publikovanja

Tab opcija publikovanja, pokazana na narednoj slici, pojačava proces publikovanja. Po defaultu, Web Publishing wizard obezbedjuje podršku za detekciju odgovarajućih fajlova. Detekcija višestrukih displeja može biti prilično dugačak proces. Korisnik može ubrzati ovaj proces time što će onemogućiti opciju **Enable Multiple Display Support**. Ovo je naročito korisno ako smo već publikovali fajlove iz projekta na Web server, i sada želimo da ažuriramo displej koji smo promjenili.

Mehanizam detekcije višestrukih displeja obezbedjuje da će Web strana biti korektno publikovana za izvorni fajl . Onemogućavanjem **Multiple Display Support** , može se prouzrokovati publikovanje nekompletne Web stranice, što može rezultirati u neočekivanom ponašanju.

Zbog toga, sugerise se da se ne dozvoli publikovanje web stranice. To se može uraditi tako da se omogući opcija **Publish Display File Only**.



Gledanje lokalno izvezenih HTML fajlova

Da bi gledali HTML fajl koji je izvezen na lokalni direktorij, jednostavno treba browsovati do staze lokalnog direktorija i kliknuti na fajl da se otvori u Web browseru. Možemo takodjer otvoriti i Web Browser i kopirati stazu direktorija fajla u polje **Addresses** na Web Browseru, kao što je pokazano na narednoj slici. Kompletna staza direktorija je napr. "C:\Documents and Settings\Administrator\My Documents\Web Publishing Example \ChemFood_ BeenRoaster.htm."



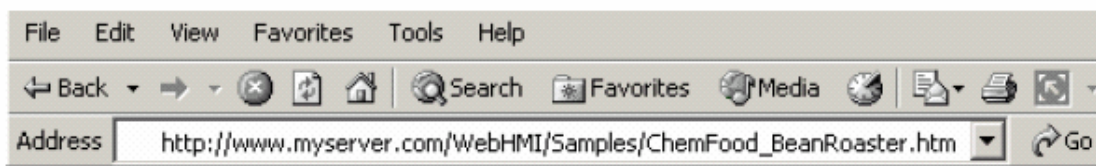
Gledanje pulikovanih HTML fajlova

Prije nego što publikovani HTML fajl može biti downloadovan sa WEebHMI Servera, moramo postaviti GenBroker aktivnim na WebHMI server čvoru koristeći genTray:

1. Lansirati GenTray sa **Programs>ICONICS GENESIS32 >Tools>GenTray**
2. Kada se GenTray otvori, pojaviće se mala trouglasta ikona u tool tray-u Windowsa. Kliknuti i izabrati **GenBroker>Start** iz pop-up menija kao što je pokazano na narednoj slici. Ovo aktivira GenBroker server.



3. Kada GenBroker server se izvršava, možemo gledati HTML fajl koji smo publikovali na Web Serveru sa bilo kojeg klijent PC-ja pomoću Web Browsersa. Da bi se gledao publikovani HTML fajl , otvoriti Web browser i kopirati URL adresu fajla u polje **Addresses** Web browsera, kako se vidi na narednoj slici.



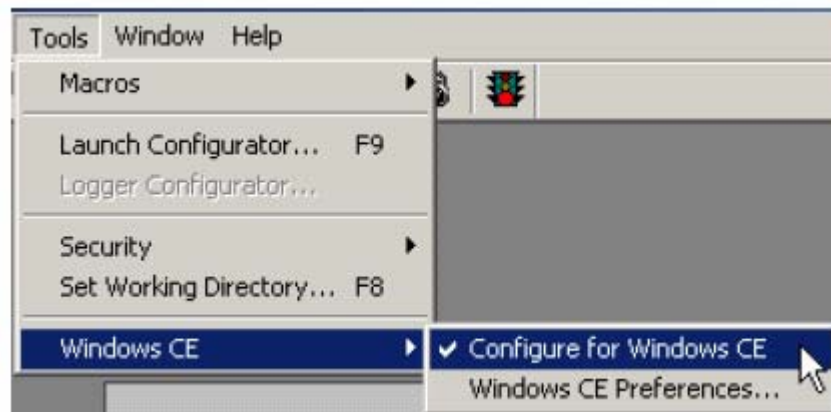
GraphWorX32 displej fajl je referenciran u HTML kodu tako da se displej može gledati kao Web strana. Web browser klijenta jednostavno downloaduje HTML fajl u kojem *.gdf fajl je referenciran. Sve što klijent treba je Web browser , nije potrebno da se ima GENESIS32 instaliran na klijent mašini. Web strana je downloadovana sa Web servera kroz intranet/internet i pojavljuje se u klijentovom prozoru Web browsera. Displej je u realnom vremenu kao da gledamo runtime displej u GraphWorX32 na serverskoj mašini , OPC tag vrijednosti se mjenjaju dinamički u displeju.

Konfiguriranje za Windows CE u GENESIS32

Svi displej fajlovi za Pocket GENESIS32 su prvo konfigurisani na desktop PC-ju koristeći GENESIS32 aplikaciju, kao napr. GraphWorX32, TrendWorX32 i AlarmWorX32, i nakon toga pohranjeni u formatu koji je kompatibilan sa Pocket PC i Windows CE. Displej fajl je nakon toga downloadovan u Pocket PC ili Windows CE uređaj, da bi se posmatrao u runtime-u. Kada se otvori displej fajl u odgovarajućoj Pocket GENESIS aplikaciji (tj. Pocket GraphWorX, Pocket TrendWorX, ili Pocket AlarmWorX) u runtime modu, možemo izvršiti sve runtime funkcionalne dinamike koje

su bile dodate za vrijeme konfiguracije. Sa nekoliko izuzetaka, Pocket GENESIS aplikacije imaju istu bazičnu runtime funkcionalnost kao i njihove partnerske GENESIS32 aplikacije na PC-jevima.

Da se konfigurira za Windows CE u GraphWorX32, TrendWorX32 i AlarmWorX32, izabrati **Windows CE – Configure for Windows CE** iz **Tools** menija, kao što je pokazano na narednoj slici.



Kada smo završili konfigurisanje displeja, izabrati **Save As** iz **File** menija. Dati fajlu ime i izabrati jedan od slijedećih tipova fajlova iz **Save As Type** boksa:

- In GraphWorX32, select either "GraphWorX32 Displays for WinCE (*.gdc)" or "GraphWorX32 Templates for WinCE (*.gdc)."
- In AlarmWorX32, select "AWXview32 CE File (*.awv)."
- In TrendWorX32, select "TWXviewerCE File (*.tce)."

Udaljene konekcije

Da bi se daljinski konektirali na OPC servere preko mreže, moramo instalirati i koristiti GenBroker.

Download GENESIS32 konfiguracionih fajlova u Pocket PC.

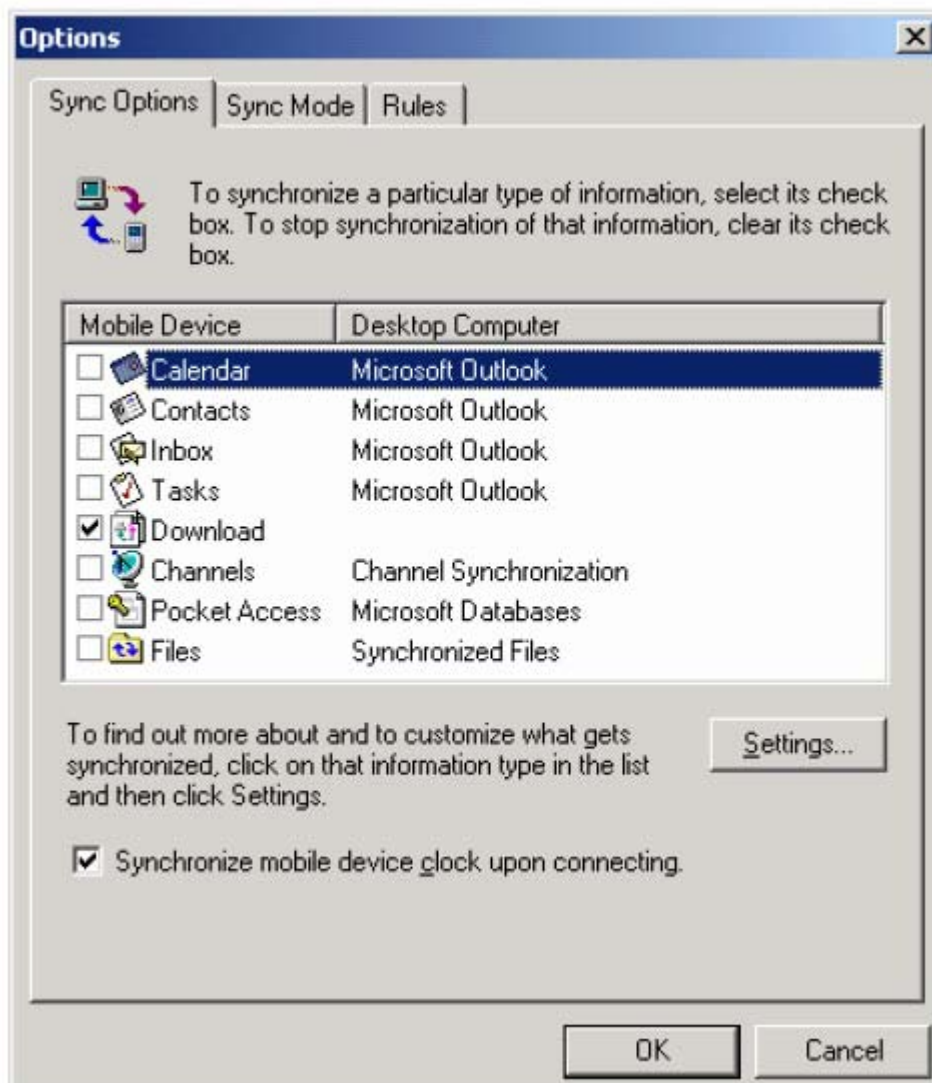
Nakon što smo konfigurisali aplikaciju u GENESIS32 i spasili je kao Pocket GENESIS displej fajl , možemo koristiti fajl download u Windows CE utility u GENESIS32 da downloadujemo konfiguracione fajlove iz GraphWorX32 , TrendWorX32 i AlarmWorX32 u Pocket ili CE uređaj. Kada razvijamo konfiguracioni fajl za Pocket GENESIS aplikaciju, ove osobina nam omogućava da downloadujemo konfiguracioni fajl u Pocket PC ili Windows CE uređaj. Ova fajl download funkcija koristi MS ActiveSync da se spoji sa CE uređajem.

Setovanje za download

Download u CE uređaj zahtjeva module i na strani desktop PC ili radne stanice kao i na strani Pocket PC ili CE uređaja, pošto MS ActiveSync servisi se koriste za spajanje i autentikaciju CE uređaja.

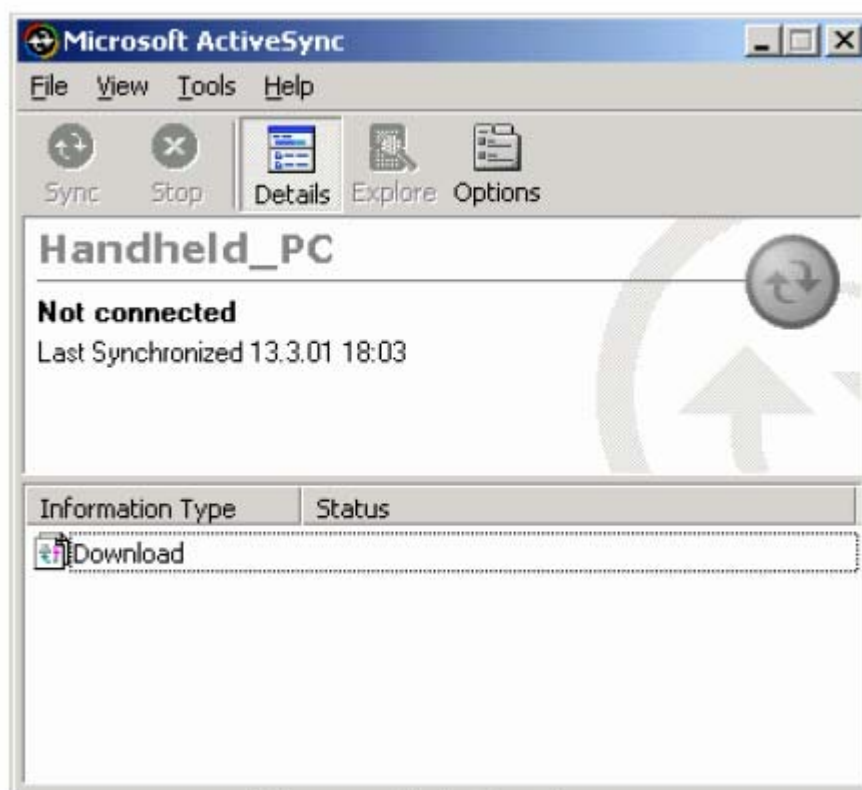
Konfigurisanje desktopa

Desktop je jedini dio koji treba biti konfiguriran. Ako je fajl download alat propisno instaliran, bit će izlistan u ActiveSync Manager listi od Active Sync modula u **Sync Options** tabu od **Options** dijalog boksa. Moramo omogućiti ActiveSync module fajl download u ActiveSync Manageru čekiranjem **Download** boksa, kao na narednoj slici:



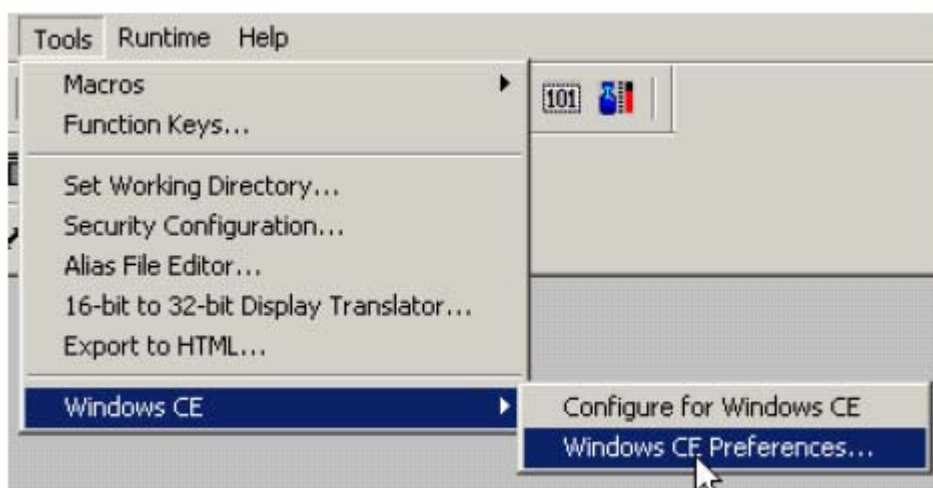
Asocijacija sa CE uređajem se mora kreirati da se sinhroniziraju download ActiveSync moduli, kao što je pokazano na narednoj slici. Kada CE uređaj odgovori bez problema, download sinhronizacioni modul je aktivan.

Kada je omogućen, fajl download alat odmah downloaduje konfiguracioni fajl za Windows CE u CE uređaj. Kada je onemogućen, konfiguracioni fajl će biti samo pohranjen.



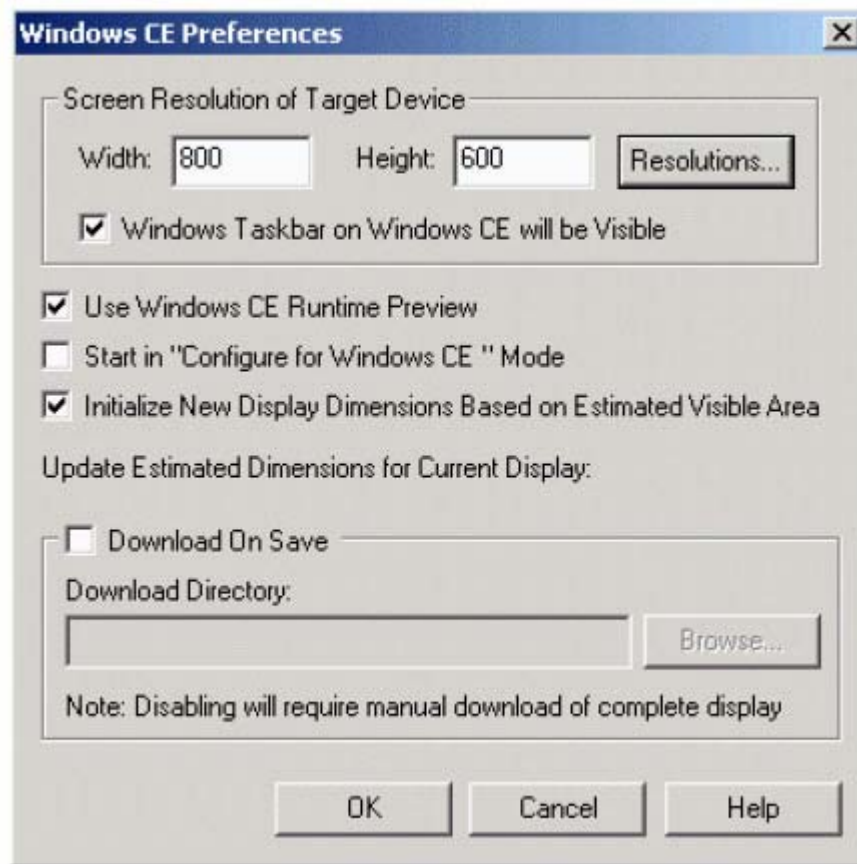
Konfigurisanje file download aplikacije

GraphWorX32, TrendWorX32 i AlarmWorX32 imaju svoje vlastite verzije fajl download alata. U ovim aplikacijama, izabrati **Tools>Windows CE> Windows CE Preferences**, kao što je prikazano na narednoj slici:



Startanje Windows CE download alata

Ovo otvara **Windows CE Preferences** dijalog boks, koji omogućava da se omogući ili onemogući alat za download. Da bi se omogućio, treba čekirati **Download on Save** ček boks. Zatim izabrati ciljni direktorij na CE uređaju, kao na narednoj slici:



Download file u CE uređaj

Kada završimo sa konfigurisanjem displej fajla i želimo da ga pohranimo na Windows CE, treba izabrati **Save As** iz **File** menija. Pohraniti fajl kao:

- Kod GraphWorX32, unjeti ime fajla i izabrati ili "GraphWorX32 Display for WinCE" (*.gdc), ili "GraphWorX32 Templates for WinCE (*.gdc)".
- U AlarmWorX32, unjeti ime fajla i izabrati "AWXWiew32 CE fajl (*.awv)".
- U TrendWorX32, unjeti ime fajla i izabrati "TWXviewer CE fajl (*.tce)".

Skript mašina i skript editor za VBScript i Jscript

GraphWorX32 ima integriran Visual Basic za aplikacije (VBA) Editor, koji omogućava da se unosi i generiše VBA skript. Ovo je moćan alat koji dozvoljava laganu ekspanziju funkcionalnosti GraphWorX32. Da bi se izvršavao VBA skript, PC mora imati instaliran VBA runtime engine. Ova engine ne može biti isporučena putem Interneta/Intraneta thin-client mašini, pošto zauzima dosta disk prostora i zahtjeva duga vremena downloadovanja. Zbog ovoga, je ICONICS uklonio VBA podršku iz WebHMI i drugih "lahkih" proizvoda, kao što je Pocket GraphWorX. Zbog toga nije moguće koristiti snagu VBA i prednosti thin-client tehnologije u WebHMI. Zbog toga mnogi korisnici razvijaju

dva različita seta displeja, jedan set sa VBA za desktop radne stanice, i drugi set bez VBA za WebHMI thin klijenta. Ovo zahtjeva dodatan rad potreban da se izbaci VBA kod i redizajnira displej.

Kao rješenje za ovaj problem, GraphWorX32 posjeduje novu skript mašinu koja može parsirati (izdvajati) VBScript i Jscript kod. Ova engine je malog obima i lako se može distribuirati preko Interneta do bilo koje thin klijent mašine. Kada korisnik planira da koristi skripting tehnologiju da bi poboljšao GraphWorX32 karakteristike a također planira i da koristi thin-client tehnologiju, kao što je WebHMI, treba koristiti novi uronjenu skripting mašinu i pisati svoj kod bilo u VBScript ili Jscript umjesto u Visual Basic za aplikacije. Na taj način on može dizajnirati jedan jedinstveni displej i sa skript i WebHMI podrškom, bez da mora da radi sa više tipova displeja.

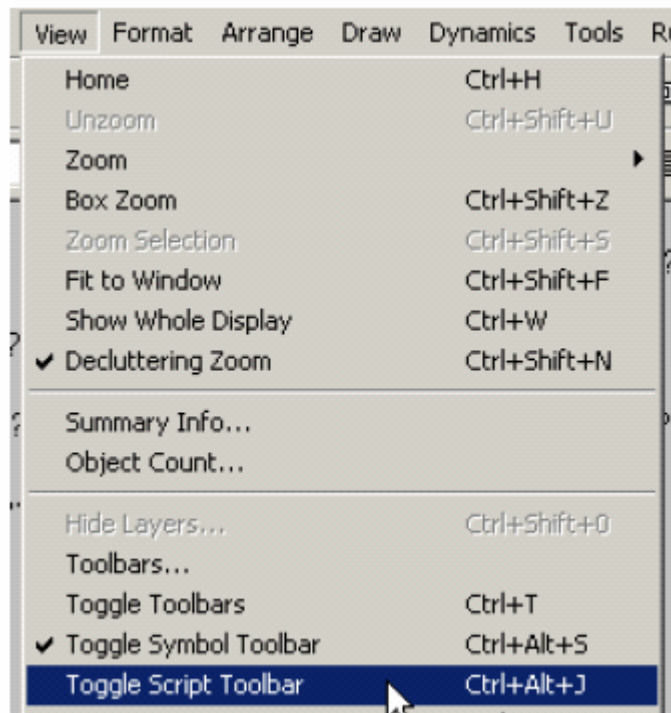
Nova script engine za VBScript i Jscript nema sve mogućnosti kao puna VBA mašina, tako da neke osobine nisu podržane. Postoje mnoga ograničenja, ali vjerovatno najveće je to da nije moguće da se obraduju događaji koji dolaze od ActiveX ili drugih kontrola koje su uronjene unutar GraphWorX32 displeja.

Nova skripting mašina je u stanju da obradjuje najvažnije displej događaje kao i sve pick akcije, ali u ovoj verziji (7.1) nije u stanju da pokrene nove događaje.

Dodatno ograničenje je da VBScript i Jscript ne podržavaju formulare (forms), tako da nije moguće kreirati elemente interfejsa sa korisnikom drugačije od jednostavnog boksa poruke i ulaznog boksa (message box , input box).

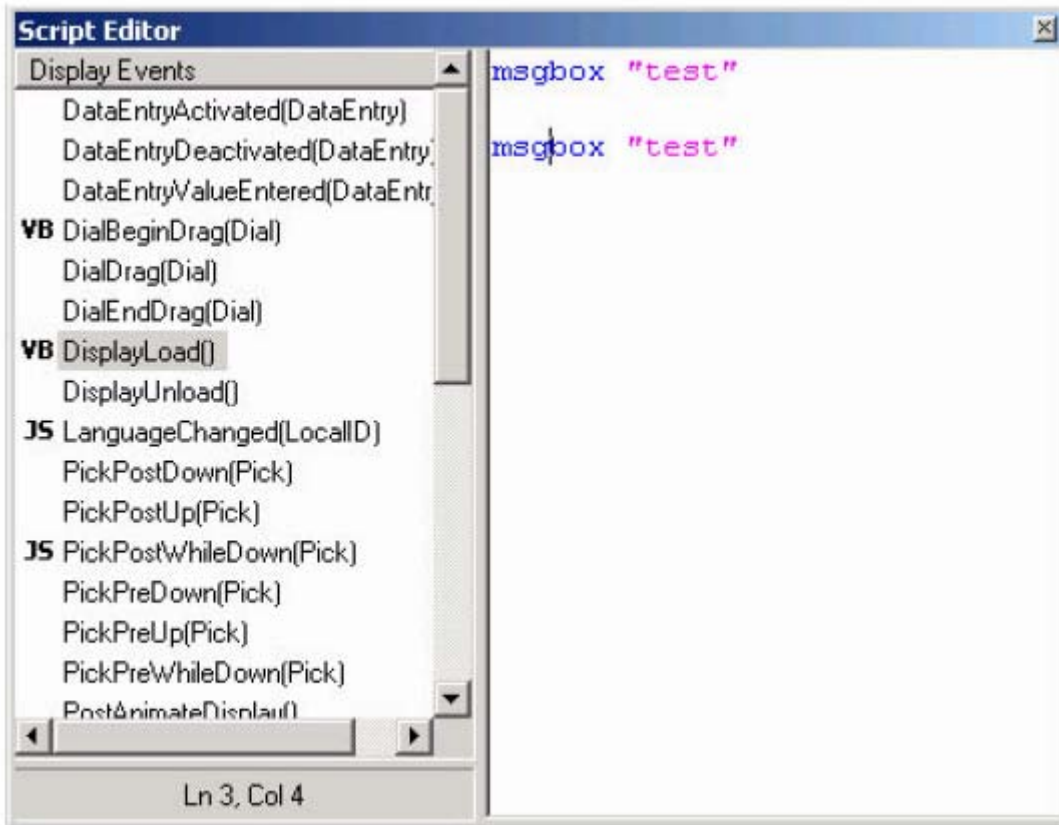
Prikazivanje skript editora

Integrirani skript editor se pojavljuje kao alatna letvica u GraphWorX32 displeju. Može se povlačiti oko ekrana kao lebdeći (floating) prozor ili se može priključiti na jedno mjesto (docked) od četiri strane displeja. Skript editor je tkz. "inetgrirana alatna letvica " (integrated toolbar) tako da nema potrebe da se posebno "otvara". Umjesto toga može da se pokaže ili sakrije izabirući iz **View** menija **Toggle Script Toolbar** , kao što je pokazano na narednoj slici:

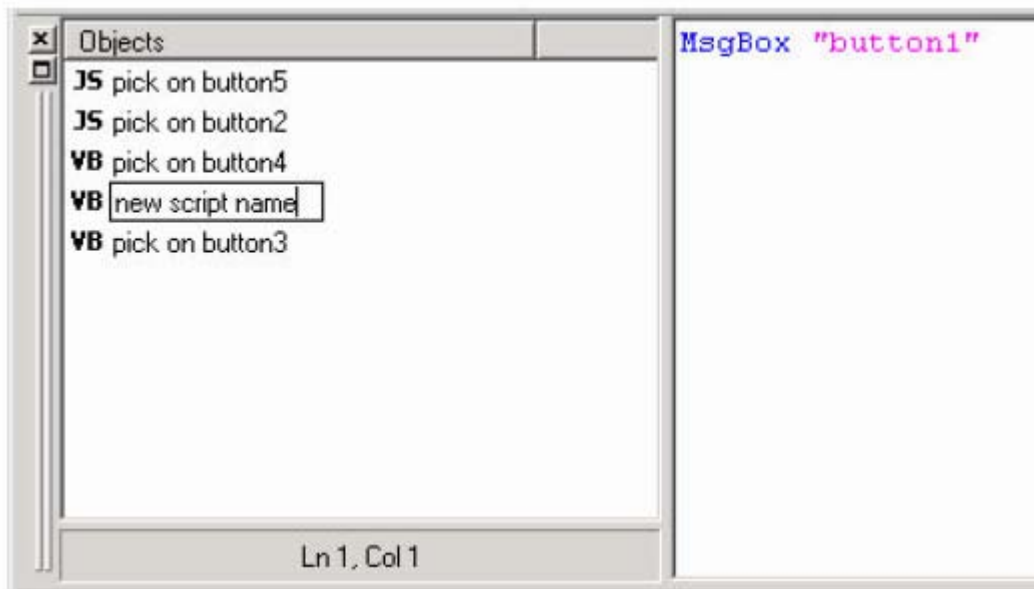


Pokazaće se skript editor, koji izgleda kao na narednoj slici. Editor je podjeljen u lijevi i desni panel. Lijevi panel je skript browser, a desni pokazuje kod za izabrani skript. Svaki skript u lijevom panelu je označen kao VBScript (**VB**) ili Jscript(**JS**), kao što se vidi sa slike.

Skript editor ima dva glavna moda editiranja: **event mode** i **object mode**. Kada je skript editor u event modu, lijevi panel izlistava sve displej događaje koji se mogu skriptovati. Kada je skript editor u object modu, lijevi panel izlistava skriptovane objekte u displeju. Korisnik može preključiti editorski mod klikanjem na zaglavlje lijevog panela. Na narednoj slici je pokazan i događaj (event) i objektni mod.



Skript editor u event modu



Skript editor u objektnom modu

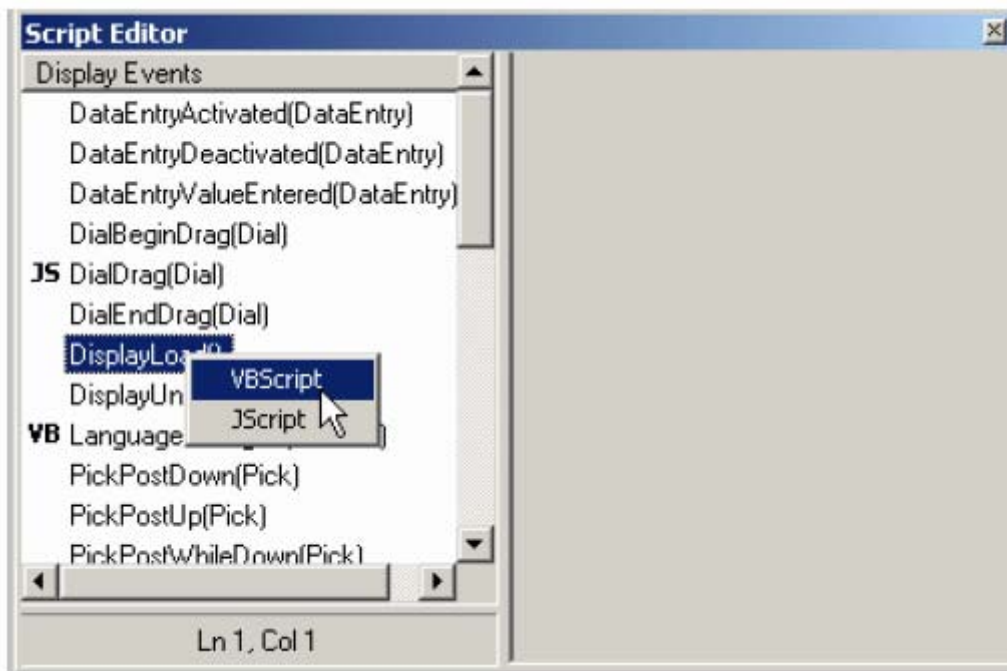
Svaki skript i njegov odgovarajući objekat su sinhronizovani u displeju. Zbog toga, ako izaberemo jedan objekat u displeju koji ima skript pridružen sa njim, tada se skript

automatski naglašava (highlighted) u skript editoru. Inverzno, ako izaberemo skript u skript editoru, tada odgovarajući objekat se takodjer izabire u displeju.

Event mode

Kada je skript editor u event modu, tada lijevi panel izlistava sve displej događaje koji se mogu skriptovati. Za svaki događaj, moguće je napisati skript da ga obradi.

Da bi se udružio skript sa događajem, desno kliknuti na događaj u skript browseru (lijevi panel). Pojaviće se pop-up meni koji je pokazan na narednoj slici. Tada možemo izabrati jezik koji želimo koristiti da se manipliše sa skriptom (VBScript ili Jscript).



Selekcija tipa skripta

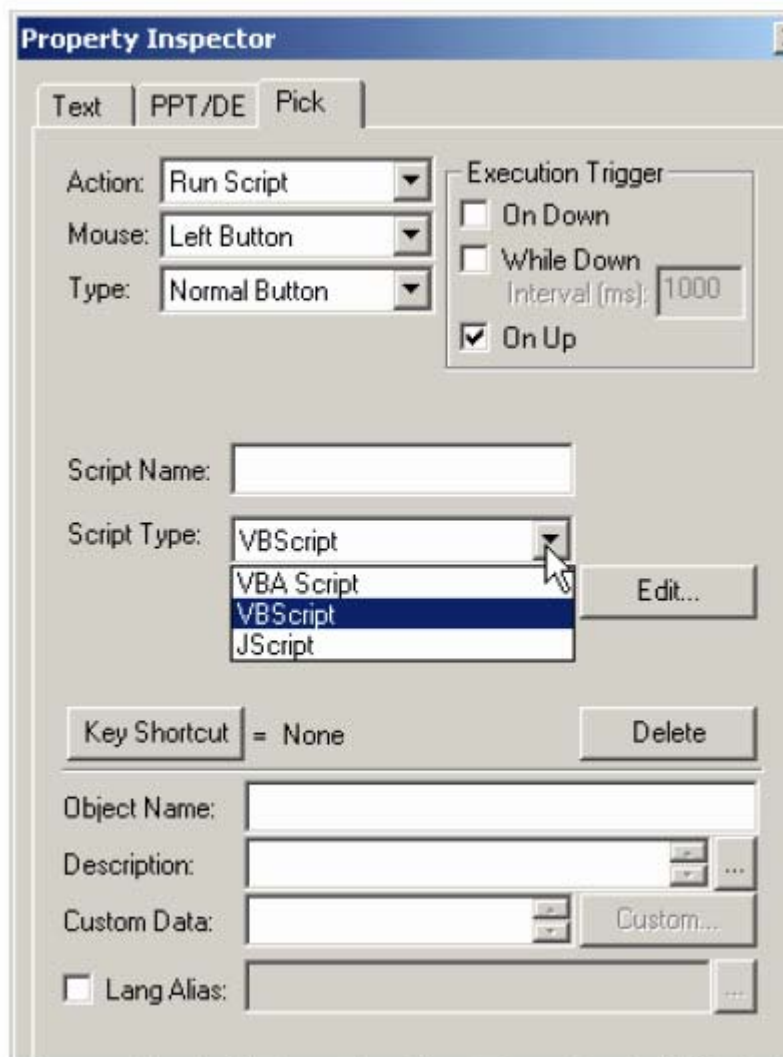
Objektni mod

Kada je skript editor u objektnom modu, možemo editirati skript koji je udružen sa objektom. Moguće je produžiti skript sa objektom putem "pick action" dinamike. Pick akcija je specijalna vrsta dinamike koja se izvršava svaki put kada kliknemo na objekat. Kada je skript editor u objektnom modu, lijevi panel izlistava skriptirane objekte u displeju. **Scripted object** je svaki GraphWorX32 objekat koji ima "run script" pick akciju pridruženu sa njim i ima tip "VBScript" ili "Jscript". Skript editor nikada ne pokazuje objekte sa regularnom VBA skript pick akcijom. VBA skript se može editirati samo sa standardnim VBA Editorom.

Da se kreira novi skript i udruži sa objektom treba dodati "pick action" dinamiku na objekat:

1. U **Pick** tabu Proprety Inspector, izabrati **Run Script** pick akciju i ukucati ime za skript u polje Script Name. Ako izostavimo ime skripte, GraphWorX32 će automatski koristiti ime objekta iz osobine pick objekta.

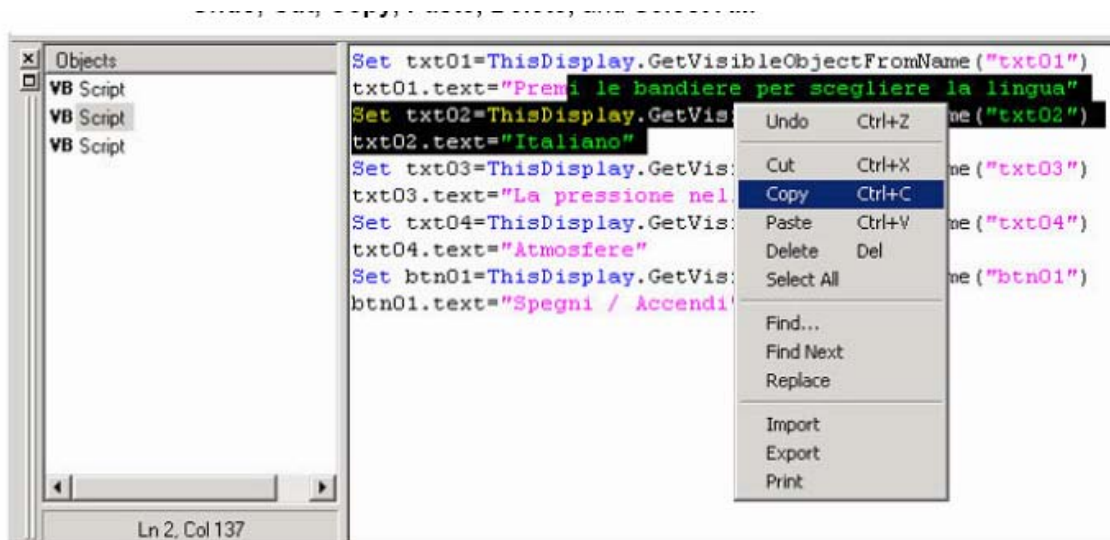
2. Izabrati tip iz Script Type drop-down liste, kako je pokazano na narednoj slici. Samo VBScript i Jscript se mogu editirati sa Script editorom.



Izabiranje tipa skripta za "run script" pick akciju

Bazne tekst operacije

Kada se klikne desnim tasterom na desni panel (skript panel), pojaviće se pop-up meni, kao na narednoj slici. Ovaj meni sadrži bazne tekst operacije: **Undo**, **cut**, **copy**, **paste**, **delete** i **select all**.



Editiranje teksta u skript editoru

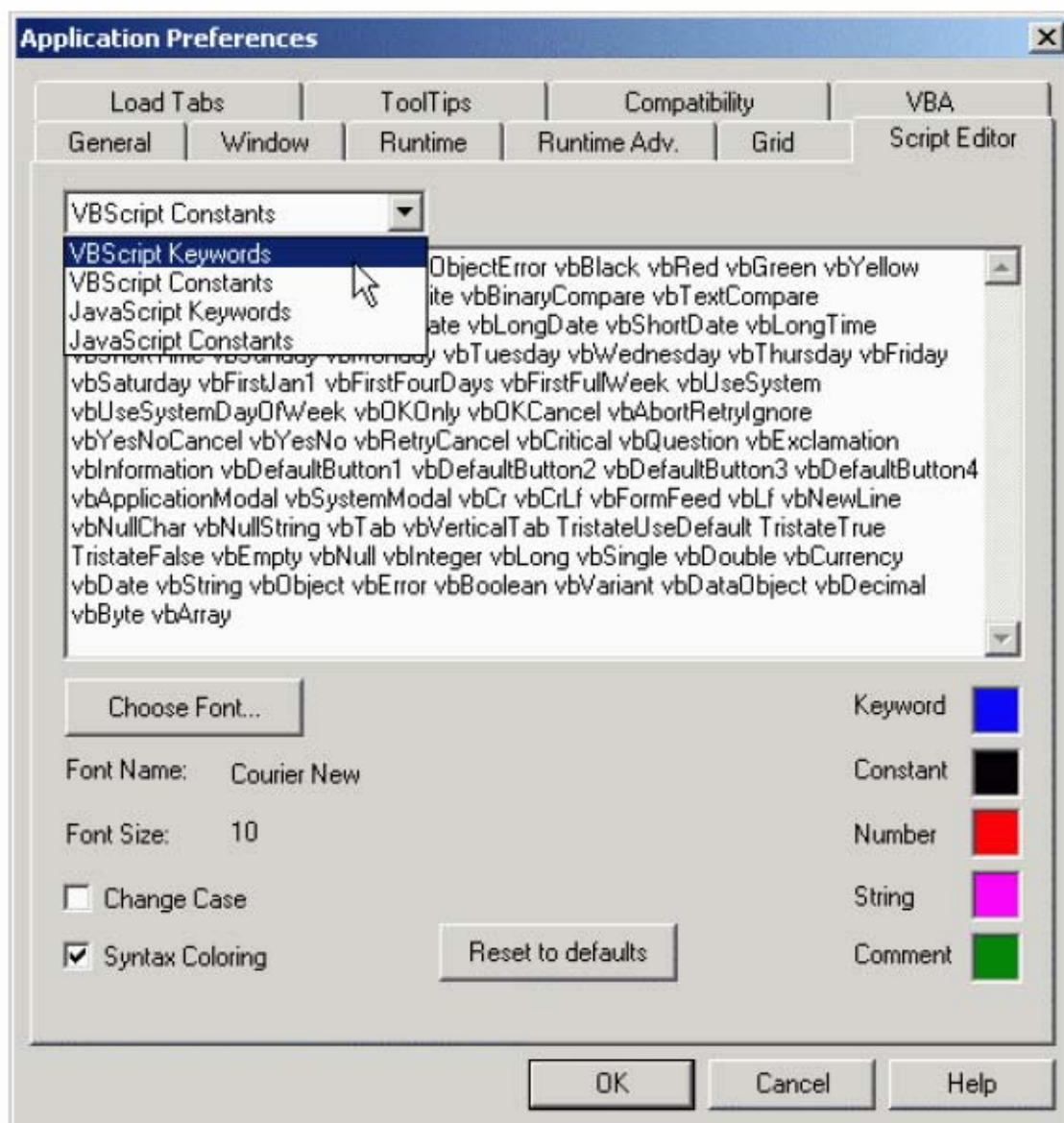
Ostale mogućnosti editiranja su: traženje i zamjena teksta (text search and replace), uvoz i izvoz skripti (importing and exporting scripts), printanje skripti, itd.

Preferencije i setinzi

Skript editor je dizajniran da se može kastomizirati za korisnike. Kastomizacioni setinzi se konfiguriraju u tabu **Script Editor**, u **Application Preferences** dijalog boks, izabirući **Application Preferences** iz **Format** menija.

Korisnik može promijeniti slijedeće setinge skript editora, :

- veličinu , stil i oblik fontova
- ključne riječi i konstante (keywords and constants)VBScript-a
- Jscript ključne riječi i konstante
- omogućenje ili onemogućenje sintaksnog bojenja
- omogućenje ili onemogućenje osobine automatske promjene velikih/malih slova



Aplikacione preference: tab skript editora

Podrška TraceWork32

Koristeći tehnologiju koja je inkorporirana u sve ICONICS proizvode, TraceWorX32 obezbeđuje online dijagnostiku i podešenje aplikacija koje se izvršavaju u GENESIS32 sistemu. TraceWorX32 je dizajniran za integratore sistema, OEM i kupce koji žele da imaju alate da mogu provesti svoj vlastiti troubleshooting i dijagnostiku.

TraceWorX32 prati runtime aktivnosti za svaku GENESIS32 aplikaciju i logira runtime podatke u log fajl na bazi korisnički konfigurisanih trejs nivoa (nivoa tragova). Log fajl obezbeđuje cjelovito, bojom kodirano detaljno izvještavanje svih aktivnosti za aplikaciju, uključujući vrijeme, datum, nivo ozbiljnosti (severity) i opis događaja ili problema.

TraceWorX32 također posjeduje nekoliko opcija za izvještavanje i tehničku podršku. Ako korisnik iskusi probleme sa bilo kojom aplikacijom, opcija sa korištenjem log fajla,

koji se može komprimirati i uključiti u email, je idealna za praćenje i arhiviranje podataka i slanje detaljnih izvještaja za tehničku podršku. Razvojni inženjeri mogu koristiti ove izvještaje da indentificiraju izvore problema.

TraceWorX32 se može aktivirati sa slijedećim registarskim ključevima za svaku GraphWorX32 komponentu. **ReleaseTraceLevel** se može setovati izmedju 0 (puna informacija o tragovima) i 1000 (praćenje i registracija tragova isključena). Kada je aktivan, generisaće se log fajl bilo u tekućem direktoriju komponenti, ili u subfolderu od "Documents and Settings". Treba tragati za *.log xml dokumentom.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\GWX32]
```

```
"ReleaseTraceLevel"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\GWXview32]
```

```
"ReleaseTraceLevel"=dword:00000000
```

VBA wizardi

Visual Basic za aplikacije (VBA) wizardi su GraphWorX32 objekti sa Visual Basic kodom iza sebe. Kod se izvršava ili u dizajn modu da pomogne da se konfigurira GraphWorX32 objekat ili u runtime modu da se izvrši specifičan task.

Normalno, VBA kod se pohranjuje u tekući dokument, pohranjen iz VBA editora kada se pohrani displej, i ponovno loadovan u VBA Editor kada se prikaz otvori u GraphWorX32. Medjutim, ako se slijede pravila koja su opisana u nastavku, VBA kod se može "vezati" sa GraphWorX32 objektom. Kada se takav objekat "uljepi" (pasted) ili ispusti (drop) u drugu instancu od GraphWorX32, u biblioteku simbola, ili se odbaci (scrap), tada ovaj kod odlazi sa tom akcijom.

VBA wizard se može izvršavati da realizuje specifični task bilo u dizajn ili runtime modu.

Dizajn mod

Korisnik može lansirati makro VBA wizarada u dizajn modu u GraphWorX32 sa dvostrukim klikom na VBA wizard.

Po defaultu, kada je jedan objekat dvaputa kliknut u dizajn modu lansira se Property Inspector. Medjutim, ukoliko se doda specijalna ključna riječ (keyword) u prvoj liniji polja **Custom Data** , može se startati i makro.

Format ključne riječi za makro koja se zove "MacroName" je:

```
OnDoubleClick=<GwxMacroName_Main.MacroName>, Parameters=<>
```

Mora postojati makro "MacroName" u modulu "GwxMacroName_Main" u VBA da se uspješno izvrši makro. Korisniku je dozvoljeno da postavi bilo koji string izmedju ugaonih zagrada od "Parameters=<>" sekcije. Svi podatci koji će se poslati makrou će

se staviti između uglatih zagrada. Time će ovi podaci biti na raspolaganju kada se makro izvršava.

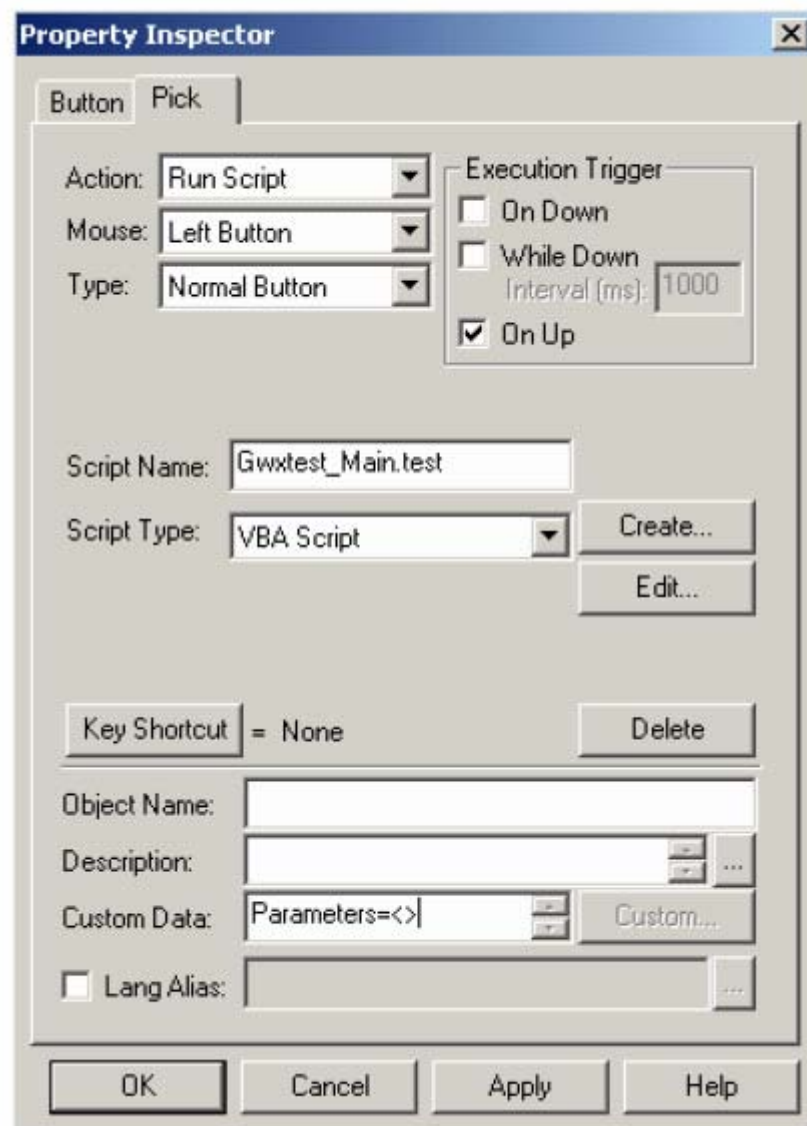
Opaska : Ime makroa ne može sadržavati prazan prostor (space).

Runtime mode

Korisnik može lansirati VBA wizard makro u runtime modu klikanjem na taster GraphWorX32 ili sa pick akcijom koja je konfigurisana da izvrši makro. Polje **Action** u **Pick** tabu na Property Inspectoru mora biti konfigurisano da **Run Script**. Polje **Script Name** mora sadržavati ime makroa u slijedećem formatu:

GwxMacroName_Main.MacroName

Izabrati VBA Skript iz drop-down liste pod **Script Type**. Kliknuti na Create taster da se izvrši VBA Script wizard.



VBA wizard u pick akciji

Polje Custom Data može sadržavati bilo koji string koji drži kastom podatke željenim , umetnutim izmedju uglatih zagrada.

Pravila za VBA wizard

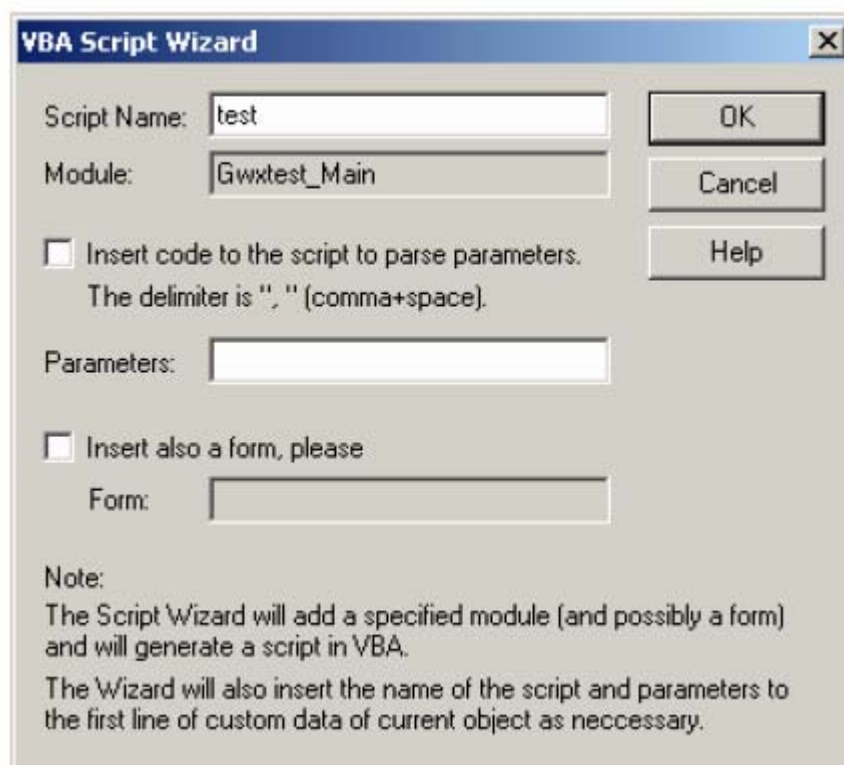
GraphWorX32 vodi računa o kodiranju iza VBA wizarda. Zbog specijalne konvencije o imenima svih modula koji pripadaju specifičnom VBA wizardu, kod u ovim modulima se može korektno pomjerati sa simbolom.

Ako je makro ime VBA wizarda "MacroName", tada nazivi svih modula sa kodom moraju početi sa stringom "GwxMacroName_". Ova tehnika dozvoljava da se više modula koda i formi koristi za jedan objekat VBA wizarda, i to olakšava pomjerenje cjelokupnog koda se objektom, kada je to neophodno.

Alat za kreiranje VBA wizarda

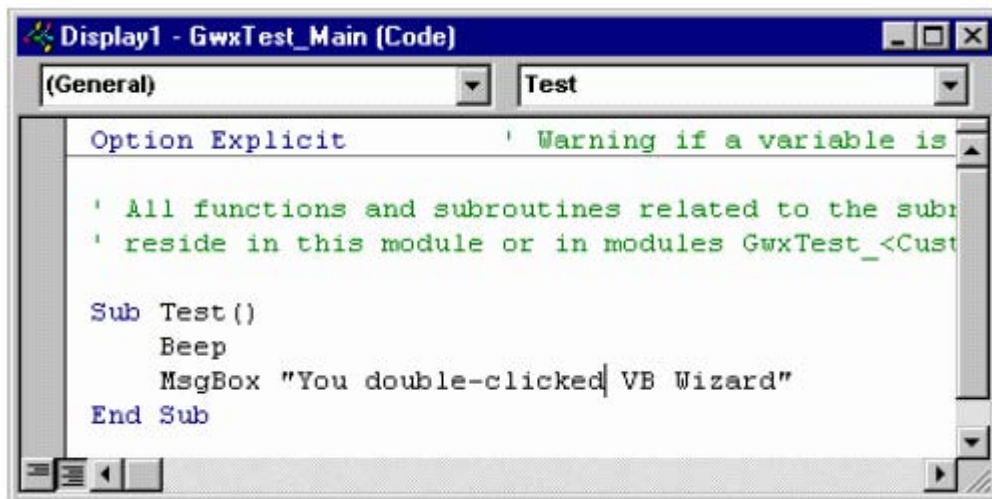
Pošto kreiranje VBA wizard objekata može biti mukotrpan zadatak, GraphWorX32 nudi VBA Script wizard koji konvertuje objekte u VBA wizard i generira VBA template kod. Nakon toga kod se može lako poboljšati i modificirati. Slijedeći primjer demonstrira korištenje ovog alata:

1. Otvoriti GraphWorX32 displej
2. Kreirati nekoliko elipsi i grupirati ih u simbol
3. Kliknuti desnim tasterom na simbol i izabrati **Create VBA Script** iz pop-up menija. Ovo će otvoriti VBA Script Wizard dijalog boks, koji je pokazan na narednoj slici:
4. Unjeti ime makroa (na primejr "Test")u polje **Script Name**.



Dijalog VBA Script wizarda

5. Kliknuti OK. Otvoriće se Visual Basic Editor, kako je pokazano na narednoj slici, i kursor će biti smješten u tijelo od "Test" subrutine u modulu "GwxTest_Main". Sada možemo unjeti kod koji će se izvršiti kada kliknemo dva puta na simbol u dizajn modu.



```
Option Explicit      ' Warning if a variable is

' All functions and subroutines related to the subr
' reside in this module or in modules GwxTest_<Cust

Sub Test ()
    Beep
    MsgBox "You double-clicked VB Wizard"
End Sub
```

Test subrutina (generisana od strane wizarada)

6. Vratimo se u GraphWorX32 i dvaput kliknimo na simbol. Čut će se beep zvuk, i pojaviće se poruka.

Makro ili ime skripte

U polju **Script Name** dijalog boks **VBA Script wizarada**, treba ukucati ime makroa. Ime makroa treba početi sa karakterom i treba sadržavati samo alfanumeričke karaktere.

Modul

Polje **Module** u dijalog boksu **VBA Script wizarada** je uvijek posivljeno (grayed), pošto se ime modula automatski generiše na bazi imena skripte.

Ako ček boks **Insert code to the script to parse parameters** je čekiran, generiše se nešto dodatnog koda u tijelu makro subrutine. To pomaže u dobijanju i pohranjivanju parametara iz VBA wizard objekta. Ovaj kod koristi "GwxTools" module da konvertuje parametre u string, koji se naziva "Strpar", koji je lokalan za makro subrutinu.

Parametri

Unjeti u polje **Paramters** bilo koji proizvoljni string u okviru dijalog boksa za **VBA Script Wizard**. Dobićemo ovaj string kada se makro bude izvršavao. Ovo opciono polje je dizajnirano da dozvoli kastomizirane podatke specifične za instancu VBA wizarada. Različite instance istih objekata mogu sadržavati različite podatke. Ova osobina služi da korisnik pojednostavi svoj kod , omogućujući mu da prenese vrijednosti u makro.

Molim, unesite takodjer i formular

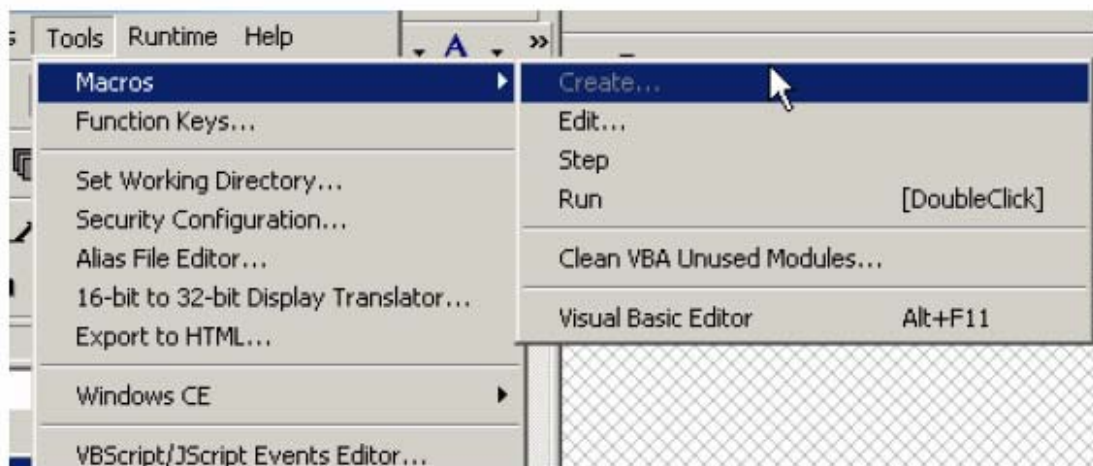
Treba čekirati ček boks **Insert also a form, please** u dijalog boku **VBA Script wizarda** ako je potrebno lansirati VBA formu iz makroa. Korisniku je dozvoljeno da kreira proizvoljan broj formulara za VBA wizard, uz ograničenje da mora poštovati konvenciju o imenima. (ako se ona ne poštuje, VBA kod nije prenesen sa objektom kada je to neophodno).

Formular

Polje **Form** u VBA Script wizardu je uvijek posivljeno (grayed out) i generiše se automatski na bazi imena skripte.

Dizajn mod VBA wizarda

Submeni **Macros** od **Tools** menija sadrži nekoliko drugih komandi za detalje za VBA wizard u dizajn modu, kao što je pokazano na narednoj slici:



VBA wizard meni

Kreiranje makroa

Izabirući **Macros> Create** iz **Tools** menija otvara **VBA Script Wizard** dijalog boks. Možemo takodjer kliknuti desnim tasterom na objekat i zatim izabrati **Create VBA Script** iz pop-up menija. Obadva menija su omogućena ako se izabere objekat u displeju koji nije VBA wizard.

Editiranje makroa

Izabirući **Macros>Edit** iz **Tools** menija , otvara se VBA editor i postavlja kurzor u tijelo makroa. Možemo takodjer kliknuti desnim tasterom na objekat i izabrati **Edit VBA Script** iz pop-up menija.

Step makro

Izabirući **Macros>Step** iz **Tools** menija izvršava makro u step (debug) modu. Otvara VBA editor i zaustavlja se na prvoj liniji programskog koda makroa. Omogućen je ako VBAS wizard objekat je izabran u displeju.

Izvršenje makroa (run macro)

Izabirući **Macros>Run** iz **Tools** menija , izvršava se makro. Možemo to takodje izvršiti kliknuvši dva puta na VBA Wizard. Omogućeno je ako VBA Wizard objekat je izabran na displeju.

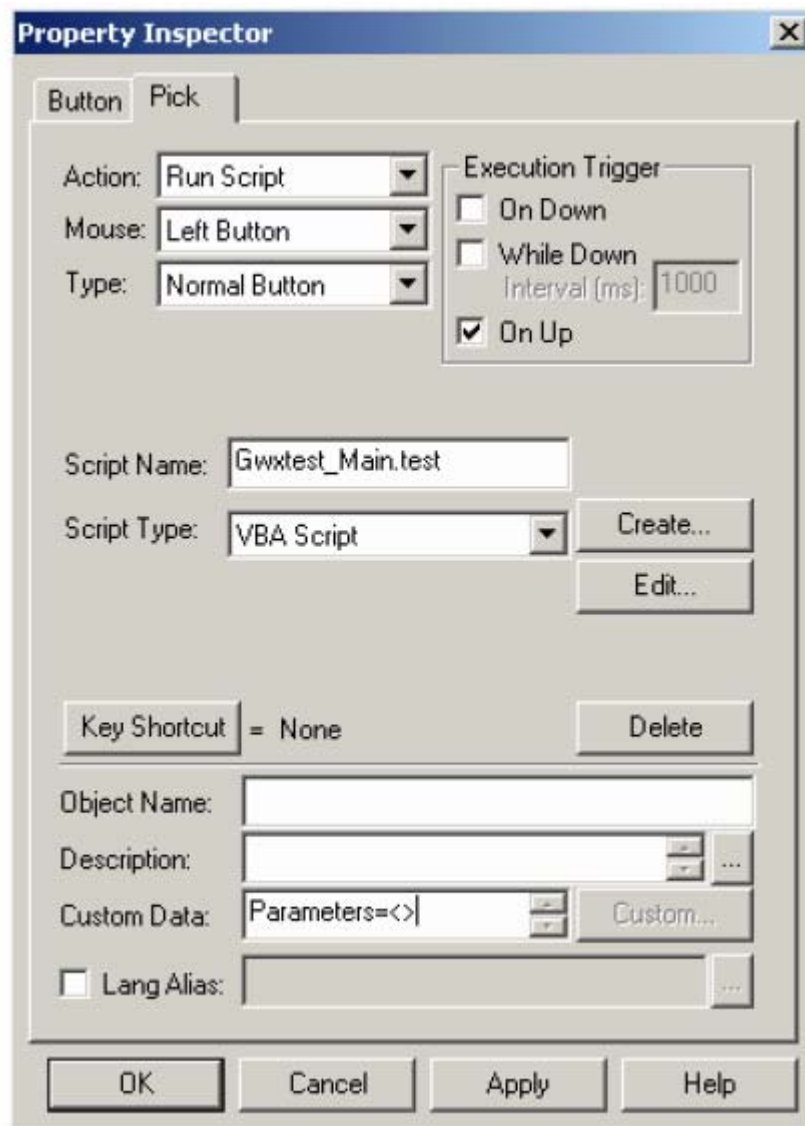
VBA wizard za runtime i dizajn mod

Kada VBA wizard je pobrisan (deleted) ili pomjeren van tekućeg displeja, VBA kod se ne briše automatski. Medjutim, izabirući **Macros>Clear VBA Unused Modules** iz **Tools** menija, otklanja sve module iz VBA Editora, koji startaju sa "Gwx" stringom i nisu referencirani iz tekućeg prikazanog VBA wizarda (bilo da su bazirani u dizajn ili runtime-u).

Runtime mode VBA wizard

Ako želimo da kreiramo novi runtime bazirani VBA wizard, potrebno je kreirati taster ili pick akciju, izabrati **Run Script** u **Action** polju, kao što je pokazano na narednoj slici. Izabrati **VBA Script** iz drop-down liste pod **Script Type**. Kliknuti na **Create** taster da se izvrši VBA Script wizard.

Opaska: Nema potrebe da se koristi Edit komanda da bi se editirao makro. Umjesto, možemo otvoriti VBA Editor, naći modul koji hoćemo editirati i editirati ga direktno.



VBA wizard u pick akciji

GraphWorX32 Active X kontrola

GraphWorX32 ActiveX ("GWXview32.ocx") je ActiveX kontrola koja je sposobna da izvršava GraphWorX32 displeje. Prednost ActiveX kontrole kao što je GraphWorX32 je da može biti uronjena (embedded) u bilo koji kontrolni kontejner (napr. Visual Basic Forms, Internet Explorer HTML stranice, GraphWorX32 displeje, itd.)

GraphWorX32 je u suštini runtime komponenta (sa nekoliko minimalnih konfiguracionih mogućnosti); dizajn samo za runtime dozvoljava da se ActiveX kompaktira u odnosu na korištenje memorije. GraphWorX32 ActiveX izvršava prikaze kreirane sa GraphWorX32. GraphWorX32 ima sve runtime mogućnosti od GraphWorX32.exe , izuzev sposobnosti da izvršava VBA skripte.

Unošenje GraphWorX32 Active X

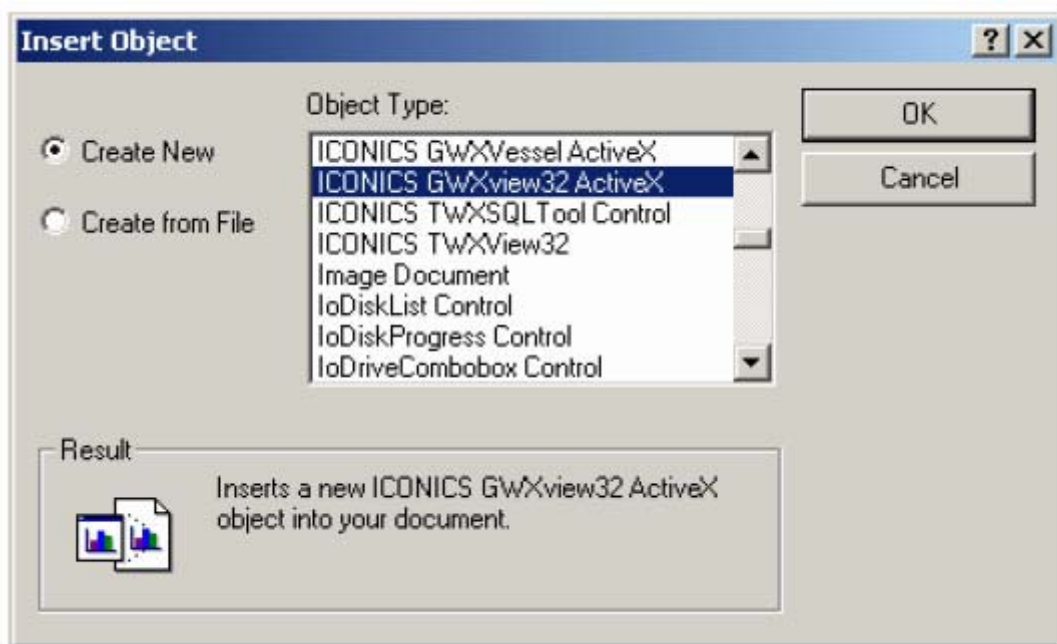
Tehnike za unošenje ActiveX kontrole mogu malo varirati između različitih kontrolnih kontejnera, mada su osnovni principi isti. U nastavku ćemo pokazati kako unjeti GraphWorX32 ActiveX u GraphWorX32 displej.

1. Iz **Edit** menija, izabrati **Insert New Object**, ili kliknuti na **Insert ActiveX Control/OLE Object** taster na **ActiveX** alatnoj letvici, ako što je pokazano na narednoj slici:



Insert ActiveX Control/OLE Object taster

2. Ovo će otvoriti **Insert Object** dijalog boks, kao na narednoj slici:



Insert object dijalog boks

2. Izabrati detalj **ICONICS GWX View32 ActiveX** iz liste raspoloživih kontrola, i zatim kliknuti **OK**. GraphWorX32 ActiveX kontrola će se pojaviti u centru displeja. Alternativno, možemo koristiti **GWXView32 ActiveX** taster na **ActiveX** toolbaru da direktno unesemo GraphWorX32 ActiveX.

GraphWorX32 ActiveX alatna letvica



Kastomizacija ActiveX alatne letvice

Bilo koji ActiveX koji je unesen u dispelj putem OLE tastera sa ActiveX toolbara se automatski dodaje na alatnu letvicu kao novi taster. Korisnik može ukloniti ActiveX tastere sa alatne letvice kliknuvši desnim tasterom na alatnu letvicu. ovo će otvoriti dijalog boks **Customize Toolbar**. Izabrati ActiveX iz liste na desnoj strani, i zatim kliknuti na **Remove**. Alternativno, taster se može ukloniti sa alatne letvice tako što se klikne na taj taster i dok Shift taster ostane pritisnut, izvući taster sa letvice.

OLE ActiveX

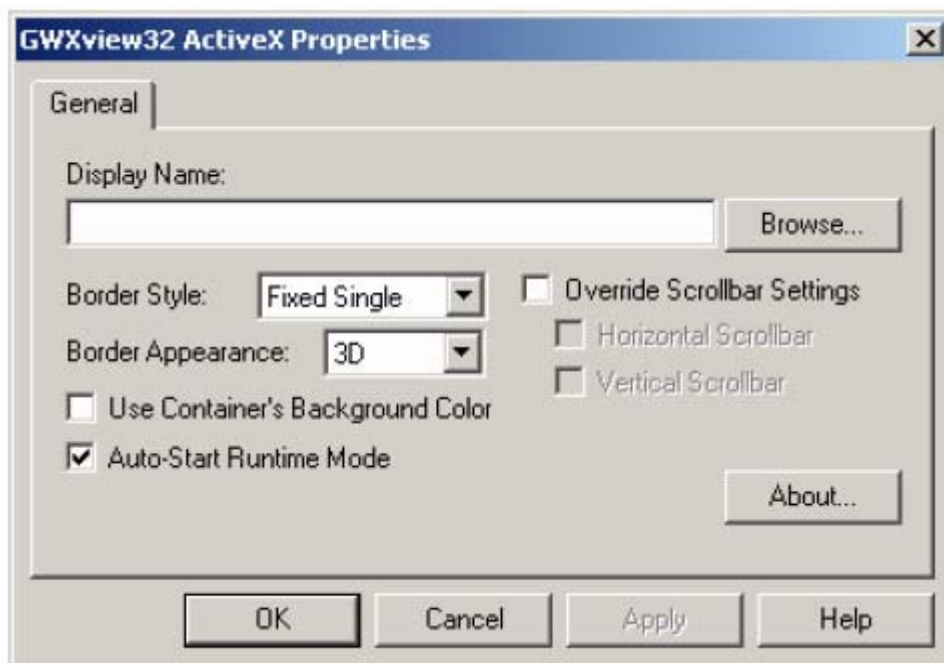


Klikanjem na **OLE** taster na **ActiveX** toolbaru, unosi se ActiveX Control/OLE objekat.

Graphics ActiveX



Klikanjem na **GWXView32 ActiveX** taster na **ActiveX** toolbaru, otvara se Graphics ActiveX.



Dijalog boks osobina GWXView32 ActiveX

Na gornjem dijalog boksu korisnik može unjeti GraphWorX32 displej unutar njega samoga.

Može se pohraniti ovaj umetnuti displej sa ili bez VBA. Po defaultu bit će pohranjen sa VBA. Međutim, ako je displej umetnut (embedded), treba biti pohranjen bez VBA pošto će se inače pojaviti slijedeća poruka greške, ako bi pokušali da pohranimo sa VBA.



Trend ActiveX



Klikanjem na **TWXView32** taster na **ActiveX** toolbaru otvara sa TrendWorX32 Viewer ActiveX

Alarm ActiveX

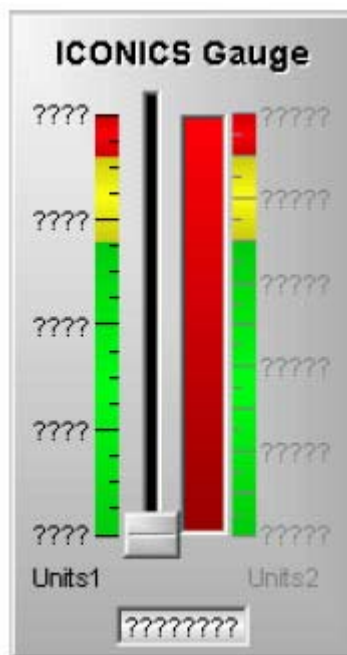


Klikanjem na **AWXView32** taster na **ActiveX** toolbaru otvara se AlarmWorX32 kontejner.

Gauge ActiveX



Klikanjem na **GWXGauge ActiveX** taster na **ActiveX** toolbaru otvara se Gauge ActiveX koji je pokazan na slijedećoj slici :



Switch ActiveX



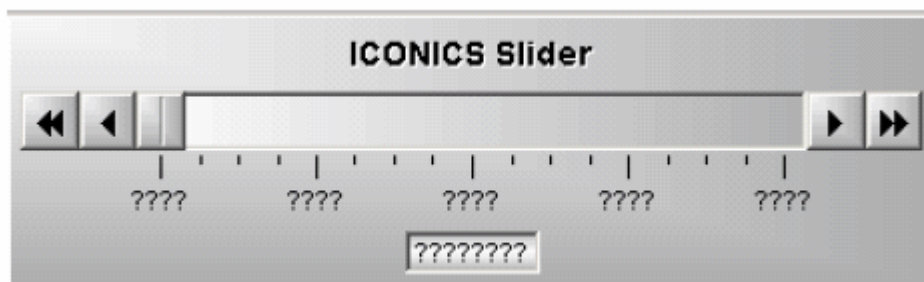
Klikanjem na **GWXSwitch ActiveX** taster na **ActiveX** toolbaru otvara se Switch ActiveX pokazan na narednoj slici:



Slider ActiveX



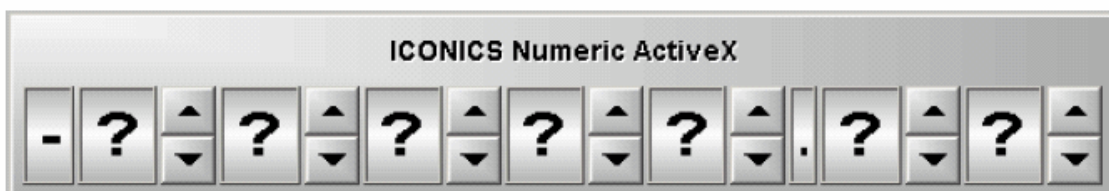
Klikanjem na **GWXSlider ActiveX** taster na **ActiveX** toolbaru



Numerički ActiveX



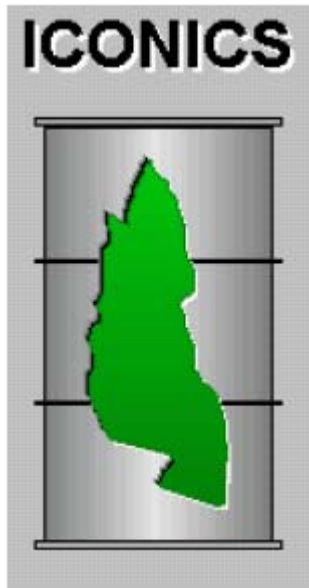
Klikanjem na **GWXNumeric ActiveX** taster na **ActiveX** toolbaru otvara se Numeric ActiveX pokazan na narednoj slici:



ActiveX za posude



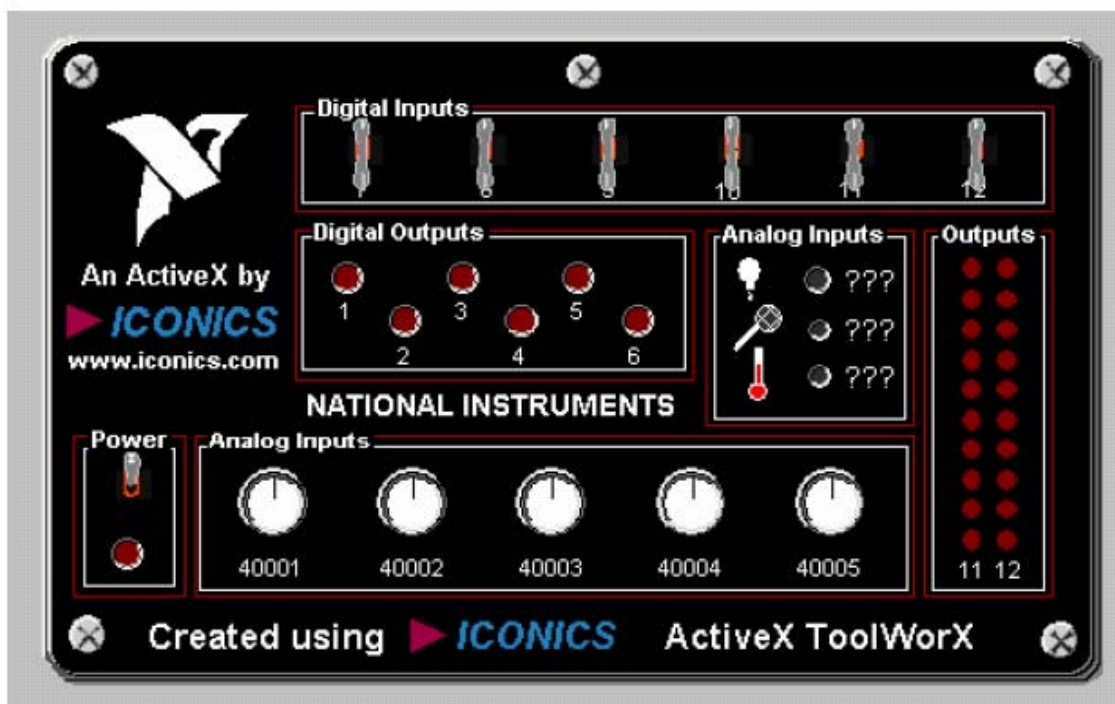
Klikanjem na **GWXVessel ActiveX** taster na **ActiveX** toolbaru otvara se ActiveX za posude koji je pokazan na narednoj slici:



National Instruments ActiveX



Klikanjem na **National Instruments ActiveX** Control taster na ActiveX toolbaru otvara se ActiveX kontrol kompanije National Instruments (NI), koji je pokazan na narednoj slici:

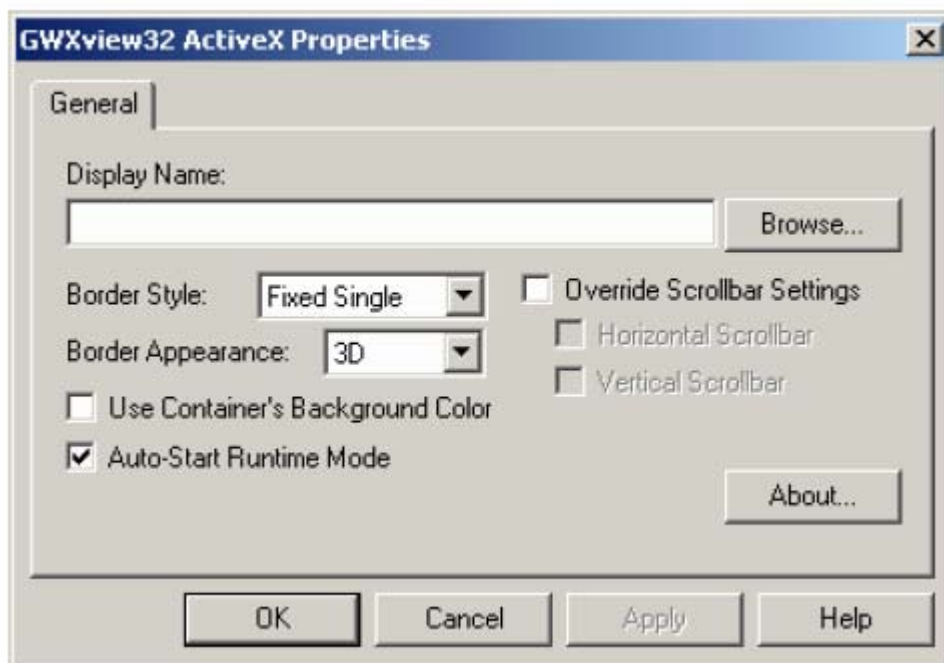


Konfigurisanje GraphWorX32 ActiveX

Dijalog boks osobina GXWView32 ActiveX je pokazan na narednoj slici , i omogućava nam da promjenimo neke attribute od GraphWorX32 kontrola.

Da bi konfigurisali GraphWorX32 ActiveX treba:

1. Dvapat kliknuti na GraphWorX32 ActiveX da se prikaže dijalog boks osobina kontrola, kao što je pokazano na narednoj slici.
2. Ispuniti parametre opisane u narednoj tabeli, i zatim kliknuti OK.



GraphWorX32 ActiveX parametri

Parametar	Opis
Display name	Specificira ime GraphWorX32 displej fajla koji će se loadovati u ovu kontrolu. Kliknuti na Browse taster da se traži.
Border style	Specificira pojavljivanje ivica prozora. "No Border" (bez ivica) ili "Fixed Single" (fiksna jednostruka),
Border Appearance	----
Use Container's background color	----
Auto-start runtime mode	Ako je čekiran , kontrola će automatski ući u runtime mode kada kontrolni kontejner ulazi u runtime mode. Ova osobina će raditi samo ako kontejner podržava AmbientUserMode osobinu. GraphWorX32 i Visual basic Forms podržavaju ovu osobinu

Override scrollbar settings	---
horizontal scrollbar	----
vertical scrollbar	---

OLE Automation

OLE Automation se koristi da pristupi osobinama i metodama objekata u GraphWorX32. Osobine (properties) se koriste da referenciraju attribute ako objekat (napr. GraphWorX32 displej) ima *BackgroundColor* osobinu. Metode se koriste da učine da objekat izvršava akciju (naprimjer, displej ima *FileOpen* metod koji loaduje novi displej) Osobine i metodi se mogu pozivati iz Visual Basic za aplikacije (VBA) , VBScript, ili JavaScript, ili iz C++ programa.

Postoji niz tipova objekata u GraphWorX32, i svaki ima svoje metode i osobine. U narednom poglavlju biće detaljno opisane metode i osobine za tipove objekata u GraphWorX32.

Registarcija kod GenRegistrar-a u dizajn modu

GraphWorX32 se takodjer registruje kod GenRegistrar-a u dizajn modu. U dizajn modu, GraphWorX32 registrira dispeč poenter na displej sa ključnom riječi: "GWX32_DESIGN_MODE".

Kada ulazi u runtime, on ponovo registruje dispeč poenter na <current-display-title> (tekući prikazani naslov), i drži ovaj ključ sve dok je u runtime-u, bez obzira koji su displeji loadovani. Ovo obezbjeđuje lagan način da se prepoznaju pomoćne instance od GraphWorX32 u runtime-u (na bazi captiona sa kojim oni startaju).

Osobina registracije u dizajn vremenu je po defaultu isključena (off), i može se uključiti koristeći registarski ključ DesignTimeRegistration pod GWX32/Compatibility osobinama, kao što je pokazano u narednom tekstu:

```
0    .. feature disabled
1    .. feature enabled
```



Omogućivanje registracije dizajn moda u GenRegistrar-u.

Kada napušta runtime mode, GraphWorX32 se registruje ponovo sa ključem "GWX32_DESIGN_MODE".

Slijedeći primjer pokazuje kako instanca GraphWorX32 u dizajn modu se može dobiti iz Visual Basic aplikacije:

```

Sub AutomateGwx()
    ' Note: Add ICONICS AutoGenRegistrar to Project |
References
    ' Note: Add GWX32 to Project | References

    ' Create an Automation GenRegistrar Wrapper
Dim genreg As New AutoGenRegistrar
If genreg Is Not Nothing Then Exit Sub ' failure

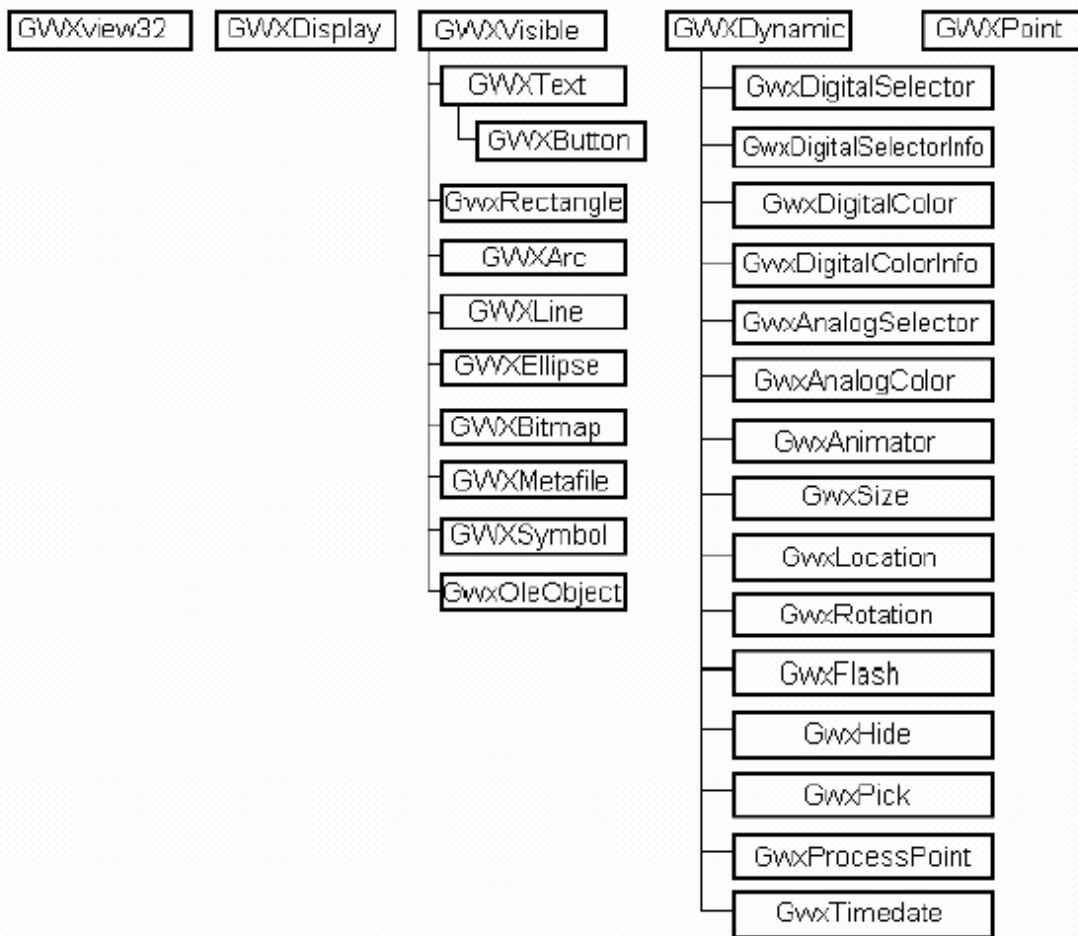
    ' Declare GWX reference and get it from GenRegistrar
    ' Note that we ask for GWX instance in design mode
Dim gwx As Gwx32.GwxDisplay
Call genreg.GetDispatch("GWX32_DESIGN_MODE", "GWX32",
"", gwx)
If gwx Is Nothing Then Exit Sub ' failure

    ' Automate GWX instance
gwx.ShowWindow
gwx.BringWindowToTop
Randomize
gwx.BackgroundColor = RGB(Rnd * 255, Rnd * 255, Rnd *
255)
End Sub

```

Hijerarhija objekata

Slijedeći dijagram pokazuje hijerarhiju objekata eksponiranih od strane GraphWorX32 putem OLE Automation



GWXview32 je tip podatka ICONICS-ovog GraphWorX32 ActiveX kontrole. Za razvojne inženjere koji koriste ICONICS-ov ActiveX ToolWorX , **GWXView32** je tip podatka bazne ActiveX kontrole iz koje se izvode nove kontrole (nove kontrole mogu naslijediti osobine i metode od GWXView32). Svaki GWXView32 objekat sadrži jednostruki **GwxDisplay** objekat.

GwxDisplay je tip objekta GraphWorX32 displeja. GraphWorX32 displej sadrži kolekciju **GwxVisible** izvedenih objekata, **GwxDynamic** izvedenih objekata, i **GwxPoint** objekata. **GwxDisplay** objekat pokazuje metode za kreiranje i dobijanje postojećih instanci ovih vidljivih i dinamičkih objekata.

GwxVisible je bazni tip objekta za objekte u GraphWorX32 displeju koji može biti vidjen. Svi vidljivi GraphWorX32 objekti (napr. **GwxRectangle**, **GwxEllipse**, **GwxText** , itd.) su izvedeni iz **GwxVisible** i kao posljedicu toga naslijedjuju sve osobine i metode **GwxVisible**.

GwxDynamic je bazni tip objekta za objekte u GraphWorX32 displeju koji izvršava dinamičku transformaciju na pridruženom vidljivom objektu. Naprimjer, **GwxSize** objekat pridružen sa **GwxRectangle** objektom bi promjenio veličinu pravougaonika na bazi OPC vrijednosti pridružene sa **GwxSize** objektom.Svi dinamički GraphWorX32 objekti (tj. **GwxSize**, **GwxLocation**, **GwxRotation**, itd.)

su izvedeni iz **GwxDynamic** i posljedično, naslijeđuju sve osobine i metode od **GwxDynamic**. Dinamički objekti nisu vidljivi sami po sebi, oni postoje da bi djelovali po osobinama vidljivosti pridruženog objekta izvedenog iz **GwxVisible**. **GwxVisible** objekat može imati mnoge **GwxDynamic** objekte pridružene sa njim , ali ima samo jedan **GwxDynamic** objekat koji je udružen sa **GwxVisible** objektom.

GwxPoint je bazni tip objekta za objekte u GraphWorX32 displeju koji referenciraju OPC podatke, izraze, lokalne varijable ili konstantne vrijednosti. Svaki **GwxDynamic** objekat može imati jedan ili više **GwxPoint** objekata pridruženih sa njim. **GwxPoint** objekti se kreiraju i uništavaju pomoću **GwxDynamic** objekata , prema potrebi.

Automation programiranje u C++ i Visual Basicu

Razumjevanje naslijeđivanja (inheritance)

GraphWorX32 OLE Automation hijerarhija objekata koristi u velikoj mjeri naslijeđivanje. Ovo pokreće neka programska pitanja i u C++ i u Visual Basicu. Da se razumiju ova programska pitanja, moramo prvo razumjeti kako se pristupa metodama i osobinama putem OLE Automation interfejsa.

Može se pristupiti OLE Automation metodi ili osobini putem imena metode/osobine ili putem ID broja pridruženog sa konkretnim metodom ili osobinom. Kada izvedeni objekti koriste naslijeđjene metode i osobine od baznog objekta, ID brojevi za naslijeđjene metode ili osobine u izvedenom objektu moraju biti promjenjeni tako da veća riječ ID broja se ostavlja na nivo naslijeđivanja izvedenog objekta ,od baznog objekta u kojem je taj metod ili osobina definirana. Naprimjer, **GwxVisible** definira **FillColor** osobinu da ima ID broj **0x3**.

Gwxrectangle naslijeđjuje **FillColor** osobinu da ima ID broj **0x10003** (pošto **GwxRectangle** je izveden iz **GwxVisible**). Slično, **GwxButton** naslijeđjuje **FillColor** osobinu tako da ima ID broj **0x20003** (pošto **GwxButton** je izveden iz **Gwxtext** koji je izveden iz **GwxVisible**).

Priroda ID brojanja za naslijeđivanje neće prouzrokovati nikakve probleme kada se pristupa objektima putem njihovih aktuelnih tipova objekata. Programerski problemi će se pojaviti kada želimo da vratimo (downcast) izvedeni tip objekta natrag na bazni tip objekta. Na primjer, **GwxDisplay** metod **GetVisibleObjectFromName** će vratiti **GwxVisible** izvedeni objekat, ali neće biti načina da znamo koji je stvarni tip objekta unaprijed u vremenu, tako da programer bi želio da doznači povraćenu vrijednost na **GwxVisible** objekat (radije nego specifični tip objekta kao napr. **GwxRectangle** ili **GwxEllipse**). Ali, ako nakon toga pokušamo da pristupimo metodi ili osobini definiranoj za **GwxVisible** od vraćenog objekta, to neće raditi , pošto ID brojevi će biti pogrešni (ako povraćeni objekat je bio ustvari **GwxRectangle** tada ID za **FillColor** bi bio **0x10003**, ali pošto je tom objektu bilo pristupljeno preko baznog tipa objekta **GwxVisible**, pokušaće da koristi ID broj **0x3**).

U Visual Basicu, rješenje za ovo je jednostavno. Ako ne znamo aktuelni tip objekta i želimo da mu pristupimo generički, treba koristiti tip objekta **Object**. Ovo će prouzrokovati da VB pristupi metodi ili osobini objekta koristeći ime a ne ID broj, i time izbjegne probleme ID brojanja.

U C++ ovaj problem može zahtjevati više rada da se prevaziđe. Tipično, kada koristimo OLE Automation objekte u C++ , mi ćemo dozvoliti Developer Studiju da generiše neku vrstu wrapper koda (koda omotača) za automation interfejs. Naprimjer, možemo koristiti Class Wizard da generiramo **COLEDispatchDriver** izveden iz wrapper klase. Ili, možemo koristiti **#import** direktivu da generiramo interfejsni wrapper kod (**TLI** i **THL** fajlove).Medjutim, obadvije ove tehnike pristupaju metodama i osobinama putem hard kodiranih ID brojeva. Zbog toga, kada koristimo takve wrapper klase nećemo biti u stanju da korektno vratimo (downcast) tipove objekata. Najbolji način da se ovo riješi je da modificiramo generisane wrapper klase za GwxVisible i GwxDynamic da pristupimo metodama i osobinama preko imena radije nego preko hard-kodiranog ID,

Dogadjaji (events)

Ovo poglavlje opisuje dogadjaje koji su izloženi GraphWorX32 integriranom skriptingu Visual Basica za aplikacije. Ovi dogadjaji su takodjer izloženi i iz GWXView32 ActiveX kontrole. (za razvojne inženjere koji koriste ActiveX ToolWorX, postoje funkcije virtuelnih članova koje korepondiraju sa dogadjajima koji su opisani u nastavku).

Event DataEntryActivated (DataEntry As object)

Ovaj dogadjaj se aktivira kada korisnik aktivira objekat za unošenje podatka (tj. korisnik klikne na objekat za unos podatka da unese novu vrijednost). Parametar **DataEntry** je **GwxprocessPoint** objekat koji je prouzrokovao dogadjaj.

Event DataEntryDeactivated (DataEntry As object)

Ovaj dogadjaj se aktivira kada korisnik deaktivira objekat za unos podatka (tj. korisnik klikne van objekta za unos podatka ili koristi escape ključ da poništi editiranje, ili nakon što korisnik unese vrijednost). Parametar **DataEntry** je **GwxprocessPoint** objekat koji je prouzrokovao dogadjaj.

Event DataEntryDeactivated (DataEntry As object)

Ovaj dogadjaj se aktivira kada korisnik deaktivira objekat za unos podatka (tj. korisnik klikne van objekta za unos podatka ili koristi escape ključ da poništi editiranje, ili nakon što korisnik unese vrijednost). Parametar **DataEntry** je **GwxprocessPoint** objekat koji je prouzrokovao dogadjaj.

Event DataEntryValueEntered (DataEntry As object)

Ovaj dogadjaj se aktivira kada korisnik unese novu vrijednost u objekat za unos podatka. Parametar **DataEntry** je **GwxprocessPoint** objekat koji je prouzrokovao dogadjaj.

Event DialBeginDrag (Dial As object)

Ovaj događaj se lansira kada korisnik klikne sa tasterom miša na dole (mouse down) na Gwx32 dialu(biraču). Parametar **Dial** je **GwxRotation** objekat birača koji je prouzrokovao događaj.

Event DialEndDrag (Dial As object)

Ovaj događaj se aktivira kada korisnik otpusti taster miša na Gwx32 biraču (dial). Ovaj događaj se takodje poziva kada korisnik okrene birač sa tasterom strelice umjesto sa tasterom miša. Parametar Dial je GwxRotation objekat diala koji je prouzrokovao događaj.

Event DisplayKeyDown (KeyCode As long , Shift As Long, CancelProcessing)

Event DisplayKeyUp(KeyCode As Long, Shift As Long, CancelProcessing)

Gornji događaji se aktiviraju kada se pritisne taster (DisplayKeyDown) ili oslobodi (DisplayKeyUp). Moguće je potisnuti procesiranje događaja sa GraphWorX32 setovanjem CancelProcessing na true.

Opaska: Ovaj događaj će se aktivirati samo u slučaju kada je omogućen bilo programski (EnableRuntimeEvents) ili ručno u Format>Application Preferences> VBA tabu.

Parametri su:

KeyCode

Key kod, kao napr **vbKeyHome** (HOME taster)

Shift

Cjeli broj koji korespondira sa stanjem SHIFT ,CTRL, i ALT tastera u trenutku događaja.

Cancel processing

Varijanta koja može potisnuti dalje procesiranje događaja ako se GraphWorX32 setuje na True.

Slijedeća tabela pokazuje vrijednosti za Shift konstante

Konstanta	Vrijednost	Opis
vbShiftmask	1	bit maska SHIFT tastera
vbCtrMask	2	bit maska CTRL tastera
vbAltMask	4	bit maska ALT tastera

Event DisplayLoad()

Ovaj događaj se aktivira neposredno nakon loadovanja displeja u GraphWorX32.

Event DisplayMouseDown(Button As Long, Shift As Long, X As Single, Y As Single, CancelProcessing)

Event Display MouseMove(Button As Long, Shift As Long, X As Single, Y As Single, CancelProcessing)

Event Display MouseUp(Button As Long, Shift As Long, X As Single, Y As Single, CancelProcessing)

Gornji događaji se aktiviraju kada se taster dvaput klikne (DisplayMouseDown), ili pritisne (DisplayMouseDown), ili otpusti (DisplayMouseUp), ili se promjeni pozicija miša (DisplayMouseMove). Moguće je potisnuti procesiranje događaja sa GraphWorX32 postavljanjem CancelProcessing na True.

Opaska: Ovaj događaj će se aktivirati samo u slučaju ako je omogućen bilo programski (EnableRuntimeEvents) ili ručno u **Format>Applications Preferences> VBA** tabu.

Parametri:

button

Cijeli broj koji korespondira sa stanjem tastera miša kod kojeg je bit setovan ako je taster pritisnut nadole (down). Argument button je polje bita sa bitovima koji korespondiraju sa lijevom tasterom (bit 0), desnom tasterom (bit 1), i srednjim tasterom (bit 2). Ovi biti korespondiraju sa vrijednostima 1,2 i 4 respektivno.

On indicira sva moguća stanja tastera miša; neka, sva ili nijedan od ova tri bita može biti setovan, indicirajući da neki, svi ili nijedan od tastera miša nije pritisnut.

Shift

Cijeli broj koji korespondira sa stanjima od SHIFT, CTRL i ALT tastera. Bit se setuje ako je taster pritisnut. Shift argument je polje bita sa najmanje značajnim bitima koji korespondiraju sa SHIFT tasterom (bit 0), CTRL tasterom (bit 1) i ALT tasterom (bit 2). Ovi biti korespondiraju sa vrijednostima 1,2 i 4 . *Shift* argument indicira stanje ovih tastera. Neki, svi ili nijedan od ovih bita može biti setovan, indifirajući da neki, svi ili nijedan od tastera je pritisnut. Na primjer, ako obadva CTRL i ALT su pritisnuti, vrijednost *shift* bi bila 6.

x,y

Broj koji specificira tekuću lokaciju pontera miša. x i y vrijednosti se uvijek izražavaju u elementima koordinantnog sistema setovanog sa **ScaleHeight**, **ScaleWidth**, **ScaleLeft** i **ScaleTop** osobinama objekta.

CancelProcessing

Varijanta koja može potisnuti dalje procesiranje događaja sa GraphWorX32 ako se setuje na True.

Event DisplayUnload()

Ovaj događaj se aktivira neposredno prije nego što se displej unoaduje iz GraphWorX32 (tj. neposredno pred loadovanje novog displeja u GraphWorX32 ili prije zatvaranja GraphWorX32).

viod LanguageChanged (long LocaleID)

Ovaj događaj se aktivira kada GraphWorX32 se preključi na novi jezik. Parametar **localeID** je lokalni ID novog jezika (nije raspoloživ za displeje koji izvršavaju *GWXView32* objekte).

Event PickPostDown(Pick As Object)

Ovaj događaj se aktivira kada korisnik pritisne dole na Gwx32 pick akciju. Događaj se aktivira nakon što se akcija pick "OnDown" izvrši. Parametar Pick je GwxPick objekat od Pick dinamike koji je prouzrokovao ovaj događaj.

Event PickPostUp (Pick As Object)

Ovaj događaj se aktivira kada korisnik oslobodi (podigne prst sa tastera miša) u Gwx32 pick akciji. Ovaj događaj je aktiviran nakon što se izvrši Pick akcija "OnUp". Parametar Pick je GwxPick dinamički objekat, koji je prouzrokovao događaj.

Event PickPosrWhileDown (Pick As Object)

Ovaj događaj se aktivira dok se pick akcija drži dole, u vremenskom intervalu specificiranom u pick objektu. Događaj se aktivira nakon što se izvrši akcija "WhileDown". Parametar Pick je GwxPick objekat Pick dinamike koji je prouzrokovao događaj.

Event PickPreDown(Pick As Object)

Ovaj događaj se aktivira kada korisnik pritisne dole na Gwx32 pick akciju. Događaj se aktivira kada se izvrši pick akcija "OnDown". Parametar Pick je GwxPick objekta od Pick dinamike koja je prouzrokovala događaj.

Event PickPreUp (Pick As Object)

Ovaj događaj se aktivira kada korisnik oslobodi (taster mousa prema gore), Gwx32 pick akciju. Ovaj događaj se aktivira prije nego što se pick akcija "OnUp" izvrši.

Parametar Pick je GwxPick objekat od Pick dinamičke akcije koja je prouzrokovala događaj.

Event PickPreWhileDown(Pick As Object)

Ovaj događaj se aktivira kada pick akcija se drži dole u vremenskom intervalu specificiranom u pick objektu. Događaj se aktivira prije nego što se pick "WhileDown" akcija izvrši. Parametar Pick je GwxPick objekat od Pick dinamike koji je prouzrokovao događaj.

Event PostAnimateDisplay()

Ovaj događaj se aktivira nakon što je GraphWorX32 završio animaciju displeja.

Event PostAnimateLayer(LayerName As String)

Ovaj događaj sa aktivira nakon što GraphWorX32 je završio animaciju specificiranog stringa

Event postDeanimateDisplay()

Ovaj događaj se aktivira nakon što je GraphWorX32 završio danimaciju displeja.

Event PostDeanimateLayer(LayerName As String)

Ovaj događaj se aktivira nakon što je GraphWorX32 zvršio deanimaciju specificiranog sloja (layera).

Event PostRuntimeStart()

Ovaj događaj se aktivira kada GraphWorX32 je završio ulazak u runtime mod rada.

Event postRuntimeStop()

Ovaj događaj se aktivira nakon što je GraphWorX32 kompletno izašao iz runtime moda.

Void PreAnimateDisplay().

Ovaj događaj se aktivira neposredno prije nego GraphWorX32 animira displej (animiranje displeja je proces zahtjevanja tagova od OPC servera. Displeji se animiraju kada se loaduju za vrijeme runtime moda, i deanimiraju kada se unloaduju za vrijeme runtime moda.

Event PreAnimateDisplay()

Ovaj događaj se aktivira neposredno prije nego što GraphWorX32 animira specificirani displej (animiranje displeja je proces zahtjevanja tagova od OPC servera koji su referencirani na displeju).

Event PreAnimateLayer (LayerName As String)

Ovaj događaj se aktivira neposredno prije nego što GraphWorX32 animira specifičirani sloj displej (animiranje sloja je proces zahtjevanja tagova od OPC servera koji su referencirani na tom sloju).

Event PreDeanimateDisplay()

Ovaj događaj se aktivira neposredno prije GraphWorX32 deanimira specifičirani sloj displeja.(deanimacija displeja je proces oslobadjanja tagova koji su prethodno traženi od OPC servera).

Event preDeanimateLayer(Layer Name As String)

Ovaj događaj se aktivira prije nego što GraphWorX32 deanimira specifičirani sloj displeja (deanimacija sloja je proces oslobadjanja tagova prethodno zahtjevanih od OPC servera).

Event PreRuntimeStart()

Ovaj događaj se aktivira prije nego što GraphWorX32 se postavi u runtime mod.

Event PreRuntimeStop()

Ovaj događaj se aktivira prije nego što GraphWorX32 izađe iz runtime moda.

Event SliderBeginDrag(Slider As Object)

Ovaj događaj se aktivira kada korisnik klikne na taster miša prema dole na Gwx32 klizaču (slider). Parametar **Slider** je **GwxLocation** objekat slajdera koji je prouzrokovao događaj.

Event SliderDrag(Slider As Object)

Ovaj događaj se aktivira kada korisnik vuče Gwx32 slajder (tj. povlači miš dok je taster miša pritisnut i pozicioniran na klizaču). Parametar **Slider** je **GwxLocation** objekat slajdera koji je prouzrokovao događaj.

Event SliderEndDrag(Slider As Object)

Ovaj događaj se aktivira kada korisnik oslobodi taster miša na Gwx32 klizaču. Ovaj događaj se također poziva kada korisnik pomiče klizač sa strelica tasterima umjesto sa mišem. Parametar **Slider** je **GwxLocation** objekat klizača koji je prouzrokovao događaj.

Osobine i metode

GraphWorX32 uključuje slijedeće opšte OLE Automation klase:

- **GwxView32**
- **GwxDisplay**
- **GwxVisible**
- **GwxDynamic**
- **Gwxpoint**

GWXView32

GWXView32 je tip objekta za GraphWorX32 ActiveX kontrolu (GWXView32.ocx). Kada koristimo ActiveX ToolWorX, osobine i metode opisani u ovom poglavlju se mogu naslijediti sa novim ActiveX kontrolama. **GWXview32** objekti se mogu kreirati putem ProgID **GWXView32.GWXview32Ctrl.1**.

Osobine

AutoStartRuntime As Boolean

Kada je TRUE , ovaj objekat ili kontrola će automatski ući u runtime mod kada kontejner objekta udje u runtime mod. Ova osobina radi samo ako kontejner podržava AmbientUserMode osobinu.

HorizontalScrollbar As Boolean

Kada je TRUE, horizontalna letvica pomicanja (horizontal scrollbar) od prozora kontrole je vidljiva, kada je FALSE horizontalni scrollbar je sakriven. Ova osobina se ignoriše ako **OverrideScrollbarSettings** je FALSE.

OverrideScrolbarSettings As Boolean

Kada je TRUE, ova osobina indicira da će objekat ili kontrola nadjahati visibility seting za scrollbar, od displeja koji se trenutno puni u kontroli. Kada je FALSE, objekat ili kontrola će koristiti setinge scrollbara, definirane u trenutno napunjenom displeju.

UseAmbientBackColor As Boolean

Kada je TRUE, pozadinska boja objekta/kontrole će automatski biti postavljena da se uparuje sa pozadinskom bojom kontejnera u kojem je ovaj objekat uronjen. Ova osobina radi samo ako kontejner podržava osobinu AmbientBackColor.

VerticalScrolbar As Boolean

Kada je TRUE, vertikalni skrolbar od prozora kontrole je vidljiv , kada je FALSE vertikalni scrolbar je sakriven. Ova osobina se ignoriše ako **OverrideScrollbarSettings** je FALSE.

DisplayName As String

Ime fajla od GraphWorX32 displeja se loaduje u ovaj objekt ili kontrolu. Setovanjem ove osobine na novo ime fajla će prouzrokovati da se loaduje novi displej.

Appearance As Integer

Izgled ivice prozora. Validne vrijednosti su:

Flat = 0

3-D = 1

BorderStyle As Integer

Stil ivice prozora. Validne vrijednosti su:

No border = 0

Normal Border = 1

Metode

Function GetDisplay() As Object

Vraća **GwxDisplay** objekat za displej koji je trenutno loadovan u ovaj objekat ili kontrolu. Nakon toga možemo koristiti ovaj objekat da pristupimo svim osobinama i metodama svih objektnih tipova opisanih u ovom poglavlju.

Function ReplaceFilePath(OldSubstring As String, New Substring As String) As Long

Radi nad imenom staze parametara pick dinamike. Vraća 0 kod uspjeha i HRESULT kada nešto ne uspije:

```
'similar to ReplaceTag
'VBA example, works only in configure mode
'replaces file path in whole display

Dim Status As Long
Status = ThisDisplay.ReplaceFilePath("C:\Windows\Temp",
"D:\Temp")
'if Status <> 0 then there was no replacement performed or
an error occurs
If Status <> 0 Then
  MsgBox "No replacements"
Else
  MsgBox "Tags replaced"
End If
```

Function ReplaceHost(OldHostName As String, NewHostName As String) As Long

Radi i nad tagovima izvora podataka kao i nad atributima imena staze i zamjenjuje samo ime noda **substring** unutar URL staze. Vraća 0 kod uspjeha i HERESULT kada nešto ne uspije:

```
'similar to ReplaceTag
'VBA example, works only in configure mode
'replaces host name in whole display

Dim Status As Long
Status = ThisDisplay.ReplaceHost("Host1", "Host2")
'if Status <> 0 then there was no replacement performed or
an error occurs
If Status <> 0 Then
  MsgBox "No replacements"
Else
  MsgBox "Tags replaced"
End If
```

Function ReplaceHostEx(OldHostNameSubstring As String, NewHostNameSubstring As String, MatchCase As Boolean, MatchWholeWord As Boolean) As Long

Radi i nad tagovima izvora podataka kao i na atributima imena staze i zamjenjuje **substring** imena čvora unutar URL staze, i podržava osjetljivost na velika/mala slova, stringove sa wildcard, i matchWholeWord flag. Vraća 0 kod uspjeha i HRESULT kada nešto ne uspije.


```
'similar to ReplaceTag
'VBA example, works only in configure mode
'replaces host name in whole display, regarding "case" and
"whole words" options
```

```
Dim Status As Long
Status = ThisDisplay.ReplaceHostEx("Host1", "Host2", True,
True)
'if Status <> 0 then there was no replacement performed or
an error occurs
If Status <> 0 Then
  MsgBox "No replacements"
Else
  MsgBox "Tags replaced"
End If
```

Function ReplaceTag(OldSubstring As String, New Substring As String) As Long

Radi nad tagovima izvora podataka u dinamičkim akcijama svih displej objekata. Zamjenjuje **OldSubstring** sa **NewSubstring** u izvjesnim situacijama. Vraća 0 pri uspjehu i HRESULT kada nešto nije u redu.

Tipično korištenje ovog metoda:

```
'VBA example, works only in configure mode
'replaces tag name in whole display

Dim Status As Long
Status = ThisDisplay.ReplaceTag("gfwsim.ramp.float",
"gfwsim.sine.double")
'if Status <> 0 then there was no replacement performed or
an error occurs
If Status <> 0 Then
  MsgBox "No replacements"
Else
  MsgBox "Tags replaced"
End If
```

Long ReplaceStringInString(BSTR StringToReplace, BSTR OldSubstring, BSTR New Substring)

Zamjenjuje **OldSubstring** sa **NewSubstring** unutar StringToReplace. Vraća 0 kod uspjeha i HRESULT kada nešto ne uspije.

Sub SelectLanguage(Language As String, LanguageSubset As String, LocaleID As Long, SynchronizeLCID AS Boolean)

Sub SetGasParentCookie(newCookie As Long)

Koristi se interno za komunikaciju izmedju ActiveX kontrole i kontejnera. Korisnik ne smije modificirati.

Sub StartRuntime()

Stavlja objekat ili kontrolu u runtime mod

Sub StopRuntime()

Uzima ovaj objekat ili kontrolu van runtime moda.

GwxDisplay

GwxDisplay je tip objekta za GraphWorX32 displeje. Osobine i metodi **GwxDisplay** tipa objekta se detaljno opisani u nastavku teksta. GwxDispaly objekti se mogu kreirati pomoću progID **Gwx32.Display**.

Osobine

BackgroundColor As OLE_COLOR

Dobija ili setuje pozadinsku boju tekućeg displeja.

Redraw As Boolean

Kada je TRUE, automation poziva kada će promjena, u vizuelnom izgledu displeja, automatski osvježiti displej. Kada je FALSE, korisnik mora explicitno osvježiti displej (na primjer, korisnik može željeti da promjeni attribute mnogih objekata, i osvježi ih sve odjedanput).

ScaleMode As GWXSCALEMODE

Setuje mod skaliranja na tekući displej. Validne vrijednosti su:

FixedScale = 0
Scaleable = 1
ScaleablePreserveAspect = 2

Metode

Sub BringWindowToTop()

Dovodi glavni prozor GraphWorX32 na vrh svih prozora u z-redu. (nije na raspolaganju za displeje koji se izvršavaju u GWXview32 objektima).

Sub ClearGradientFill()

Čisti (clear) prethodno setovanu pozadinsku gradijentnu ispunu (fill)(tj. restaurira punu pozadinsku ispunu – solid background fill).

Sub CloseAllPopupWindows()

Zatvara sve trenutno otvorene pop-up prozore- (nije raspoloživa u ActiveX ToolWorX-u).

Function ClosePopupWithTitle(titleSubstring As String) As Boolean

Zatvara popup prozor koji ima specificirani substring kao dio ili cijeli tekst u title baru popup prozora. Vraća TRUE ako specificirani popup prozor je uspješno zatvoren, inače FALSE. (Nije raspoloživ u ActiveX ToolWorx-u).

Sub CloseWindow()

Zatvara prozor za ovaj displej (ovaj metod je sličan metodu ExitApplication), međutim CloseWindow takodjer radi i za displeje koji se izvršavaju u popup prozorima kao i u uronjenim (embedded) popup prozorima.

Function ConvertGwxSymbolToLibraryObject(gwxSymbol As Object, libraryObject As Object) As Boolean

Konvertuje GwxSymbol objekat u Symbol Library objekat. Kao takav, može biti unjet u jedan od fajlova kategorije u biblioteci simbola (*.sdf) pomoću automation biblioteke simbola (symbol library automation).

Opaska: Ova funkcija radi samo u GraphWorX32 dizajn modu.

Function CreateAnalogColorDynamic(visibleObjectName As String, dynamicObjectName As String, changeFill As Boolean, changeLine As Boolean, changeShadow As Boolean, startFillColor As OLE_COLOR, endFillColor As OLE_COLOR, startLineColor As OLE_COLOR, endLineColor As OLE_COLOR, startShadowColor As OLE_COLOR, endShadowColor As OLE_COLOR, defaultColorAbove As Boolean, defaultColorBelow As Boolean) As Object

Ova funkcija kreira GwxAnalogColor objekat sa specifičnim atributima i priključuje ih na vidljivi objekat sa specificiranim imenom objekta. Vraća novo kreirani **GwxAnalogColor** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ je samo u konfiguracionom modu.

Function CreateAnalogSelector(symbolName As String, dynamicObjectName As String, hiddenWhenAbove As Boolean, hiddenWhenBelow As Boolean) As Object

Kreira **GwxAnalogSelector** objekat sa specificiranim atributima i priključuje ih na **GwxSymbol** objekat sa specificiranim imenom objekta. Vraća novokreirani **GwxAnalogSelector** objekat u slučaju uspjeha, a NULL ako operacija nije uspjela. Raspoloživ samo u konfiguracionom modu.

Function CreateAnimator(symbolName As String, dynamicObjectName As String, animateWhenTrue As Boolean, visibleWhenOff As Boolean, currentFrameWhenOff As Boolean) As Object

Kreira objekat **GwxAnimator** –a sa specificiranim atributima i priključuje ga na **GwxSymbol** sa specificiranim imenom objekta. Vraća novo kreirani **GwxAnimator** objekat u slučaju uspjeha, NULL ako operacija ne uspije, Raspoloživ je samo u konfiguracionom modu.

Function CreateArc(arcType As GWXARCTYPE, centerX As Single, centerY As Single, radiusX As Single, radiusY As Single, startAngle As Single, endAngle As Single, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As GWXLINestyle, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String) As Object

Kreira i vraća **GwxArc** objekat sa specificiranim atributima, u tekućem displeju. Uglovi su specificirani u stepenima . Raspoloživ je samo u konfiguracionom modu.

Function CreateBitmap(filename As String, left As Single, top As Single, width As Single, height As Single, isTransparent As Boolean, transparentColor As Long, lineColor As Long, lineWidth As Long, lineStyle As GWXLINestyle, hasShadow As Boolean, shadowColor As Long, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String) As Object

Kreira i vraća **GwxBitmap** objekat sa specificiranim atributima, u tekućem displeju. Parametar FileName se odnosi na ime fajla tipa *.BMP. Raspoloživ je samo u konfiguracionom modu.

Function CreateButton(buttonType As GWXBUTTONTYPE, x As Single, y As Single, label As String, alignment As GWXTEXTALIGNMENT, stretchText As Boolean, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As Long, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As Long, isHidden As Boolean, objectName As String) As Object

Kreira i vraća objekat **GwxButton** sa specificiranim atributima, u tekući displej. Objekat tastera neće biti operativan sve dok **Gwxpick** objekat je priključen na njega. Raspoloživ je samo u konfiguracionom modu.

Function CreateDigitalColorDynamic(visibleObjectName As String, dynamicObjectName As String, changeColorWhenTrue As Boolean, changeFill As Boolean, changeLine As Boolean, changeShadow As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, shadowColor As OLE_COLOR, dataSource As String) As Object

Kreira **GwxDigitalColor** objekat sa specificiranim atributima i priključuje ih na vidljivi objekat sa specificiranim imenom objekta. Objekat se kreira sa jednom početnom konekcijom podatka. Dodatne konekcije podataka se mogu dodati ponovno pozivajući ovu funkciju za isti vidljivi objekat. Vraća novo kreirani **GwxDigitalColor** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ je samo u konfiguracionom modu.

Function CreateDigitalSelector(symbolName As String, dynamicObjectName As String) As Object

Kreira **GwxDigitalSelector** objekat sa specificiranim atributima i priključuje ih na **GwxSymbol** objekat sa specificiranim imenom objekta. Vraća novo kreirani **GwxDigitalSelector** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Novo kreirani objekat nema konekcija sa podacima. Da bi se dodale konekcije treba koristiti **GwxDigitalSelector::SetConnectionInfo**. Raspoloživ je samo u konfiguracionom modu.

Function CreateEllipse(left As Single, top As Single, width As Single, height As Single, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As GWXLINestyle, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String) As Object

Kreira i vraća objekat **GwxEllipse** sa specificiranim atributima u tekući displej. Raspoloživ je samo u konfiguracionom modu.

Function CreateFlashDynamic(visibleObjectName As String, dynamicObjectName As String, hideObject As Boolean, flashWhenTrue As Boolean, altStateWhenOff As Boolean, changeFill As Boolean, changeLine As Boolean, changeShadow As Boolean, altFillColor As OLE_COLOR, altLineColor As OLE_COLOR, altShadowColor As OLE_COLOR) As Object

Kreira objekat **GwxFlash** sa specificiranim atributima i priključuje ga na vidljivi objekat sa specificiranim imenom objekta. Vraća novo kreirani **GwxFlash** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ je samo u konfiguracionom modu.

Function CreateHideDynamic(visibleObjectName As String, dynamicObjectName As String, hideWhenTrue As Boolean, disableObject As Boolean) As Object

Kreira objekat **GwxHide** sa specificiranim atributima i priključuje ga na vidljivi objekat sa specificiranim imenom objekta. Vraća novokreirani objekat GwxHide u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ samo u konfiguracionom modu.

Sub CreateLayer(layerName As String)

Kreira novi sloj u displeju i doznačuje ga na ime specificirano u parametru.

Function CreateLocationDynamic(visibleObjectName As String, dynamicObjectName As String, offsetX As Single, offsetY As Single, slider As Boolean, tracking As Boolean, numberOfDetents As Integer, continuousUpdate As Boolean) As Object

Kreira objekat **GwxLocation** sa specificiranim atributima i priključuje ga na vidljivi objekat sa specificiranim imenom objekta. Parametri **offsetX** i **offsetY** se odnose na distancu koju će objekat preći od svoje tekuće lokacije. Vraća novo kreirani objekat **GwxLocation** u slučaju uspjeha, NULL ako operacija nije uspjela. Raspoloživ je samo u konfiguracionom modu.

Function CreateMetafile(filename As String, left As Single, top As Single, width As Single, height As Single, objectName As String) As Object

Kreira i vraća objekat **GwxMetafile** sa specificiranim atributima, u tekućem displeju. Parametar FileName odnosi se na ime fajla tipa WMF/EMF. Raspoloživ samo u konfiguracionom modu.

Function CreatePickDynamic(visibleObjectName As String, dynamicObjectName As String, pickAction As GWXPICKACTION, pickType As GWXBUTTONTYPE, executionTrigger As GWXEXECUTIONTRIGGER, mouseButton As GWXMOUSEBUTTON, initiallySelected As Boolean, groupName As String, filename As String, modal As Boolean, center As Boolean, value1 As String, value2 As String, value3 As String) As Object

Kreira **GwxPick** objekat sa specificiranim atributima i priključuje ga na vidljivi objekat sa specificiranim imenom objekta. Validne vrijednosti za **GWXPICKACTION** su:

PickLoadDisplay	= 0
PickDragDropLoad	= 1
PickPopupWindow	= 2
PickDownloadValue	= 3
PickToggleValue	= 4
PickLaunchApp	= 5
PickClose	= 6
PickRunScript	= 7
PickCustomCommand	= 8
PickEmbeddedWindow	= 9
PickDisplayBack	= 10
PickDisplayForward	= 11
PickSetAliases	= 12
PickAliasesDialog	= 13
PickLayerHideOn	= 14
PickLayerHideOff	= 15
PickLayerHideToggle	= 16
PickPopupUserMenu	= 17
PickSwitchLanguage	= 18
PickSelectThema	= 19
PickLaunchWebPage	= 20

Validne vrijednosti za **GWXBUTTONTYPE** su :

ButtonNormal	= 0
ButtonCheck	= 1
ButtonRadio	= 2

Validne vrijednosti za **GWXEXECUTIONTRIGGER** su:

TriggerOnDown	= 1
TriggerWhileDown	= 2
TriggerOnDnWhileDn	= 3
TriggerOnUp	= 4
TriggerOnDnOnUp	= 5
TriggerWhileDnOnUp	= 6
TriggerOnDnWhileDnOnUp	= 7

Validne vrijednosti za **GWXMOUSEBUTTON** su :

MouseButtonLeft	= 0
MouseButtonMiddle	= 1
MouseButtonRight	= 2

Zavisno od vrijednosti pickAction, neki parametri se mogu ignorisati od GraphWorX32. Vraća novo kreirani Gwxpick objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ samo u konfiguracionom modu.

Function CreatePolyline(vertices As Boolean, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As GWXLINestyle, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String) As Object

Kreira i vraća objekat **GwxLine** sa specificiranim atributima, u tekućem displeju. Parametar **vertices** je polje (array) vrijednosti sa pokretnim zarezom tako da se elementi 0,2,4,6,... polja su x-koordinate vertica linije, a elementi 1,3,5,7,.. polja su y-koordinate vertica linije. Kada se poziva ovaj metod iz C++ aplikacije, parametar **vertices** treba biti "safe array" , sa donjom granicom setovanom na 1. Raspoloživ je samo u konfiguracionom modu.

Function CreateProcessPoint(textObjectName As String, dynamicObjectName As String, dataType As GWXDATATYPE, update As Boolean, dataEntry As Boolean, hasInitialValue As Boolean, initialValue, format As String) As Object

Kreira objekat **GwxProcessPoint** sa specificiranim atributima i priključuje ga na **GwxText** objekat sa specificiranim imenom objekta. Vraća novo kreirani **GwxProcesspoint** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ samo u konfiguracionom modu.

Function CreateRectangle(left As Single, top As Single, width As Single, height As Single, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As GWXLINestyle, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String, rounded As Boolean) As Object

Kreira i vraća objekat **GwxRectangle** sa specificiranim atributima, u tekući displej. Raspoloživ samo u konfiguracionom modu.

Function CreateRotationDynamic(visibleObjectName As String, dynamicObjectName As String, startAngle As Single, endAngle As Single, pivotX As Single, pivotY As Single, clockwise As Boolean, dial As Boolean, tracking As Boolean, numberOfDetents As Integer, continuousUpdate As Boolean) As Object

Kreira objekat **GwxRotation** sa specificiranim atributima i priključuje ga na vidljiv objekat sa specificiranim imenom objekta. Parametri **pivotX** i **pivotY** su ofseti od centra objekta. Uglovi su specificirani u stepenima. Vraća novokreirani **GwxRotation** objekat u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživ je samo u konfiracionom modu.

Function CreateSizeDynamic(visibleObjectName As String, dynamicObjectName As String, sizeType As GWXSIZEDYNTYPE, clip As Boolean, startSize As Single, endSize As Single) As Object

Kreira objekat GwxSize sa specificiranim atributima i priključuje ga na vidljivi objekat sa specificiranim imenom objekta. Parametri **startSize** i **endSize** se procentualne vrijednosti u opsegu 0.0 do 1.0. Validne vrijednosti za **GWXSIZEDYNTYPE** su :

SizeLeft	= 0
SizeRight	= 1
SizeUp	= 2
SizeDown	= 3
SizeUpLeft	= 4
SizeUpRight	= 5
SizeDownLeft	= 6
SizeDownRight	= 7
SizeLeftRight	= 8
SizeUpDown	= 9
SizeLeftRightBias	= 10
SizeUpDownBias	= 11
SizeAllFour	= 12
SizeLeftRightUp	= 13
SizeLeftRightDown	= 14
SizeUpDownLeft	= 15
SizeUpDownRight	= 16

Vraća novokreirani objekat **GwxSize** u slučaju uspjeha, NULL ako operacija ne uspije. Raspoloživa je samo u konfiguracionom modu.

Function CreateStateField(textObjectName As String, dynamicObjectName As String, update As Boolean, dataEntry As Boolean, hasInitialValue As Boolean, initialValue, stateConfiguration As String, defaultState As String) As Object

Parametar **stateConfiguration** ima slijedeći format:

value1<TAB>statestring1<CR-LF>value2<TAB>statestring2<CR-LF> ... etc.

Parametar **defaultState** je string koji će se pokazati ako nije nadjeno stanje uparivanja.

Function CreateSymbol(objectName As String) As Object

Grupira sve trenutno selektirane objekte u GwxSymbol objekat , dajući rezultirajućem simbol objektu specificirano ime objekta. Vraća rezultirajući **GwxSymbol** objekat. Raspoloživa je samo u konfiguracionom modu.

Function CreateText(x As Single, y As Single, text As String, alignment As GWXTEXTALIGNMENT, stretchText As Boolean, isFilled As Boolean, fillColor As OLE_COLOR, lineColor As OLE_COLOR, lineWidth As Long, lineStyle As GWXLINESTYLE, hasShadow As Boolean, shadowColor As OLE_COLOR, edgeStyle As GWX3DEDGESTYLE, isHidden As Boolean, objectName As String) As Object

Kreira i vraća objekat **GwxText** sa specificiranim atributima , u tekući displej. Raspoloživa je samo i konfiguracionom modu.

Function CreateTimedate(textObjectName As String, dynamicObjectName As String, formatType As GWXTIMEDATEFORMATTYPE, timeFormat As String, dateFormat As String) As Object

Kreira **GwxTimeDate** objekat sa specificiranim atributima i priključuje ga **GwxText** objektu sa specificiranim imenom objekta. Vraća novo kreirani **GwxTimeDate** objekat ako je uspješna, NULL ako operacija nije uspješna. Raspoloživa je samo u konfiguracionom modu.

Function DeleteDynamic(objectName As String) As Boolean

Briše dinamički objekat sa specificiranim imenom objekta. Vraća TRUE ako je uspješna, FALSE ako je neuspješna. Raspoloživa samo u konfiguracionom modu.

Function DeleteObject(objectName As String) As Boolean

Briše vidljivi objekat sa specificiranim imenom objekta. Vraća TRUE u slučaju uspjeha, FALSE ako operacija ne uspije. Raspoloživa samo u konfiguracionom modu.

Sub DeselectAllObjects()

Deselektira sve trenutno selektirane vidljive objekte. Raspoloživa samo u konfiguracionom modu.

Function DisplayBack() As Boolean

Otvora prethodno napunjen (loadovan) displej u historijski displej fajl . Nije raspoloživa u ActiveX ToolWorX-u.

Function DisplayForward() As Boolean

Otvora slijedeći displej u historiji displej fajlova. Nije raspoloživa u ActiveX ToolWorX-u.

Sub DuplicateSelection()

Kreira duplikate od trenutno selektiranih objekata. Raspoloživa samo u konfiguracionom modu. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Sub EnableRuntimeEvents(LeftButton As Boolean, MiddleButton As Boolean, RightButton As Boolean, MouseMove As Boolean, Keyboard As Boolean)

Ovaj metod omogućava ili onemogućava startanje događaja tastature ili miša u VBA.

LeftButton

Setovati na True da dobije DisplayMouse događaje koji se odnose na akcije lijevog tastera miša.

MiddleButton

Setovati na True da se dobiju DisplayMouse događaji koji se odnose na akcije sa srednjim tasterom miša.

RightButton

Setovati na True da se dobiju DisplayMouse događaji koji se odnose na akcije desnog tastera miša.

MouseMove

Setovati na True da se dobiju događaji DisplayMouseMove

Keyboard

Setovati na True da se dobiju događaji vezani za tastaturu.

Tipično korištenje metoda:

```
Private Sub GwxDisplay_PreRuntimeStart()  
    ' Enable right click events only  
    Call ThisDisplay.EnableRuntimeEvents( _  
        False, False, True, False, False)  
End Sub
```

Sub ExitApplication()

Izlazi iz GraphWorX32 . Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Function ExportBitmapCaptureOfView(filename As String) As Boolean

Izvozi bitmap (BMP) fajl sa specificiranim imenom. Bitmapa je slika tekućeg GraphWorX32 pogleda. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Function FileNew() As Boolean

Starta novi (prazni) displej. Vraća TRUE kod uspjeha, FALSE ako akcija nije uspješna.

Function FileOpen(filename As String) As Boolean

Loaduje specificirani displej fajl . Vraća TRUE kod uspjeha, FALSE ako akcija nije uspješna. (U ActiveX ToolWorX-u ovaj metod samo otvara displej resursa, ne i fajl displej na disku).

Function FileOpenSetAliases(filename As String, aliasCommandType As GWXALIASCOMMAND, aliasCommandString As String) As Boolean

Otvora specificirani fajl i inicijalno setuje specificirane aliase. Ako je trenutno u runtime modu, aliasi će biti setovani prije nego što se zahtjevaju bilo kakvi tagovi. Parametar aliasCommandType može imati jednu od slijedećih vrijednosti:

AliasNoCommand	= 0
AliasSetFromFile	= 1
AliasSetDirect	= 2

Ako **aliasCommandType** je **AliasSetFromFile**, tada **aliasCommandString** treba biti ime fajla koje specificira tekst fajl koji sadrži tabom razdijeljene parove alias imena i alias definicija. (vidjeti takodjer i **SetAliasesFromFile()**). Ako **aliasCommandType** je **AliasSetDirect**, **aliasCommandString** treba imati slijedeći format:

```
aliasName1<TAB>aliasDef1<CR-LF>aliasName2<TAB>aliasDef2<CR-LF> ...
```

itd.

(U ActiveX ToolWorX ovaj metod samo otvara displeje resursa, ne i displeje na disku).

Function FilePrint() As Boolean

Štampa tekući displej. Vraća TRUE za uspjeh , FALSE u slučaju neuspjeha. Nije raspoloživa u ActiveX ToolWorX-u.

Function FileSave() As Boolean

Pohranjuje tekući displej. Vraća TRUE kod uspjeha, FALSE kod neuspjeha. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Function FileSaveAs(filename As String) As Boolean

Pohranjuje tekući displej koristeći specificirano ime fajla. Vraća TRUE kod uspjeha, FALSE kod neuspjeha. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima .

Sub getClientDimensionsPixels (left As long, top As Long, width As Long, height As Long).

Dobija veličinu pravougaonika GraphWorX32 klijenta i lokaciju u pikselima. Klijentski pravougaonik je površina na glavnom prozoru koja ne uključuje granice, naziv prozora (title bar), i menu bar.

Sub GetDisplayDimensions(width As Long, height As Long)

Dobija dimenzije displeja (radna površina work area/world bounds) trenutno loadovanog displeja.

Function GetDynamicObjectFromName(objectName As String) As Object

Dobija dinamički objekat sa specificiranim imenom objekta. Vraćeni objekat je tipa **GwxDynamic** ili jedan od izvedenih tipova iz **GwxDynamica** (**GwxSize**, **GwxLocation**, **GwxRotation**, **GwxHide**, **GwxFlash**, **GwxPick**, **GwxDigital color**, **GwxprocessPoint**, **GwxTimeDate**). Vraća NULL ako se ne nadje uparujući objekat.

Function GetEventScriptText(Event As GWXEVENTNAME) As String

GraphWorX32 dozvoljava udruživanje VBScript i JScript sa događjima. Skripte udružene sa događajem se izvršavaju kada je događaj trigerovan (napr. nakon runtime starta, ili prije animacije sloja). Ove skripte se mogu editirati pomoću novog integriranog toolbara za skripte, ili putem OLE Automation interfejsa. Svaki skript može biti napisan koristeći VBScript ili Jscript jezik nezavisno od drugih događaja. Ova funkcija omogućava čitanje (ne i promjenu) izvornog koda skripte koja je trenutno udružena sa specificiranim događajem.

```
enum {
    DataEntryActivated          = 0,
    DataEntryDeactivated       = 1,
    DataEntryValueEntered      = 2,
    DialBeginDrag              = 3,
    DialDrag                   = 4,
    DialEndDrag                = 5,
    DisplayLoad                = 6,
    DisplayUnload              = 7,
    LanguageChanged            = 8,
    PickPostDown               = 9,
    PickPostUp                 = 10,
    PickPostWhileDown         = 11,
    PickPreDown                = 12,
    PickPreUp                  = 13,
    PickPreWhileDown          = 14,
    PostAnimateDisplay         = 15,
    PostDeanimateDisplay       = 16,
    PreAnimateDisplay          = 17,
    PreDeanimateDisplay        = 18,
    PreAnimateLayer            = 19,
    PostAnimateLayer           = 20,
    PreDeanimateLayer         = 21,
    PostDeanimateLayer         = 22,
```

```

        PreRuntimeStart      = 23,
        PostRuntimeStart     = 24,
        PreRuntimeStop       = 25,
        PostRuntimeStop      = 26,
        SliderBeginDrag      = 27,
        SliderEndDrag        = 28,
        SliderDrag            = 29
    } GWXEVENTNAME;

```

Function GetEventScriptType(Event As GWXEVENTNAME) As GWXSCRIPTTYPE

GraphWorX32 u tekućoj verziji 7.1 omogućava udruživanje VBScripta i JavaScripta sa događajima. Skripte udružene sa događajem se izvršavaju kada se događaj trigeruje (napr. nakon što runtime starta, ili prije animacije sloja). Ove skripte se mogu editirati pomoću novo integriranog toolbara za skripte ili putem OLE Automation interfejsa. Svaki skript se može pisati koristeći VBScript ili JavaScript jezik nezavisno od drugih događaja. Ova funkcija dozvoljava čitanje (ne i promjenu) izvornog koda skripte koji je trenutačno udružen sa specificiranim događajem.

```

enum {
    ScriptTypeNone      = -1,
    ScriptTypeVBA       = 0,
    ScriptTypeVBScript  = 1,
    ScriptTypeJScript   = 2
} GWXSCRIPTTYPE;

```

```

'VBA example, works in configure and runtime mode
'VBA example creates VB script on event PostRuntimeStart

'sets PostRuntimeStart event script type to VB script
ThisDisplay.SetEventScriptType _
    PostRuntimeStart, ScriptTypeVBScript
'adds text to the script
ThisDisplay.SetEventScriptText PostRuntimeStart, _
    "MsgBox " & Chr(34) & "Test OK" & Chr(34)
MsgBox ThisDisplay.GetEventScriptType(PostRuntimeStart)
MsgBox ThisDisplay.GetEventScriptText(PostRuntimeStart)

```

Function GetFileContainsVBA() As Boolean

Funkcija **GetFileContainsVBA()** vraća TRUE vrijednost ako tekući displej sadrži VBA dio i FALSE ako ne sadrži VBA dio.

Prisustvo VBA dijela se kontroliše loadovanjem non-VBA displeja i pohranjivanjem displeja kao non-VBA , ili korištenjem **FileSaveAsType(1)** (sadrži VBA) , ili **FileSaveAsType(0)** (ne sadrži VBA) automation metoda.

Function GetFileName() As String

Vraća ime tekućeg loadovanog displeja.

Function GetFilePath() As String

Vraća stazu gdje je lociran tekuće loadovani fajl . Povraćena staza ne uključuje ime fajla tekuće loadovanog displeja.

Function GetFileVersion() As Long

Funkcija GetFileVersion() vraća slijedeće vrijednosti:

- -1 .. OpenFileError (&HFFFFFFF)
- 0 .. Ver520
- 1 .. Ver600
- 2 .. Ver600SP
- 3 .. VerMGraphics40
- 4 .. Ver610
- 9999 .. VerLatest (&H270F)

Function GetFrameWindowHandle() As Long

Dobija handle za prozor od okvira prozora u kojem se izvršava GwxDisplay objekat. Nije raspoloživa u ActiveX ToolWorX-u

Function GetGASThemes() As String

Dobija tekuće setovane GlobalAliasing teme u istom formatu kao što su generisane od strane Browsera tema:

```
'VBA/VB Script example, works only in runtime mode
'example for setting and getting themes (Floors theme is
changed in child windows)
ThisDisplay.SetGASThemes _
    "Buildings=BuildingB;Floors=Floor2/+1;"
MsgBox ThisDisplay.GetGASThemes
```

Function GetHeadObject() As Object

Dobija objekat sa vidljivom glavom (head visible object). U konfiguracionom modu objekt glave (head object) je selektirani objekat sa punim plavim i crvenim hvataljkama za promjenu veličine. U runtime modu, objekat glava je pickable objekat koji trenutno ima fokus. Ako nema objekta glave, ovaj metod će vratiti NULL:

Function GetLoadTab(index As Long, label As String, File As String, AliasCmdType As Long, AliasCmdString As String, GlobalAliasCmdString As String) As Boolean

Vraća TRUE kada je informacija LoadTab uspješno dobijena. Indeks parametar je indeks baziran na nuli od load taba. Tab informacija se vraća putem setovanja preostalih uključenih parametara.

Function GetLoadTabsCount() As Long

Vraća broj load Tabu u displeju.

Function GetNumberOfTopLevelVisibles() As Long

Vraća broj vidljivih objekata u korijenu simbola displeja (referenciranih kao "Top Level Visibles", pošto su oni objekti na najvišem nivou hijerarhije simbola).

Function GetObjectTypeName() As String

Vraća ime tipa ovog objekta ("GwxDisplay")

Function GetOpenPopupByIndex(popupIndex As Long) As Object

Vraća GwxDisplay objekat od otvorenog popup ili uronjenog popup prozora , sa specificiranim nula-baziranim indeksom (**popupIndex** , indeksira listu popupa i uronjenih (embedded) popupa koji su u vlasništvu ovog displeja). Nije raspoloživa u ActiveX ToolWorX-u.

Function GetOpenPopupCount() As Long

Vraća broj otvorenih popup i uronjenih popup prozora koji su u vlasništvu ovog displeja . Nije raspoloživa u ActiveX ToolWorX-u.

Function GetOpenPopupWithTitle(titleSubstring As String) As Object

Dobija displej objekat otvorenog popup prozora koji ima ovaj specificirani substring kao dio ili čitav tekst u nazivnoj letvici (title bar) od popup prozora. Vraća objekat loadovanog displeja u popup prozor (tipa **GwxDisplay**) u slučaju uspjeha, NULL u slučaju neuspjeha. Nije raspoloživa u ActiveX ToolWorX-u.

Function GetParentDisplay() As Object

Ako se ovaj displej objekat izvršava u popup prozoru ili uronjenom prozoru, ovaj metod vraća displej objekat od roditeljskog displeja (roditeljski (parent) display je onaj displej koji je otvorio popup prozor). Ako displej se ne izvršava u popup/embedded prozoru, ovaj metod vraća NULL . Nije raspoloživa u ActiveX ToolWorX-u.

Function GetPointObjectFromName(pointName As String) As Object

Dobija objekat tačke podatka sa specificiranim imenom tačke. Vraćeni objekat je tipa **GwxPoint**. Vraća NULL ako niti jedan uparujući objekat nije nadjen.

Function GetSystemWideLanguage(Language As String, LanguageSubset As String, LocaleID As Long, SynchronizeLCID As Boolean) As Boolean

Dobija trenutno izabrani jezik za GENESIS aplikaciju.

Function GetTransparency(TransparencyEnabled As Boolean, transparentColor As OLE_COLOR, AlphaEnabled As Boolean, AlphaBlending As Integer) As Boolean

Dobija informaciju o setinzima transparentije. Ova funkcija radi samo u GraphWorX32 i Win2K ili novijem OS.

```
'VBA example, works in configure and runtime mode
'this VBA example gets transparency and translucency
settings

Dim    TE As Boolean, _
        TC As OLE_COLOR, _
        AE As Boolean, _
        AB As Integer

ThisDisplay.GetTransparency TE, TC, AE, AB
MsgBox "Transparency: " & TE & vbCrLf & "Color: " & _
        TC & vbCrLf & "Translucency: " & AE & _
        vbCrLf & "Color: " & AB
```

void GetViewDimensions(long* left, long* top, long* width, long* height)

Dobija veličinu i lokaciju pravougaonika u GraphWorX32 ekranu (view)

Function GetVisibleObjectFromIndex(index As Long) As Object

Vraća vidljiv najveći nivo (Top level visible), sa datim nula-baziranim indeksom. Objekat sa indeksom nula je objekat koji je najdalji u z-nizu. Ova funkcija je korisna za iteraciju kroz sve vidljive objekte najvišeg nivoa.

Function GetVisibleObjectFromName(objectName As String) As Object

Dobija vidljivi objekat sa specificiranim imenom objekta. Vraćeni tip objekta će biti **GwxVisible** ili jedan od izvedenih tipova iz **GwxVisible** (**GwxRectangle**, **GwxEllipse**, **GwxLine**, **GwxText**, **GwxArc**, **GwxBitmap**, **GwxMetafile**, **GwxSymbol**, **GwxOleObject**, **GwxButton**). Vraća NULL ako se ne nadje uparenje (matching),

Function GetVisibleObjectFromPosition(x As Long, y As Long, GoIntoSymbol As Boolean) As Object

Vraća objekat kod specificirane pozicije u displeju. Starta od korijena displeja i ide dublje, uključujući slojeve sve dok se ne nadje najviši objekat. Ako se argument GoIntoSymbol setuje na True, traženje se nastavlja sve dok konačnog vidljivog objekta na datoj poziciji. Inače, funkcija će vratiti simbol najvišeg nivoa koji sadrži objekat.

```
'VBA example, works in configure and runtime mode
'this example gets the lowest level visible at random
position

Dim w As Long, h As Long
Dim o As Object
ThisDisplay.GetDisplayDimensions w, h
'gets display width and height
If ThisDisplay.GetNumberOfTopLevelVisibles > 0 Then
'checks if display has at least one visible object
! and runs a loop for find one
  Do
    Set o = ThisDisplay.GetVisibleObjectFromPosition( _
      Rnd * w, Rnd * h, True)
'tries to get visible object from random position - _
!if object is GwxSymbol, it goes into symbol and _
!continues searching
  Loop Until Not (o Is Nothing)
'after any object is found, loop ends and example _
!shows name of the object
  MsgBox o.objectName
Else
  MsgBox "Example needs at least one visible object"
End If
```

Sub GetWindowDimensionsPercent(left As Single, top As Single, width As Single, height As Single)

Dobija GraphWorX32 veličinu i lokaciju glavnog prozora kao procenat totalne veličine ekrana. Dobijeni parametri će imati vrijednosti u opsegu 0.0 do 1.0.

Sub GetWindowDimensionsPixels(left As Long, top As Long, width As Long, height As Long)

Dobija veličinu i lokaciju u pikselima glavnog prozora GraphWorX32 .

Function GetWorkingDirectory() As String

Vraća GENESIS32 radni direktorij. Ovo je direktorij gdje GENESIS32 aplikacije kao GraphWorX32 traže displej fajlove i druge konfiguracione fajlove.

Function GwxGetVbaProject() As Object

Vraća VBA Project interfejs. Nije raspoloživ za displeje koji se izvršavaju u GWXview32 objektima

Sub HideWindow()

Sakriva glavni prozor GraphWorX32

Function InsertLibraryObject(libraryObject As Object) As Boolean

Function InsertLibraryObject(libraryObject As Object) As Boolean

Umeće objekat biblioteke simbola u displej. Ovaj bibliotečki objekat se može kreirati ili dobiti putem OLE Automation biblioteke simbola .

Opaska: Ova funkcija radi samo u GraphWorX32 dizajn modu.

Function InsertLibraryObjectByIndex(category As String, index As Long) As Boolean

Omogućava lagano unošenje bilo kojeg simbola iz biblioteke simbola putem OLE Automation. Parametar **category** korespondira sa imenom fajla kategorije biblioteke simbola (*.sdf fajl),; parametar **index** je nula-bazirani indeks simbola u kategoriji.

Opaska: Ova funkcija radi samo u dizajn modu GraphWorX32.

Function InsertLibraryObjectByName(category As String, symbol As String) As Boolean

Dozvoljava lagano umetanje bilo kojeg simbola iz biblioteke simbola putem OLE Automation. Parametar **category** korespondira sa imenom kategorije fajl a biblioteke simbola (*.sdf); parametar **symbol** je ime simbola iz fajl a kategorije.

Opaska: Ova funkcija radi samo u dizajn modu GraphWorX32.

Tipično korištenje ovog metoda:

```

Dim SL As SymbolLibrary.SymbolLibrary
Dim SC As SymbolLibrary.SymbolCategory
Dim o As Object
Set SL = New SymbolLibrary.SymbolLibrary
Set SC = SL.GetCategoryFromIndex(0)
If SC.GetNumberOfSymbols > 0 Then
    ThisDisplay.InsertLibraryObjectByIndex _
        Dir(SC.GetCategoryName), _
        Round(Rnd * SC.GetNumberOfSymbols)
    ThisDisplay.InsertLibraryObjectByName _
        Dir(SC.GetCategoryName), _
        SC.GetSymbolName(Round(Rnd * SC.GetNumberOfSymbols))
Set o = SC.GetSymbolFromIndex( _
    Round(Rnd * SC.GetNumberOfSymbols))
ThisDisplay.InsertLibraryObject o
Else
    MsgBox "Example needs at least one symbol"
End If

```

Function InsertOleObject(progID As String, objName As String) As Boolean

Ova funkcija omogućava nam da kreiramo OLE Automation objekat i nakon toga ga unesemo u GraphWorX32 displej. Ovaj automation interfejs se može pozivati preko VBA ili VBScript/Jscript ili C++. Ovaj metod zahtjeva da korisnik obezbjedi ime objekta koji želi da kreira. Korisnik može kreirati bilo koju vrstu objekta ali mora da zna ime. Na primjer, ICONICS koristi slijedeće objekte:

- "ICONICS.SCRAAlarmCtrl.1"
- "ICONICS.Event32.1"
- "ICONICS.Periodic32.1"
- "GWXGAUGE.GWXGaugeCtrl.1"
- "GWXSWITCH.GWXSwitchCtrl.1"
- "TWXViewer.TWXViewerCtrl.1"
- "AWXview32.AWXview32Ctrl.1"
- "GWXVIEW32.GWXview32Ctrl.1"
- "GWXNUMERIC.GWXNumericCtrl.1"
- "GWXVESSEL.GwxVesselCtrl.1"
- "ICONICS.FunctionalKeyCtrl.1"
- "MarqueeLaunch.MarqueeCtl.1"
- "PopupLaunch.PopupCtl.1"

Function IsEmbeddedPopupWindow() As Boolean

Ako ovaj displej se izvršava u uronjenom popup prozoru ova funkcija je TRUE, a u drugim slučajevima je FALSE. Nije raspoloživa u ActiveX ToolWorX-u.

Function IsModified() As Boolean

Vraća TRUE ako je displej bio modifikovan nakon što je posljednji put bio pohranjen (saved), u drugim slučajevima vraća FALSE. Nije raspoloživa u ActiveX ToolWorX-u.

Function IsRuntimeMode() As Boolean

Vraća TRUE ako je GraphWorX32 trenutno u runtime modu, inače će vratiti FALSE.

Sub LogicalPointToClient(x As Long, y As Long)

Konvertuje GraphWorX32 logičke koordinate u klijentove koordinate. Nije raspoloživa u ActiveX ToolWorX-u.

Sub LogicalPointToScreen(x As Long, y As Long)

Konvertuje GraphWorX32 lokalne koordinate u ekranske koordinate. Nije raspoloživa u ActiveX ToolWorX-u.

Sub MaximizeWindow()

Maksimizira GraphWorX32 glavni prozor. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Sub MinimizeWindow()

Minimizira GraphWorX32 glavni prozor. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Function MoveSelectionBackward() As Boolean

Pomjera izabrani objekat unatrag za jednu poziciju. Koristi se da vizuelno sortira objekte i stavi jedan objekat ispod drugoga.

Function MoveSelectionForward() As Boolean

Pomjera izabrani objekat unaprijed za jednu poziciju. Koristi se da vizuelno sortira objekte i da stavi jedan objekat iznad drugoga.

Function MoveSelectionToBack() As Boolean

Pomjera izabrani objekat iza svih drugih objekata na displeju.

Function MoveSelectionToFront() As Boolean

Pomjera izabrani objekat ispred svih drugih objekata na displeju.

Function OpenEmbeddedWindow(filename As String, center As Boolean, hidden As Boolean) As Object

Otvara GraphWorX32 uronjeni prozor. Vraća objekat displeja loadovan u uronjeni prozor (tipa GwxDisplay) u slučaju uspjeha, NULL u slučaju neuspjeha. Nije raspoloživa u ActiveX ToolWorX-u.

Function OpenEmbeddedWinSetAliases(filename As String, center As Boolean, hidden As Boolean, aliasCommandType As GWXALIASCOMMAND, aliasCommandString As String) As Object

Otvara uronjeni popup prozor sa specificiranim displej fajlom i inicijalno setuje specificirane aliase. Ako se trenutno izvršava u runtime modu, aliasi će biti setovani prije nego što se zahtjevaju tagovi. Nije raspoloživa u ActiveX ToolWorX-u.

Function OpenPopupWindow(filename As String, modal As Boolean, center As Boolean, hidden As Boolean) As Object

Otvara GraphWorX32 popup prozor. Vraća objekat displeja loadovanog u popup prozor (ili tip **GwxDisplay**) u slučaju uspjeha, NULL u slučaju neuspjeha. Nije raspoloživa u ActiveX ToolWorX-u

Function OpenPopupWinSetAliases(filename As String, modal As Boolean, center As Boolean, hidden As Boolean, aliasCommandType As GWXALIASCOMMAND, aliasCommandString As String) As Object

Otvara popup prozor sa specificiranim displej fajlom i inicijalno setuje specificirane aliase. Ako je trenutno u runtime modu, aliasi će biti setovani prije nego što su tagovi zahtjevani. Nije raspoloživa u ActiveX ToolWorX-u

Function OpenSetAliasesDialog(aliasCommandType As GWXALIASCOMMAND, aliasCommandString As String, objectName As String) As Boolean

Otvara Set Aliases dijalog. Ovaj dijalog se koristi da omogući operatoru da unese nove alias definicije za set specificiranih alias imena. Za opise **aliasCommandType** i **aliasCommandString**, pogledati **FileSet Aliases()**. Parametar **aliasCommandString** je primarno sredstvo konfigurisanja dijaloga. Za svako jedinstveno alias ime specificirano u **aliasCommandString**, bit će tab dodat na dijalog boks. Alias definicije specificirane sa alias imenima će biti izlistane u odgovarajućem kombo boksu strane osobina (proprety page). Parametar **ObjectName** specificira na koji vidljivi objekat će se primjeniti novi alias setinzi (ako nije specificirano ime objekta, novi alias setinzi će se primjeniti na cjelokupni displej). Nije raspoloživa u ActiveX ToolWorX-u

Function OpenTagBrowser(hWndParent As Long, tagName As String) As Boolean

Otvora browser tagova sa specificiranom ručkom (handle) prozora kao prozor-roditelj. Parametar **tagName** prima ime taga koji je specificirao korisnik. Vraća TRUE ako je korisnik pritisnuo OK taster, FALSE ako korisnik je pritisnuo Cancel taster.

Function PopAllCurrentSymbol() As Boolean

iskače sa tekućeg nivoa editiranja simbola skroz unatrag do korijenskog nivoa. Vraća TRUE kod uspjeha, FALSE kod neuspjeha. Raspoloživa samo u konfiguracionom modu.

Function PopCurrentSymbol() As Boolean

Iskače za jedan nivo sa tekućeg nivoa editiranja simbola. Vraća TRUE kod uspjeha, FALSE kod neuspjeha . Raspoloživa samo u konfiguracionom modu.

Function QueryDataType(dataSource As String, dataType As GWXDATATYPE) As Boolean

Query OPC server za donje i gornje granice opsega vrijednosti od specificiranog imena taga (**dataSource**).

Sub RefreshWindow()

Ponovno iscrtava vidljivi dio GraphWorX32 displeja.

Sub RemoveLayer(layerName As String)

Otklanja specificirani sloj iz displeja.

Function ReplaceFilePath(OldSubstring As String, NewSubstring As String) As Long

Radi nad parametrima imena staze od pick dinamike. Vraća 0 u slučaju uspjeha i HRESULT kada nešto nije u redu.

Function ReplaceHost(OldHostName As String, NewHostName As String) As Long

Radi i nad tagovima izvora podataka i atributima imena staza i zamjenjuje **substring** imena čvora samo unutar URL staze. Vraća 0 u slučaju uspjeha i HRESULT kada nešto ne uspije.

Function ReplaceHostEx(OldHostNameSubstring As String, NewHostNameSubstring As String, MatchCase As Boolean, MatchWholeWord As Boolean) As Long

Radi i nad tagovima izvora podataka i atributima imena staza i zamjenjuje **substring** imena čvora samo unutar URL staze, i podržava osjetljivost na mala/velika slova, wildcard stringove i MatchWholeWord flag. Vraća 0 u slučaju uspjeha i HRESULT kada nešto ne uspije.

Function ReplaceTag(OldSubstring As String, NewSubstring As String) As Long

Radi nad tagovima izvora podataka u dinamičkim akcijama svih displej objekata. Zamjenjuje **OldSubstring** sa **NewSubstring** u izvjesnim situacijama. Vraća 0 u slučaju uspjeha, HRESULT kada nešto ne uspije.

Tipično korištenje ovog metoda:

```
'VBA example, works only in configure mode
'replaces tag name in whole display

Dim Status As Long
Status = ThisDisplay.ReplaceTag("gfwsim.ramp.float",
"gfwsim.sine.double")
'if Status <> 0 then there was no replacement
'performed or an error occurs
If Status <> 0 Then
  MsgBox "No replacements"
Else
  MsgBox "Tags replaced"
End If
```

long ReplaceStringInString (BSTR StringToReplace, BSTR OldSubstring, BSTR NewSubsring)

Zamjenjuje **OldSubstring** sa **NewSubstring** unutar **StringToreplace**. Vraća 0 u slučaju uspjeha i HRESULT kada nešto ne uspije.

Sub RestoreWindow()

Restaurira veličinu i poziciju (ne-minimiziranog/ne-maksimiziranog) prozora . Nije raspoloživa za displeje koji se izvršavaju u GWXView32 objektima.

Function SelectPaletteColor(SelectedColor As OLE_COLOR) As Boolean

Otvora GraphWorX32 dijalog boks kolor palete. Parametar SelectedColor prima boju koju je selektirao korisnik. Vraća TRUE ako korisnik pritisne OK taster, FALSE ako

korsnik pritisne Cancel taster. Nije raspoloživa za displeje koji se izvršavaju u GWXView32 objektima.

Function SelectPaletteColorRGB(red, green, blue) As Boolean

Popupira dijalog sa kolor paletom omogućavajući korisniku da izabere jednu od boja. Vraća TRUE ako je boja izabrana, vraća FALSE u svim drugim slučajevima. Svaka boja pokazana na monitoru se može dekomponirati u tri komponente : crvena(red) ,zelena(green) i plava(blue). Svaka od ovih komponenta boja ima intenzitet u opsegu 0-255. Ako korisnik izabere boju, 3 parametra su popunjena sa red,green i blue kompozicijom izabrane boje.

Naprimjer, ako korisnik izabere svjetlo žutu boju, tada ćemo dobiti povratno slijedeće:

Red=255

Green=255

Blue=0

Opaska: Ovaj metod radi u Visual Basic za aplikacije samo ako parametri su deklarirani kao Variant.

Function SetAliasDefinition(aliasName As String, newDefinition As String) As Boolean

Setuje alias definiciju specificiranih alias imena za sve dinamičke objekte na displeju. Ova funkcija se može koristiti u runtime modu da lagano promjeni konekcije sa podacima dinamičkih objekata "u letu" (on the-fly). Vratice FALSE ako se nisu našla uparajuća alias imena, TRUE u suprotnom slučaju.

Function SetAliases(commandString As String) As Boolean

Setuje višestruke alias definicije za cijelokupni displej. Parametar **commandString** ima slijedeći format:

```
aliasName1<TAB>aliasDef1<CR-LF>aliasName2<TAB>aliasDef2<CR-LF> ...  
etc.
```

Function SetAliasesFromFile(filename As String) As Boolean

Setuje višestruke alias definicije za cijeli displej. Parametar **filename** specificira tekst fajl koji sadrži parove alias imena i alias definicija razdjeljene sa tabom (format ovog fajla je isti format kao i kod stringa koji se prenosi kod **GwxDisplay Set Aliases()** metoda. Fajlovi ovog tipa se mogu kreirati u **Notepadu**, mogu se izvesti iz **Excela** (pohranjeno kao tab delimitirani), i mogu se kreirati koristeći **Gwx32 Alias File Editor**.

Sub SetDisplayDimensions(width As Long, height As Long)

Setuje dimenzije displeja (radne površine(work area /world bounds) na tekućem loadovanom displeju.

Sub SetEventScriptText(Event As GWXEVENTNAME, text As String)

GraphWorX32 u Verziji 7.1 dozvoljava pridruživanje VBScript i Jscript sa događajima. Skripte pridružene sa događajem se izvršavaju kada je događaj trigerovan (napr. , nakon starta runtime-a, ili prije nego što se sloj animira). Ove skripte se mogu editirati putem novog integriranog skript toolbara ili putem OLE Automation interfejsa. Svaki skript se može pisati koristeći VBScript ili Jscript jezik nezavisno od drugih događaja. Ova funkcija omogućava i pisanje (promjenu) izvornog skript koda koji je u tom trenutku izabran za specificirani događaj.

Sub SetEventScriptType(Event As GWXEVENTNAME, Type As GWXSCRIPTTYPE)

GraphWorX32 u verziji 7.1 omogućava udruživanje VBScript i Jscript događaja. Skripte udružene sa događajem se izvršavaju kada se događaj trigeruje (napr. , nakon starta runtime-a, ili prije animacije sloja). Ove skripte se mogu editirati putem novo integriranog skript toolbara ili putem OLE automation intefejsa. Svaki skript se može pisati korištenjem VBScript ili Jscript jezika nezavisno od drugog događaja. Ova funkcija omogućava i pisanje (promjenu) izvornog skript koda koji je u tom trenutku izabran za specificirani događaj.

```
'VBA example, works in configure and runtime mode
'VBA example creates VB script on event PostRuntimeStart

ThisDisplay.SetEventScriptType _
    PostRuntimeStart, ScriptTypeVBScript
ThisDisplay.SetEventScriptText _
    PostRuntimeStart, "MsgBox " & Chr(34) & _
    "Test OK" & Chr(34)
```

Odgovarajući primjer u VBScript-u

```
ThisDisplay.SetEventScriptText 23,"msgbox ""test"" "
ThisDisplay.SetEventScriptType 23,1
```

Function SetFileVersion(NewFileVersion As GWXSETFILEVERSION) As GWXGETFILEVERSION

Funkcija **SetFileVersion()** setuje verziju displeja za slijedeće pohranjivanje. Primjetimo da je set verzija podržan kod pohranjivanja manji nego set verzija detektovan kod čitanja fajla. Drugim riječima GraphWorX32 detktuje više verzija fajlova , ali dozvoljava pohranjivanje u novije verzije, verzije prije 5.20 nisu podržane.

SetFileVersion() input values

- 2 .. SetVer600SP
- 3 .. SetVerMGraphics40
- 4 .. SetVer610
- 9999 .. SetVerLatest (&H270F)

Automation interfejs je vrlo moćan i može biti korišten za automatizovano procesiranje displeja.

Visual Basic Code Sample

```
Private Sub DisplayVersion()
    ' Specify the path and filename of an existing display
    Const strPath As String = _
        "C:\PathToYourDisplay\"
    Const strFilename As String = _
        strPath & "YourDisplay.gdf"
    Const SetVer600SP as Long = 2
    Const VerLatest as Long = 9999

    ' Create GWX instance
    Dim g As New Gwx32.GwxDisplay

    ' Check if GWX was created
    If g Is Nothing Then
        MsgBox "Failed to create GWX"
        Exit Sub
    End If

    ' Show GWX in the front and load the requested display
    Call g.ShowWindow
    Call g.BringWindowToTop
    Call g.FileNew
    Call g.FileOpen(strFilename)

    ' Check the file version
    Dim ver As long
    ver = GetFileVersion(g)

    ' If this is the latest version, save in the
    ' version 6.00
    If (ver = VerLatest) Then
```

```

        g.SetFileVersion(SetVer600SP)
        g.FileSave
    End If

    ' Exit GWX, the FileNew suppresses the message box
    ' if to save changes.
    g.FileNew
    g.ExitApplication
End Sub

' Get file version and trace the current version in the
' Immediate window for debugging purposes.
Private Function GetFileVersion(g As GwxDisplay) As Long
    Const OpenFileError as Long = -1
    Const Ver520 as Long = 0
    Const Ver600 as Long = 2
    Const Ver600SP as Long = 2
    Const VerMGraphics40 as Long = 2
    Const Ver610 as Long = 2
    Const VerLatest as Long = 9999

    Dim ver As long

    ' Check the version
    ver = g.GetFileVersion
    Select Case (ver)
        Case OpenFileError:
            Debug.Print "Version: FileOpenError"
        Case Ver520:
            Debug.Print "Version: Ver520"
        Case Ver600:
            Debug.Print "Version: Ver600"
        Case Ver600SP:
            Debug.Print "Version: Ver600SP"
        Case VerMGraphics40:
            Debug.Print "Version: VerMGraphics40"
        Case Ver610:
            Debug.Print "Version: Ver610"
        Case VerLatest:
            Debug.Print "Version: VerLatest"
    End Select

    ' Check if the display contains VBA
    If g.GetFileContainsVBA Then
        Debug.Print "VBA: Yes"
    Else
        Debug.Print "VBA: No"
    End If

    GetFileVersion = ver
End Function

Function SetGASThemes(Themes As String) As Boolean

```

Postavlja Globalne alias teme. Format je isti kao i onaj koji generira browser tema, tako da se preporučuje korištenje Browsera tema da se generiše željena kombinacija a zatim pomoću copy/paste unese rezultat u ovu funkciju kao string.

```
'VBA/VB Script example, works only in runtime mode
'example for setting and getting themes (Floors theme is
changed in child windows)
ThisDisplay.SetGASThemes
"Buildings=BuildingB;Floors=Floor2/+1;"
MsgBox ThisDisplay.GetGASThemes
```

Function SetGradientFill(style As GWXGRADIENTSTYLE, color1 As OLE_COLOR, color2 As OLE_COLOR, isTwoColor As Boolean, brightness As Single, reverseColors As Boolean, steps As Integer, horizontalOffset As Single, verticalOffset As Single) As Boolean

Setuje gradijentnu ispunu (fill) za pozadinu displeja.
Parametar **style** može imati jednu od slijedećih vrijednosti :

GradientHorizontal	= 0
GradientVertical	= 1
GradientSquare	= 2

Ostatak parametara korespondira sa onima u konfiguracionom korisničkom interfejsu za gradijentnu ispunu.

Function SetHeadObject(objectName As String) As Boolean

Setuje glavu objekta (head object) da bude vidljiv objekat sa specificiranim imenom. Vraća TRUE kod uspjeha, FALSE kod neuspjeha.

Function SetLayerOverrideHide(layerName As String, hidden As Boolean) As Boolean

Setuje osobinu sakrivanja mogućnosti nadjahivanja sloja (layer override hide) specificiranog sloja displeja.

Sub SetModifiedFlag(Modified As Boolean)

Modificirani flag se koristi da odredi da li displej je bio modificiran nakon što je posljednji put bio pohranjen. Ako je modificirani flag setovan na true, tada GraphWorX32 će zahtijevati da se pohrane promjene na displeju , prije nego što se zatvori displej ili loaduje novi displej.

Function SetRuntimeFocusProperties(ShowHandCursor As Boolean, ShowFocusRectangle As Boolean, fillColor As OLE_COLOR, BorderColor As OLE_COLOR) As Boolean

Setuje attribute za naglašavanje (highlight) od runtime fokusa.

Function SetRuntimeUpdateRate(updateRate As Long) As Boolean

Setuje brzinu ažuriranja OPC podataka za displej.

Function SetSystemWideLanguage(Language As String, LanguageSubset As String, LocaleID As Long, SynchronizeLCID As Boolean) As Boolean

Setuje GENESIS jezik i jezički subset za dokument, LocaleID za aplikaciju, i flag ako je potrebno sinhronizovati jezik dokumenta sa aplikacijom.

Function SetTooltipProperties(ShowDynamicTips As Boolean, ShowDynamicObjectName As Boolean, ShowDynamicObjectDescription As Boolean, ShowDataSourceName As Boolean, ShowDataSourceValue As Boolean, ShowStaticTooltips As Boolean, ShowStaticObjectName As Boolean, ShowStaticObjectDescription As Boolean) As Boolean

Setuje osobine tooltipa za displej.

Function SetTransparency(EnableTransparency As Boolean, transparentColor As OLE_COLOR, EnableAlpha As Boolean, AlphaBlending As Integer) As Boolean

Setuje transparentnost displeja u runtime-u. Radi samo u GraphWorX32 na Win2K i novijim verzijama OS.

```
'VBA example, works in runtime mode
'this example toggles transparency and translucency settings

Dim TE As Boolean, TC As OLE_COLOR, AE As Boolean, _
    AB As Integer
ThisDisplay.GetTransparency TE, TC, AE, AB
'following lines enable/disable transparency and
translucency
If TE Then TE = False Else TE = True
If AE Then AE = False Else AE = True
'if translucency is enabled, sets alpha blending value to
100 of 255
If AE Then AB = 100 Else AB = 0
'applies changed values
ThisDisplay.SetTransparency TE, TC, AE, AB
MsgBox "Transparency: " & TE & vbCrLf & "Color: " & _
    & TC & vbCrLf & "Translucency: " & _
    AE & vbCrLf & "Color: " & AB
```


Sub SetViewDimensions(left As Long, top As Long, width As Long, height As Long)

Setuje veličinu i lokaciju pravougaonika u GraphWorX32 pogledu (view). Dimenzije view-a definiraju koji dio radne površine/cijelog displeja je vidljiv.

Setovanje view dimenzija se može koristiti da zumira i panira pogled displeja.

Sub SetWindowDimensionsPercent(left As Single, top As Single, width As Single, height As Single)

Setuje veličinu i lokaciju glavnog prozora GraphWorX32 kao procenat totalne veličine ekrana. Parametri trebaju biti vrijednosti u opsegu 0.0 do 1.0. Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Sub SetWindowDimensionsPixels(left As Long, top As Long, width As Long, height As Long)

Setuje veličinu i poziciju u pikselima glavnog prozora GraphWorX32 . Nije raspoloživa za displeje koji se izvršavaju u GWXview32 objektima.

Sub ShowWindow()

Pokazuje glavni prozor GraphWorX32

Sub StartRuntime()

Postavlja GraphWorX32 u runtime mod.

Sub StopRuntime()

Uzima GraphWorX32 iz runtime moda (u konfiguracioni mod).

Function TestCustomSecurityItem(customString As String) As Boolean

Vraća TRUE ako tekući korisnik ima sigurnosne privilegije za specificirani kastomizirani detalj sigurnosti, inače je FALSE.

Function ToggleLayerOverrideHide(layerName As String) As Boolean

Preklapa (toggle) osobinu skrivanja nadjahivanja sloja (layer override hide) za specificirani sloj displeja.

Function ToggleRuntime() As Boolean

Preklapa tekući mod operatora izmedju konfiguracionog i runtime moda. Vraća TRUE ako metod stavlja GraphWorX32 u runtime mod, FALSE ako metod stavlja GraphWorX32 u konfiguracioni mod.

Function UngroupSymbol(objectName As String) As Boolean

Degrupira objekat **GwxSymbol** sa specificiranim imenom objekta. Mada funkcija uništava specificirani **GwxSymbol**, ne uništava objekte koji su grupirani u tom simbolu. Vraća TRUE kod uspjeha, FALSE kod neuspjeha. Raspoloživa je samo u konfiguracionom modu.

Sub ViewBoxZoom()

Ulazi u "Box Zoom" mod. Ovaj mod omogućava korisniku da razvuče (stretch) boks koji definira novu površinu gledanja (view). Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewFitToWindow()

Zumira tako da svi objekti u dispelju fituju u pogled (view). Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewHome()

Zumira na home pogled (home view). Home view je onaj pogled kod kojeg je posljednji put pohranjen displej. Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewPan(offsetX As Long, offsetY As Long)

Panira (skrolira) pogled sa specificiranim ofsetima. Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewShowWholeDisplay()

Zumira tako da dimenzije cjelokupnog displeja fituju u pogledu (view). Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewUnzoom()

Vraća unatrag od prethodne zum komande. Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewZoomCustomDialog()

Pokreće kastomizirani zum dijalog. Ovaj dijalog omogućava korisniku da unese novi procenat zumiranja. Nije raspoloživa u ActiveX ToolWorX-u.

Sub ViewZoomPercent(newZoom As Single)

Zumira unutra sa specificiranim procentom. Nije raspoloživa u ActiveX ToolWorX-u.

GwxVisible

GwxVisible je tip objekta iz kojeg se izvode vidljivi GraphWorX32 objekti (elipse, pravougaonici, itd.). Drugim riječima , svi vidljivi GraphWorX32 objekti imaju osobine i metode od GwxVisible.

Osobine (properties)

Angle As Single

Dobija ili setuje ugao rotacije objekta. Ugao se specificira u stepenima.

edgeStyle As GWX3DEDGESTYLE

3-D stil ivice od granice objekta. Validne vrijednosti za GWX3DEDGESTYLE su :

EdgeNone	= 0
EdgeRaised	= 5
EdgeEtched	= 6
EdgeBump	= 9
EdgeSunken	= 10

fillColor As OLE_COLOR

Dobija ili setuje boju ispune vidljivog objekta

hasShadow As Boolean

TRUE ako objekat ima sjenu, FALSE ako objekat nema sjene.

isFilled As Boolean

TRUE ako je objekat ispunjen, FALSE ako objekat nije ispunjen.

Keyword As String

Ovaj string se koristi da pohrani ključnu riječ (keyword). Koristiti ovu osobinu da se identificira objekat za funkcionalnost ažuriranja djeljenih objekata GraphWorX32

lineColor As OLE_COLOR

Dobija ili setuje boju linije ili granice vidljivog objekta.

lineStyle As GWXLINESTYLE

Stil linije ili granice objekta. Validne vrijednosti za **GWXLINESTYLE** su :

LineSolid	= 0
LineDash	= 1
LineDot	= 2
LineDashDot	= 3
LineDashDotDot	= 4
LineNone	= 5

lineWidth As Long

Širina linije ili granice objekta. Ova vrijednost mora biti u opsegu 0 do 10.

objectName As String

Dobija ili setuje ime objekta kao vidljiv objekat. Ime objekta se koristi da identificira objekat kada se koriste izvjesne metode OLE Automation (napr. **Gwxdisplay::GetVisibleObjectFromName**).GraphWorrX32 će osigurati da su imena objektata jedinstvena. Ako doznačimo ime objekta koje već postoji za neki drugi vidljivi objekat u displeju, GraphWorX32 će dodati broj na ime objekta (napr. "tank" će postati "tank1", "tank1" će postati "tank2" , itd).

Selected As Boolean

TRUE ako je objekat izabran, FALSE ako objekat nije izabran. Izabrani objekti su objekti koji imaju gripove za promjenu veličine (resize) oko njih . Ova osobina se može primjeniti samo u konfiguracionom modu.

shadowColor As OLE_COLOR

Setuje ili dobija boju sjenke (shadow) od vidljivog objekta.

UserCustomData As String

Ovaj string se koristi da pohrani kastomizirane podatke. Koristiti ovu osobinu da se pridruže dodatni podatci sa vidljivim objektom.

UserDescription As String

Opisni string za vidljivi objekat. Tipično, ovaj string se koristi da se pokaže kao informacioni tekst u tooltipu.

Visible As Boolean

TRUE ako je objekat vidljiv, FALSE ako je objekat sakriven.

METODE

Sub ClearGradientFill()

Čisti (clear) prethodno postavljenu gradijentnu ispunu (fill) za vidljivi objekat.

FlipHorizontal()

Omogućava flipovanje (preokretanje) oko horizontalne ose. Efekat je isti kao da se zrcali u horizontalnom ogledalu.

FlipVertical()

Omogućava flipovanje (preokretanje) oko vertikalne ose. Efekat je isti ka da se zrcali u vertikalnom ogledalu.

Primjer koda

```
Dim obj as GWXRectangle  
set obj=ThisDisplay.GetVisibleObjectFromName("myobject")  
obj.FlipHorizontal  
obj.FlipVertical
```

Function GetDataSources(includeDuplicates As Boolean)

Dobija sve tagove pridružene sa vidljivim objektom. Vraća (varriant) polje stringova (array of strings) – svaki string u polju je ime taga. Polje može opcionalno uključiti ili ne uključiti dupla imena tagova.

Function GetDynamicObjectFromIndex(index As Long) As Object

Dobija dinamički objekat sa specificiranim nula-baziranim indeksom, koji je priključen na vidljivi objekat (indeks nula je prvi koji je priključen na dinamiku). Ova funkcija je korisna za iteraciju kroz sve dinamičke objekte priključene na vidljivi objekat.

Function GetDynamicObjectFromName(nameSubstring As String) As Object

Dobija dinamički objekat sa specififiranim imenom (ili djelom imena)., koji je priključen na vidljivi objekat.

Function GetNumberOfDynamics() As Long

Vraća broj dinamičkih objekata koji su priključeni na ovaj vidljivi objekat.

Sub GetObjectDimensions(left As Single, top As Single, width As Single, height As Single)

Dobija veličinu i lokaciju objekta. Ovaj metod ne radi u VBScriptu i Jscriptu.

Sub GetObjectDimensions2(left, top, width, height)

Ovaj metod radi u Visual Basicu za aplikacije kao i u VBScript i JScript. Da bi radio i u VBA , treba deklarirati parametre kao Variant:

```
'VB example, works only in runtime mode
'example for getting object dimensions
(GetObjectDimensions2 returns variants)
'visible is GwxVisible object
'example requires an object named "test"

Dim l, t, w, h, visible
Set visible = ThisDisplay.GetVisibleObjectFromName("test")
visible.GetObjectDimensions2 l, t, w, h
MsgBox "Left: " & l & vbCrLf & "Top: " & t & _
      vbCrLf & "Width: " & w & _
      vbCrLf & "Height: " & h
```

Function GetObjectTypeName() As String

Vraća ime tipa ovog objekta.

Function GetParentObject() As Object

Ovaj metod dozvoljava korisniku da dobije referentnu tačku ka roditeljskom (parent) objektu u hijerariji GraphWorX32 objekata. U narednom primjeru , pretpostavimo da imamo dva objekta koja se zovu "child1" i "child2" grupirana zajedno u simbolu koji se zove "parent". Startujući od poentera na child1, dobićemo pointer na parent a zatim displej je ime.

```

Dim obj As GwxRectangle
Set obj = ThisDisplay.GetVisibleObjectFromName("child1")
If obj Is Nothing Then
    MsgBox "can't find child1"
Else
    Set parent = obj.GetParentObject
    If parent Is Nothing Then
        MsgBox "can't get parent"
    Else
        MsgBox parent.objectName
    End If
End If

```

Sub MoveObject(offsetX As Single, offsetY As Single)

Pomjera objekat sa specificiranim ofsetom. Ovaj metod radi samo za objekte koji su vidljivi objekti najvišeg nivoa.

Sub RefreshObject()

Prouzrokuje da vidljivi objekat bude ponovno iscrtan.

Sub Rotate90(clockwise As Boolean)

Ovaj metod rotira objekte za 90 stepeni. Uzima jedan Boolov parametar kao ulaz da se postavi da li je rotacija u smjeru kazaljke na satu ili nije. Posmatrajmo naredni jednostavni primjer kako da rotiramo pravougaonik za 90 stepeni u smjeru kazaljke na satu.

```

Dim rect As GwxRectangle
Set rect = ThisDisplay.GetVisibleObjectFromName("rectangle")
If rect Is Nothing Then
    MsgBox "An error has occurred while trying to get the object"
Else
    rect.Rotate90 True
End If

```

Function SetAliasDefinition(aliasName As String, newDefinition As String) As Boolean

Setuje alias definiciju za sve dinamičke objekte pridružene sa ovim vidljivim objektom. Ako ovaj GwxVisible je također GwxSymbol, ova funkcija se rekurzivno primjenjuje na sve objekte grupirane unutar simbola. Vraća FALSE ako uparujuća imena nisu nadjena, inače TRUE.

Function SetAliases(commandString As String) As Boolean

Setuje višestruke alias definicije za ovaj vidljivi objekat. Ako ovaj GwxVisible je takodjer GwxSymbol, ova funkcija se rekurzivno primjenjuje na sve objekte grupirane unutar simbola.

Function SetAliasesFromFile(filename As String) As Boolean

Setuje višestruke alias definicije za ovaj vidljivi objekat. Ako ovaj GwxVisible je takodjer GwxSymbol, ova funkcija se rekurzivno primjenjuje na sve objekte grupirane unutar simbola.

Function SetGradientFill(style As GWXGRADIENTSTYLE, color1 As OLE_COLOR, color2 As OLE_COLOR, isTwoColor As Boolean, brightness As Single, reverseColors As Boolean, steps As Integer, horizontalOffset As Single, verticalOffset As Single) As Boolean

Setuje gradijent ispune za ovaj vidljivi objekat.

Parametar style može imati jednu od slijedećih vrijednosti:

GradientHorizontal	= 0
GradientVertical	= 1
GradientSquare	= 2

Ostatak parametara korespondira sa onima u korisničkom interfejsu za konfigurisanje gradijentne ispune (gradient fill).

Sub SetObjectDimensions(left As Single, top As Single, width As Single, height As Single)

Setuje veličinu i lokaciju objekta. Ovaj metod radi samo za objekte koji su najgornji nivo vidljivih objekata.

Sub StretchObject(scaleX As Single, scaleY As Single, anchorX As Single, anchorY As Single)

Razvlači (stretch) objekat na bazi specificiranih faktora skale. Parametri scaleX i scaleY su vrijednosti tako da 0.0 je 0% a 1.0 je 100%. Sidrišne (anchor) vrijednosti se koriste da promjene način kako objekti pomjeraju poziciju za vrijeme razvlačenja.

Na primjer, da se promjeni veličina objekta iz njegovog centra, sidrišne vrijednosti trebaju biti centralne koordinate objekta. Ovaj metod radi samo za objekte koji su najgornji nivo vidljivih objekata.

GwxText

GwxText ima sve osobine i metode od **GwxVisible**, plus dodatne osobine koje su opisane u nastavku:

Osobine

Font As IFontDisp

Dobija ili setuje font tekst objekta

alignment As GWXTEXTALIGNMENT

Poravnanje teksta (lijevo, centar, desno). Ova osobina je samo značajna za višelinijske (multiline) tekst stringove. Validne vrijednosti **GWXTEXTALIGNMENT** su:

TextAlignLeft	= 0
TextAlignCenter	= 1
TextAlignRight	= 2

BoundaryAlignment As GWXTEXTALIGNMENT

Poravnanje teksta relativno na pravougaonik u kojem je sadržan tekst . Validne vrijednosti **GWXTEXTALIGNMENT** su:

TextAlignLeft	= 0
TextAlignCenter	= 1
TextAlignRight	= 2

stretchText As Boolean

TRUE ako veličina fonta treba da se promjeni ako se tekst objekat rasteže. FALSE ako veličina fonta ostaje ista .

text As String

Ovaj string je tekst koji je prikazan sa tekst objektom.

GwxRectangle

GwxRectangle ima sve osobine i metode od **GwxVisible**, plus dodatne osobine koje su:

Osobine

rounded As Boolean

RoundingX As Long

RoundingY As Long

GwxArc

GwxArc ima sve osobine i metode od **GwxVisible**, plus dodatne osobine koje su:

Osobine

endAngle As Single

arcType As GwxARCTYPE

startAngle As Single

GwxSymbol

GwxSymbol se koristi da grupiše kolekciju izvedenih objekata iz **GwxVisible**. **GwxSymbol** objekti se mogu koristiti da grupišu druge **GwxSymbol** objekte, omogućavajući tako ugnježeno (nested) grupiranje. **GwxSymbol** ima sve osobine i metode od **GwxVisible**, plus dodatne metode koje su:

Metode

Function GetDataSourcesRecursive(includeDuplicates As Boolean)

Dobija sva imena tagova pridružena sa simbolom (uključujući i sub-objekte). Vraća (variant) polje stringova (svaki string u polju je ime taga). Polje može opciono uključiti ili ne uključiti dupla imena tagova.

Function GetNumberOfChildVisibles() As Long

Vraća broj trenutnih objekata djece grupiranih u ovom simbolu.

Function GetVisibleObjectFromIndex(index As Long) As Object

Dobija vidljivi objekat sa specificiranim nul-baziranim indeksom, koje je jedno trenutno dijete ovog simbola (indeks nula je prvi dijete objekat). Ova funkcija je korisna za iteraciju kroz sve neposredne djeca objekte grupirane u ovom simbolu.

Function GetVisibleObjectFromName(nameSubstring As String) As Object

Dobija vidljivi objekat sa specificiranim imenom (ili dijelom imena) , koje je neposredno dijete ovog simbola.

GetVisibleObjectFromPosition(x as long, y as long) as Object

Vraća objekat kod specificirane pozicije na displeju. Starta od korjena displeja i ide dublje uključujući slojeve, sve dok najviši objekat nije nadjen. Ako argument GoIntoSymbol je setovan na TRUE, traženje se nastavlja sve do konačnog vidljivog objekta na datoj poziciji. U protivnom, funkcija vraća simbol najgornjeg nivoa koji sadrži objekat. Funkcija simbola vraća vidljivi objekat unutar simbola kod specificirane pozicije:

```
'VBA example, works in configure and runtime mode
'this example gets the lowest level visible at random
position

Dim w As Long, h As Long
Dim o As Object
ThisDisplay.GetDisplayDimensions w, h
'gets display width and height
If ThisDisplay.GetNumberOfTopLevelVisibles > 0 Then
'checks if display has at least one visible object
'and runs a loop for find one
Do
  Set o = ThisDisplay.GetVisibleObjectFromPosition(Rnd * w,
  Rnd * h, True)
'tries to get visible object from random position
'- if object is GwxSymbol, it goes into symbol
'and continues searching
Loop Until Not (o Is Nothing)
'after any object is found, loop ends and example
'shows name of the object
MsgBox o.objectName
Else
  MsgBox "Example needs at least one visible object"
End If
```

Function PushCurrentSymbol() As Boolean

Stavlja (push) simbol objekta na stack editora simbola. Ovo u suštini će učiniti objekte grupirane u simbolu, kao vidljive objekte najgornjeg nivoa.

GwxOleObject

GwxOleObject ima sve osobine i metode od **GwxVisible**, plus dodatne metode koji su opisani u nastavku:

Metode

Function GetOLEObject() As Object

Dobija aktuelni OLE objekat omotan (wrapped) sa **GwxOleObject**. Ovaj metod omogućava pristup do uronjenog ActiveX objekta. Na primjer, pretpostavimo da **TrendWorX32 Viewer** objekat koji se zove "twx_01" je unesen u **GraphWorX32** displej. Tada je moguće da se koristi slijedeći kod da se pristupi ovoj ActiveX kontroli u runtime-u i radi sa njom:

```
Dim o As GwxOleObject
Set o = ThisDisplay.GetVisibleObjectFromName("twx_01")

Dim twx As TWXVIEWERLib.TWXViewer
Set twx = o.GetOLEObject()

twx.BackColor = RGB(0, 255, 0)
```

GwxEllipse

GwxEllipse ima sve osobine i metode od **GwxVisible**.

GwxLine

Gwxline ima sve osobine i metode od **GwxVisible**.

GwxBitmap

Gwxbitmap ima sve osobine i metode od **GwxVisible**.

GwxMetafile

Gwxmetafile ima sve osobine i metode od **GwxVisible**.

GwxButton

GwxButton objekat nije potpuno funkcionalan sve dok ima i jedan **Gwxpick** dinamički objekat pridružen sa njim. **GwxButton** ima sve osobine i metode od **GwxVisible** i **GwxText**.

GwxDynamic

GwxDynamic je tip objekta iz kojeg se izvode dinamički GraphWorX32 objekti (veličina , lokacija, itd). Drugim riječima , svi dinamički GraphWorX32 objekti imaju osobine i metode od **GwxDynamic**.

Osobine

dataSource As String

Primarni izvor podatka za dinamički objekat. **Data source** je string koji predstavlja OPC ime taga, izraz, konstantnu vrijednost, ili GraphWorX32 lokalnu varijablu.

highRange As String

Predstavlja nadjahani visoki opseg za ovu dinamiku. **HighRange** je string koji predstavlja OPC ime taga, izraz, konstantnu vrijednost, ili GraphWorX32 lokalnu varijablu. Ova osobina se koristi samo za dinamiku baziranu na analognom izvoru podatka, i ignorira se za dinamike koje su bazirane na digitalnim konekcijama.

lowRange As String

Predstavlja nadjahani niski opseg za ovu dinamiku. **LowRange** je string koji predstavlja OPC ime taga, izraz, konstantnu vrijednost, ili GraphWorX32 lokalnu varijablu. Ova osobina se koristi samo za dinamiku baziranu na analognom izvoru podatka, i ignorira se za dinamike koje su bazirane na digitalnim konekcijama.

objectName As String

Dobija ili setuje ime objekta od dinamičkog objekta. Ime objekta se koristi da identificira objekat kada se koriste izvjesne OLE Automation metode (napr. **GwxDisplay::GetDynamicObjectFromName**). GraphWorX32 će obezbjediti da su imena objekata jedinstvena.

RangeOverride As Boolean

Dobija ili setuje status nadjahivanja opsega. Kada je FALSE, GraphWorX32 će koristiti opsege pridružene sa primarnim **DataSource**. Kada je TRUE, GraphWorX32 će koristiti opsege definirane u **HighRange** i **LowRange** osobinama.

Ova osobina se samo koristi za dinamike bazirane na analognim izvorima podataka (**GwxSize**, **GwxLocation**, **GwxRotation**, **GwxAnalogColor**, **GwxAnalogSelector**, **GwxProcessPoint**), i ignorira se za dinamike koje su bazirane na digitalnim konekcijama.

requestDataType As GWXDATATYPE

Omogućava da se specificira tip podatka koji će se koristiti da se zahtjeva izvor podatka od OPC servera.

TimerRate As Long

Brzina ažuriranja frekvencije za tajmer bazirane dinamičke tipove (ova osobina je ignorisana za dinamike koje nisu bazirane na tajmeru). Dinamike na bazi tajmera uključuju: GwxFlash, GwxAnimator, i GwxPick.

UserCustomData As String

Ovaj string se koristi da pohrani kastomizirane podatke. Koristiti ovu osobinu da se pridruže bilo koji dodatni podatci sa dinamičkim objektom.

UserDescription As String

Opisni string za dinamički objekat. Tipično, ovaj string se koristi da se prikaže kao informacioni tekst u tooltipu.

Metode

Function GetAliasDefinition(aliasName As String) As String

Dobija alias definiciju od specificiranog alias imena.

Function GetDataSourcePointObject() As Object

Dobija objekat **GwxPoint** za izvor podatka ovog dinamičkog objekta.

Function GetHighRangePointObject() As Object

Dobija **GwxPoint** za gornji opseg ovog dinamičkog objekta.

Function GetLowRangePointObject() As Object

Dobija **GwxPoint** za donji opseg ovog dinamičkog objekta.

Function GetNumberOfSubDynamics() As Long

Vraća broj trenutnih dijete sub-dinamičkih objekata korištenih u ovom dinamičkom objektu. Sub-dinamički objekti su korišteni od dinamičkih tipova: **GwxDigitalColor**, **GwxDigitalSelector**, i **GwxPick**.

Function GetObjectName() As String

Vraća ime tipa ovog objekta.

Function GetSubDynamicObjectFromIndex(index As Long) As Object

Dobija sub-dinamički objekat sa specificiranim nula-baziranim indeksom, koji je jedno neposredno dijete (child) ovog dinamičkog objekta. (indeks nula je prvo dijete objekat).

Function GetVisibleObject() As Object

Dobija vidljivi objekat na kojeg je ova dinamika priključena.

Function SetAliasDefinition(aliasName As String, newDefinition As String) As Boolean

Setuje alias definiciju samo za ovaj dinamički objekat. Vraća FALSE ako uparivanje alias imena nije nadjeno, inače TRUE.

Function SetAliases(commandString As String) As Boolean

Setuje višestruke alias definicije za ovaj dinamički objekat.

Function SetAliasesFromFile(filename As String) As Boolean

Setuje višestruke alias definicije za ovaj dinamički objekat.

GwxDigitalSelector

Objekat **GwxDigitalSelector** se koristi da se sakriju/pokažu individualni objekti iz seta objekata baziranih na digitalnom (Boolovom) signalu. **GwxDigitalSelector** objekti su udruženi samo sa **GwxSymbol** objektima (GwxSymbol objekat definira set objekata nad kojima objekat **GwxDigitalSelector** djeluje).

GwxDigitalSelector ima sve osobine i metode od GwxDynamic, plus dodatne metode opisane u nastavku.

Metode

Function SetConnectionInfo(objectNumber As Integer, dataSource As String, showWhenTrue As Boolean) As Boolean

Setuje konekcije sa podacima za ovaj digitalni objekat. Parametar **objectNumber** je nul-bazirani indeks koji predstavlja trenutnu dijecu ovog simbol objekta, na koji je ova dinamika priključena. (indeks nula je prvi dijete objekat).

GwxDigitalSelectorInfo

Ovaj objekat uključuje osobine individualnih konekcija podataka za **GwxDigitalSelector** objekte. **GwxDigitalSelectorInfo** ima sve osobine i metode **GwxDynamic**.

GwxDigitalColor

Objekat **GwxDigitalColor** se koristi da promjeni boju pridruženog vidljivog objekta baziranog na jednom ili više digitalnih signala. **GwxDigitalColor** ima sve osobine i metode **GwxDynamic**, plus dodatne metode opisane u nastavku.

Metodi

Function DeleteSubDynamic(index As Integer) As Boolean

Ova funkcija briše subdinamički objekat na poziciji indeksa. Vraća true kod uspjeha. Vraća false ako ne uspije, tipično kada indeks je van opsega.

Function SwapSubDynamics(index1 As Integer, index2 As Integer) As Boolean

Ova funkcija zamjenjuje (swap) subdinamičke objekte na pozicijama index1 i index2. Vraća true kod uspjeha. Vraća false kod neuspjeha, tipično kada indeks je van opsega.

GwxDigitalColorInfo

Objekat **GwxDigitalColorInfo** uključuje osobine od individualnih konekcija podataka za **GwxDigitalColor** objekte. **GwxDigitalColorInfo** ima sve osobine i metode **GwxDynamic**, plus dodatne osobine opisane u nastavku:

Osobine

UseFillColor As Boolean

UseLineColor As Boolean

UseShadowColor As Boolean

Ove osobine omogućavaju promjenu boja na visuelnom objektu.

FillColor As OLE_COLOR

LineColor As OLE_COLOR

ShadowColor As OLE_COLOR

Ove osobine definiraju boje za akciju boje.

ChangeColorOnTrue As Boolean

Ova osobina specificira da li je akcija aktivirana kada se evaluira OPC tag kao true ili false.

GwxAnalogSelector

Objekat **GwxAnalogSelector** se koristi da sakrije /pokaže individualne objekte iz seta objekata na bazi analognog signala. **GwxAnalogSelector** objekti sa samo udruženi sa **GwxSymbol** objektima (**GwxSymbol** objekat definira set objekata nad kojima djeluje **GwxAnalogSelector** objekat). **GwxAnalogSelector** ima sva osobine i metode **GwxDynamic** , plus dodatni metod opisan u nastavku.

Metode

Function SetSelectorRanges(rangeArray) As Boolean

Setuje opsege selektora za ovu dinamiku. Vraća TRUE za uspjeh, FALSE za neuspjeh.

Parametar rangeArray je **SAFEARRAY** od varijabli sa pokretnim zarezom (floats). Broj elemenata u polju mora biti jednak broju objekata u analognom selektoru. Svaki indeks u polju korespondira indeksu objekta u analognom selektoru. Svaki element polja predstavlja procentualnu vrijednost za gornju granicu opsega tog objekta. Svaki element polja mora biti vrijednost izmedju 0 i 1. Svaki element polja mora biti od prethodne vrijednosti elementa. Posljednji element polja mora uvijek biti 1.

GwxAnalogColor

Objekat **GwxAnalogColor** se koristi da promjeni boju pridruženog vidljivog objekta baziranog na analognom signalu. **GwxAnalogColor** ima sve osobine i metode **GwxDynamic**.

GwxAnimator

Objekat **GwxAnimator** se koristi da sakrije/pokaže individualne objekte iz seta objekata u specificiranoj sekvenci, na bazi isteklog vremena(elapsed time). **GwxAnimator** je aktiviran/deaktiviran na bazi vrijednosti digitalnog signala. Objekti **GwxAnimator** su samo udruženi sa **GwxSymbol** objektima (**GwxSymbol** objekat definira set objekata

nad kojima djeluje **GwxAnimator** objekt.). **GwxAnimator** ima sve osobine i metode **GwxDynamic**.

GwxSize

Objekat **GwxSize** se koristi da promjeni veličinu pridruženog vidljivog objekta na bazi vrijednosti analognog signala. **GwxSize** ima sve osobine i metode **GwxDynamic**.

GwxLocation

Objekat **GwxLocation** se koristi da promjeni lokaciju pridruženog vidljivog objekta na bazi vrijednosti analognog signala. **GwxLocation** ima sve osobine i metode **GwxDynamic**, plus dodatne metode opisane u nastavku.

Metode

Sub GetCoordinateFromLocationDelta(offset As Single, cx, cy)

Ovo je napredna funkcionalnost dodata za naprednije korisnike koji žele da razviju kompleksne aplikacije. Opšte aplikacije neće naći korist od ove funkcije.

Function GetLastEnteredValue()

Vraća posljednju unesenu vrijednost, koja može biti različita od tekuće prikazane vrijednosti.

GwxRotation

Objekat **GwxRotation** se koristi da promjeni ugao pridruženog vidljivog objekta na bazi vrijednosti analognog signala. **GwxRotation** ima sve osobine i metode od **GwxDynamic**, plus dodatne metode opisane u nastavku.

Metode

Function GetLastEnteredValue()

Vraća posljednju unesenu vrijednost, koja može biti različita od tekuće prikazane vrijednosti.

GwxFlash

Objekat **GwxFlash** se koristi da preklopi (toggle) boju ili vidljivost pridruženog vidljivog objekta sa specificiranom brzinom. Objekat **GwxFlash** se aktivira/deaktivira na bazi vrijednosti digitalnog signala. **GwxFlash** ima sve osobine i metode **GwxDynamic**,

GwxHide

Objekat **GwxHide** se koristi da preklopi (toggle) vidljivost ili omogućeno/onemogućeno stanje pridruženog vidljivog objekta na bazi digitalnog signala. **GwxHide** ima sve osobine i metode **GwxDynamic**,

GwxPick

Udruživanjem GwxPick objekta sa vidljivim objektom označava da će pridruženi vidljivi objekat izvršavati aktivnost kada se klikne na njega od strane operatora za vrijeme runtime-a. GwxPick objekti se mogu udružiti sa GwxButton objektima da se naprave push tasteri pune funkcionalnosti. GwxPick ima sve osobine i metode **GwxDynamic**,

Osobine

aliasCommandString As String

Ako je osobina **AliasCommandType** jednaka **AliasSetFromFile**, **AliasCommandString** treba biti ime fajla koje specificira tekst fajl koji sadrži parove alias imena i alias definicija separirane sa tabom. Ako **AliasCommandType** je **AliasSetDirect**, **AliasCommandString** treba imati slijedeći format:

```
aliasName1<TAB>aliasDef1<CR-LF>aliasName2<TAB>aliasDef2<CR-LF> ...  
etc.
```

aliasCommandType As GwxAliasCommand

AliasCommandType može biti jedna od slijedećih vrijednosti:

AliasNoCommand	= 0
AliasSetFromFile	= 1
AliasSetDirect	= 2

Nameparametar As String

Značenje imena parametra varira na bazi vrijednosti od PickAction osobine.

Ako **PickAction** je jednaka **PickLoadDisplay**, **PickDragDropLoad**, **PickPopupWindow**, ili **PickEmbeddedWindow**, **NameParameter** je ime fajla Gwx32 displeja (*.gdf).

Ako je PickAction jednako **PickLaunchApp**, **NameParameter** je jedno izvršivo ime fajla (*.exe).

Ako je **PickAction** jednako **PickRunScript** , **NameParameter** je VBA makro ime.

Ako **PickAction** je **PickSetAliases** ili **PickAliasesDialog**, **NameParameter** je ime vidljivog objekta.

ScriptText As String

GraphWorX32 u verziji 7.1 omogućava udruživanje VBScript i Jscript sa pick akcijom. Skripte udružene sa pick akcijom se izvršavaju kada pick događaj je trigerovan (naprimjer, kada se pritisne lijevi taster miša). Ove skripte se mogu editirati putem novo integriranog skript toolbara, ili putem OLE Automation interfejsa. Svaka skripta se može pisati koristeći VBScript ili Jscript jezik nezavisno od drugih skripti pridruženih sa drugim pick akcijama. Ova osobina omogućava čitanje i pisanje izvornog koda skripte, trenutno pridruženog sa pick akcijom.

ScriptType As GWXSCRIPTTYPE

GraphWorX32 u verziji 7.1 omogućava udruživanje VBScript i Jscript sa pick akcijom. Ove skripte udružene sa pick akcijom se izvršavaju kada se pick događaj trigeruje (naprimjer, kada se pritisne lijevi taster miša). Ove skripte se mogu editirati putem novo integriranog skript toolbara, ili putem OLE Automation interfejsa. Svaka skripta se može pisati koristeći VBScript ili Jscript jezik nezavisno od drugih skripti pridruženih sa drugim pick akcijama. Ova osobina omogućava dobijanje i postavljanje skript jezika , trenutno pridruženog sa pick akcijom.

```
'VBA example, works only in configure mode
'example for creating pick action PickRunScript and set
script text and type

Dim o As Object
'creates rectangle

Set o = ThisDisplay.CreateRectangle( _
    50, 50, 100, 100, True, 100, 0, 2, _
    LineSolid, False, 0, EdgeNone, False, _
    "rectangle", False)
'adds pick action PickRunScript on the rectangle
Set o = ThisDisplay.CreatePickDynamic( _
    "rectangle", "dynamic", PickRunScript, _
    ButtonNormal, TriggerOnDown, MouseButtonLeft, _
    False, "", "", False, False, "", "", "")
'sets script properties - name of the script is
'derived from pick action name
o.ScriptType = ScriptTypeVBScript
o.ScriptText = "MsgBox " & Chr(34) & "Test OK" & Chr(34)
```

tracking As Boolean

Ako pick dinamika je ona tipa preključivanja (toggle) ili download pick i ček boks ili radio dugme tip pick-a, setovanje trackinga (praćenja) na TRUE će prouzrokovati da stanje tastera prati vrijednosti podataka pridruženih sa pick dinamikom.

Metode

Sub SimulateClick(mouseButton As GWXMOUSEBUTTON)

Prouzrokuje ovu pick dinamiku da se ponaša kao da je korisnik kliknuo na ovaj pick sa specificiranim tasterom miša.

GwxProcessPoint

Objekti **GwxProcessPoint** su udruženi sa **GwxText** objektima da kreiraju polja alfanumeričkih displeja i polja unosa podataka. Procesna tačka se koristi da pokaže i/ili downloaduje vrijednost pridruženog signala (analognog, digitalnog, ili stringa). **GwxProcessPoint** ima sve osobine i metode **GwxDynamic**.

Osobine

ConfirmBeforeWrite As Boolean

TRUE da promptira korisnika sa konfirmacionom porukom prije upisivanja vrijednosti podataka.

UseKeypad As Boolean

TRUE da se koristi onscreen numerički pad ili tastatura za unos podataka.

Metode

Function GetLastEnteredValue()

Vraća posljednju unesenu vrijednost, koja može biti različita od tekuće prikazane vrijednosti

GwxTimeDate

Objekti **GwxTimeDate** su odruženi sa **GwxText** objektima da kreiraju polja koja prikazuju tekuće vrijeme i/ili datum. **GwxTimeDate** ima sve osobine i metode **GwxDynamic**.

GwxPoint

GwxPoint je tip objekta koji GraphWorX32 koristi da predstavi konekcije sa podacima. Ako nekoliko **GwxDynamic** objekata su spojeni na isti izvor podatka, oni referenciraju jedan dijeljeni **GwxPoint** objekat. **GwxPoint** objekti manipuliraju sa OPC tagovima, izrazima, konstantnim vrijednostima, i GraphWorX32 lokalnim varijablama. **GwxPoint** objekti ne mogu biti eksplicitno kreirani ili uništeni. GraphWorX32 automatski upravlja sa životnim dobima (lifetime) od **GwxPoint** objekata na bazi konekcija izvora podataka od **GwxDynamic** objekata u displeju.

Osobine

dataType As GWXDATATYPE

Tip podatka ovog point objekta. Validne vrijednosti za **GWXDATATYPE** su :

DataTypeShort	= 2
DataTypeLong	= 3
DataTypeFloat	= 4
DataTypeDouble	= 5
DataTypeString	= 8
DataTypeBool	= 11
DataTypeByte	= 17

highRange As Variant

Gornji opseg vrijednosti pridružene sa ovom tačkom- objektom.

lowRange As Variant

Donji opseg vrijednosti pridružene sa ovom tačkom- objektom.

Value As Variant

Tekuća vrijednost podatka ove tačke objekta. Ova osobina se ažurira sa novim vrijednostima za vrijeme runtime moda.

Metode

Function GetObjectName() As String

Vraća ime tipa ovog objekta

Function GetPointName() As String

Dobija ime tačke (stringa izvora podatka) ove tačke objekta.

GraphWorX32 VBA projekat

GraphWorX32 VBA projekat se loaduje kada god startamo VBA Editor iz GraphWorX32 aplikacije pritiskom na ALT+F11. Ovaj projekat sadrži grupe modula po defaultu, kao:

- GraphWorX32 izvorne (native) objekte
- Module – ThisDisplay modul i GwxTools modul
- Forme

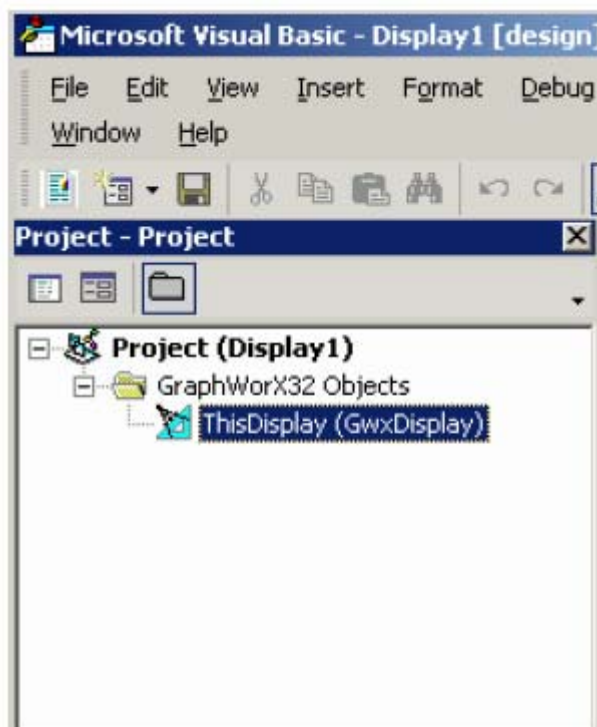
Svaki modul može sadržavati VBA kod : – funkcije, subrutine, handlerere događaja (event handler), i globalne deklaracije.

GwxTools Module

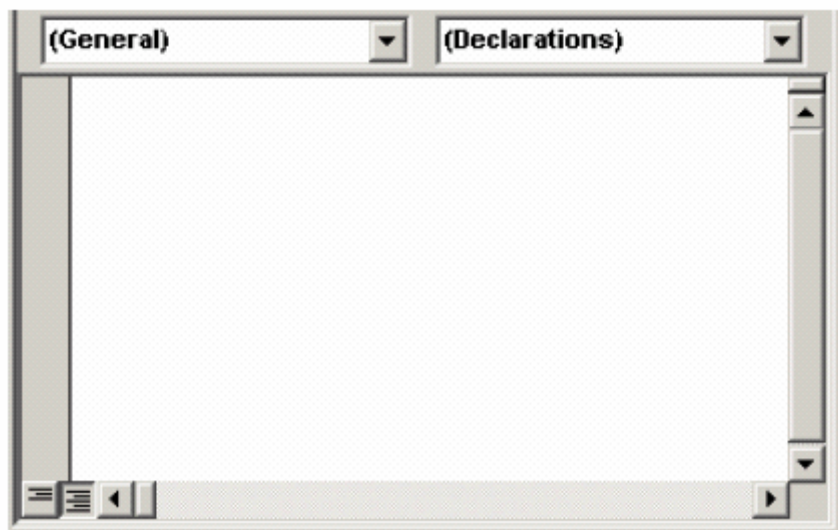
GwxTools je kastomizirani modul sa zajedničkim subrutinama koje se koriste u VBA wizardu. GwxTools modul nije raspoloživ za korisnika sve dok se ne koristi VBA Wizard.

Modul ThisDisplay

ThisDisplay je specijalni modul koji predstavlja tekući GraphWorX32 displej.



Da bi se prikazao kod za ThisDisplay modul, treba otvoriti ProjectExplorer iz View menija od VBA Editora. Ekspandirati GraphWorX32 grupu i dvaput kliknuti na ThisDisplay detalj (GwxDisplay). Otvoriće se ThisDisplay prozor sa kodom koji je prikazan na narednoj slici:



ThisDisplay sadrži dva kombo boksa na vrhu.

1. Gornji lijevi kombo boks omogućava nam da izaberemo detalje kao:

- Opšti (general)
- **GwxDisplay**

Prazan displej ima samo gornja dva detalja. Ako imamo ActiveX kontrole u displeju, ActiveX detalji će također biti pokazani u ovom kombo boks.

2. Gornji desni kombo boks pokazuje događaje za lijevu kombo selekciju. Naprimjer, **GwxDisplay** detalj ime događaje kao što su **DisplayLoad**, **DisplayUnload**, i druge.

Manipuliranje događajima sa ThisDisplay modulom

Handleri događaja se mogu dodati na displej koristeći VBA. Da bi se ovo realizovalo, treba imati displej koji je loadovan na ekran.

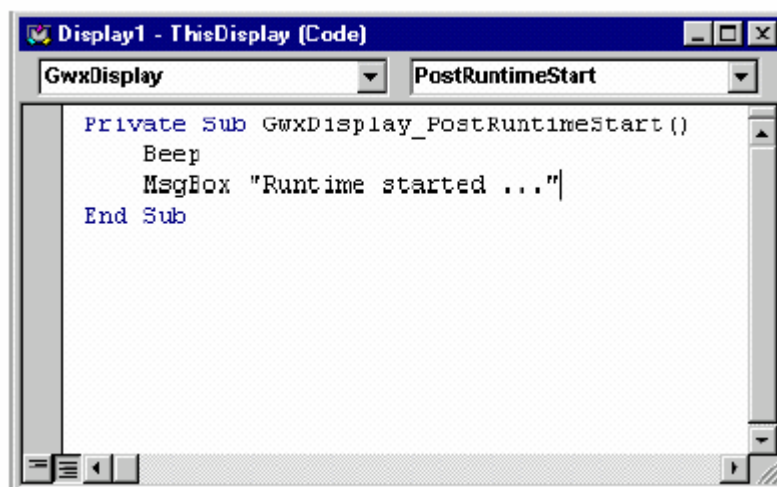
1. Otvoriti VBA Editor selektirajući **Macros>Visual Basic Editor** iz GraphWorX32 Tools menija. GraphWorX32 VBA projekat će se loadovati po defaultu.
2. Otvoriti **ThisDisplay** modul iz VBA Editora.
3. Izabrati **GwxDisplay** modul u lijevom kombo boks **ThisDisplay** modula.
4. Izabrati željeni događaj iz liste u desnom kombo boks.
5. Zaglavlje subrutine se automatski dodaje u tekući modul. Unjeti VBA kod u tijelo subrutine.
6. Pohraniti GraphWorX32 projekat izabirući **Save Display1** iz **File** menija.

7. Izaći iz VBA Editora izabirući **Close and Return to GraphWorX32** iz **File** menija.

Primjer

Napisati kod za iskakanje (pop-up) boks poruke koji će prikazati poruku "Runtime started", kada postavimo displej u runtime.

1. Izabrati GwxDisplay modul.
2. Izabrati PostRuntimeStart događaj iz liste u desnom kombo boksu.
3. Umetnuti VBA kod, kako je pokazano na narednoj slici, u tijelo subrutine u zaglavlju subrutine:



```
Display1 - ThisDisplay (Code)
GwxDisplay PostRuntimeStart
Private Sub GwxDisplay_PostRuntimeStart ()
    Beep
    MsgBox "Runtime started ...!"
End Sub
```

4. Preključiti u GraphWorX32 bilo zatvarenjem VBA Editora ili pritišćući na **ALT+TAB** tastere.
5. Testirati događaj startujuć runtime klikanjem na **Runtime** na GraphWorX32 meni baru ili pritišćući na **CTR+M** tastere.
6. Treba da čujemo zvuk (beep) i da se pojavi boks poruke sa porukom **"Runtime started"**.

Kako kontrolirati GraphWorX32 Automation pomoću VBA

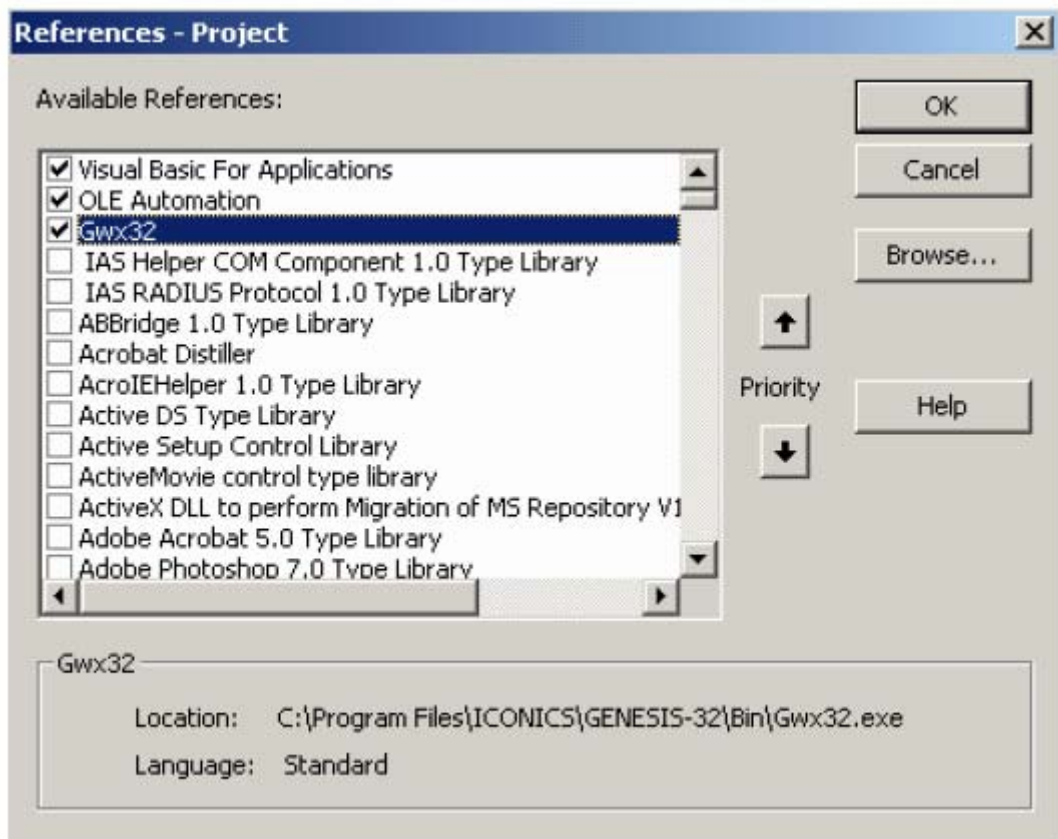
VBA programer ima punu kontrolu nad osobinama i izvornim objektima tekućeg displeja i može kontrolirati bogati animacioni interfejs od GraphWorX32.

Automation osobine i metode su pristupačne putem ThisDisplay modula. ThisDisplay , je predstava tekućeg GraphWorX32 displeja i sadrži sve Automation osobine i metode.

Opaska:

VBA može prikazati Automation metode i osobine samo ako je Automation objekat čekiran u **References** dijalogu, koji se može otvoriti iz meni bara. Izabrati **References**

iz **Tools** menija VBA Editora. Primjetimo da GraphWorX32 je čekiran automatski za svaki novi displej, kako je pokazano na donjoj slici:



Kako pristupiti izvornim displej objektima GraphWorX32

GraphWorX32 izvorni objekti, kao elipse, pravougaonici, simboli i dinamičke akcije, su izloženi VBA po defaultu. Međutim, oni mogu da se referenciraju iz VBA ako imaju Object Name doznačen putem dijaloga Property inspector.

Referenca (ustvari dispeč poenter) na imenovani objekat se može potpuno dobiti pomoću jednog od slijedećih metoda od **ThisDisplay**, **symbol**, i **Visible** objekata.

- **ThisDisplay.GetDynamicObjectFromName**
- **ThisDisplay.GetDynamicVisibleObjectFromIndex**
- **ThisDisplay.GetVisibleObjectFromName**
- **Visible.GetDynamicObjectFromIndex**
- **Visible.GetDynamicObjectFromName**
- **symbol.GetVisibleObjectFromIndex**
- **symbol.GetVisibleObjectFromName**

ThisDisplay.GetVisibleObject FromName ("Name").

Ova funkcija uzima ime vizuelnog objekta u displeju i izvještava objekat pridružen sa ovim imenom. Nakon pohranjivanja rezultata ove funkcije u varijablu, korisnik može

promjeniti osobine objekta u displeju promjenom polja varijable. Slijedeći primjer rotira pravougaonik sa imenom "Square" za 45 stepeni od horizontale.

```
Dim obj As GwxRectangle
Set obj = ThisDisplay.GetVisibleObjectFromName("Square")
obj.Angle = 45
```

ThisDisplay.GetDynamicObjectFromName("Name").

Ova funkcija uzima ime dinamičkog objekta u displeju i izvještava dinamički objekat pridružen sa tim imenom. Dinamički objekat nije fizički, observabilni objekat nego abstrakcija za operaciju izvršenu nad vidljivim objektom(tj. Hide, Rotation, Size). Kada se dinamički objekat pohrani kao varijabla, korisnik može promjeniti njene osobine. Slijedeći primjer mjenja izvor podatka za GraphWorX32 Hide objekat na nulu, sakrivajući vidljivi objekat pridružen sa dinamičkim objektom.

```
Dim obj As GwxHide
Set obj= ThisDisplay.GetDynamicObjectFromName("hd")
obj.dataSource = 0
```

ThisDisplay.GetVisibleObjectFromIndex(Long Value).

Ova funkcija izabira objekat baziran na redoslijedu u kojem su bili kreirani vidljivi objekti na ekranu.(prvi vidljivi objekat koji se stavlja na ekran ima indeks 0). Ova funkcija je korisna za iteraciju kroz sve objekte u displeju. Slijedeći kod će promjeniti prvi kreirani objekat u zeleno a drugi u crveno.

```
Dim obj1 As Object
Dim obj2 As Object
Set obj1 = ThisDisplay.GetVisibleObjectFromIndex(0)
Set obj2 = ThisDisplay.GetVisibleObjectFromIndex(1)
obj1.fillColor = RGB(0, 255, 0)
obj2.fillColor = RGB(255, 0, 0)
```

Symbolname.GetVisibleObjectFromIndex(Long value)

Svaki GraphWorX32 simbol ima svoj vlastiti indeks koji drži zapis o objektima unutar njega. Metod GetVisibleObjectFromIndex, kada se doda uz ime simbola, nalazi vidljivi objekat unutar simbola sa specificiranim indeksom. Ova funkcija je korisna za iteraciju kroz sve objekte u simbolu. Slijedeći primjer prebacije treći vidljivi objekat u simbolu sa imenom "sym" u zeleno.

```
Dim sym1 As GwxSymbol
Dim obj As Object
Set sym1 = GetVisibleObjectFromName("sym")
Set obj = sym1.GetVisibleObjectFromIndex(2)
obj.fillColor = RGB(0,255,0)
```

(Primjetimo da treći objekat ima indeks 2. Prvi ima indeks 0).

SymbolName.GetVisibleObjectFromName("Name")

Nalazi vidljivi objekat unutar simbola pomoću imena objekta specificiranog u GraphWorX32.

VisibleObjectName.GetDynamicObjectFromIndex(Long value)

Svaki put kada je dinamički objekat doznačen na jedinstveni vidljivi objekat, njemu je doznačen i indeks. Prvi dinamički objekat koji je doznačen dobija indeks nula. Pošto jedan vidljivi objekat može biti udružen sa mnogim dinamičkim objektima, ova funkcija obezbjeđuje koristan način manipuliranja sa dinamičkim objektima. Slijedeći kod u primjeru uzima drugi dinamički objekat udružen sa pravougaonikom sa imenom "rect" i mijenja njegov donji opseg na 10.

```
Dim o_Vis As GwxRectangle  
Dim o_Dyn As Object  
Set o_Vis = ThisDisplay.GetVisibleObjectFromName("rect")  
Set o_Dyn = o_Vis.GetDynamicObjectFromIndex(1)  
o_dyn.lowRange = 10
```

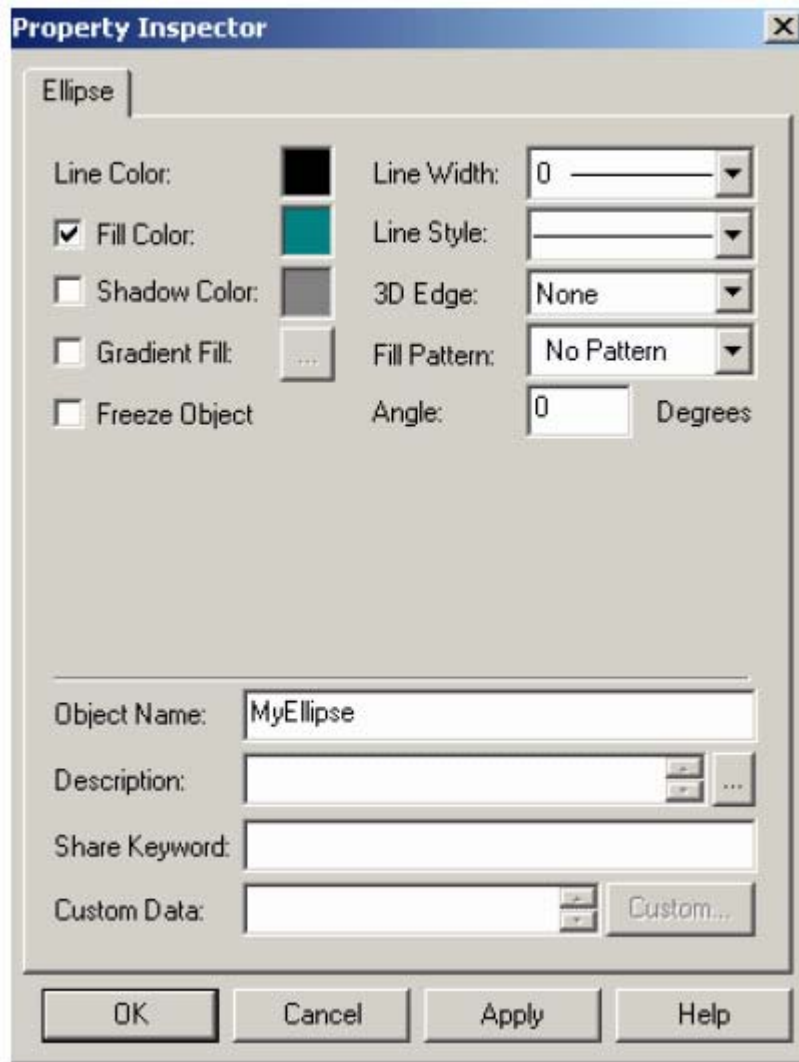
VisibleObjectName.GetDynamicObjectFromName("Name")

Ova funkcija uzima ime od dinamičkog objekta udruženog sa vidljivim objektom i omogućava korisniku da je predstavi dinamičkom objektu sa varijablom.

Primjer

Napisati kod za promjenu boje elipse GraphWorX32 izvornog objekta za vrijeme runtime-a:

1. Loadovati postojeći displej ili kreirati novi GraphWorX32 displej.
2. Nacrtať elipsu klikanjem na **Ellipse** ikonu na **Draw** toolbaru.
3. Dvput kliknuti na elipsu. Otvara se dijalog boks **Proprety inspector** kako je pokazano na narednoj slici. Unjeti ime **MyEllipse** u polje ime objekta i zatim klinuti na **OK**.
4. Otvoriti VBA Editor pritišćući na **Alt+F11**.



5. Izabrati **ThisDisplay** modul
6. Izabrati **GwxDisplay** modul u lijevom kombo boksu.
7. Izabrati **PostRuntimeStart** događaj iz liste u desnom kombo boksu.
8. Zaglavlje subrutine se automatski umeće u tekući modul. Unjeti VBA kod u tijelo subrutine, kako je pokazano na slijedećoj slici:


```
Private Sub GwxDisplay_PostRuntimeStart()  
    ' Change color of ellipse "MyEllipse"  
    ' to random green color when entering  
    ' runtime.  
  
    ' Define a variable of type GwxEllipse  
    Dim obj As GwxEllipse  
  
    ' Assign ellipse to variable, stop on failure  
    Set obj = ThisDisplay.GetVisibleObjectFromName("MyEllipse")  
    If obj Is Nothing Then Stop  
  
    ' Set new fill color  
    Randomize  
    obj.FillColor = RGB(0, Rnd * 255, 0)  
  
    ' Release the object  
    Set obj = Nothing  
End Sub
```

9. Pohraniti GraphWorX32 projekat izabirući Save Display1 iz File menija.
10. Testirati primjer prebacujući se na GraphWorX32 displej i ulazeći u runtime.
11. Boja elipse treba da se promjeni na slučajnu zelenu boju.

Imena objekata moraju biti jedinstvena

Ponekad je poželjno pristupiti na isti način istim objektima u svim duplikatima (klonovima) željenog simbola. Ova tehnika je korisna, napr. za VBA Wizarde, koji imaju makro iza simbola. Ovaj makro se dijeli između svih duplikata simbola i može se izvršavati na bilo kojem od ovih duplikata.

Da bi se omogućila ovakva osobina, GraphWorX32 podržava tzv. 'parcijalna' imena za objekte u simbolima. Parcijalno ime koje završava sa podcrtanom linijom (underscore) '_', napr. MyEllipse_. Duplikati od MyEllipse_ su tada MyEllipse_1, MyEllipse_2, itd. Ono što je važno je metod simbola GetVisibleObjectFromSymbol(partialName), koji prihvata ime i vraća prvo pojavljivanje specificiranog objekta u specificiranom simbolu.

Primjer 1

Predpostavimo da imamo simbol koji se sastoji od pravougaonika i elipse. Mi želimo da pišemo kod koji modificira ove objekte u svakoj kopiji simbola. Moramo doznačiti parcijalna imena na obadva objekta (dajmo im imena MyRect_ i MyEll_). Kod koji će omogućiti da se pristupi ovim objektima u specifičnom simbolu je :

```

Dim sym As GwxSymbol
Set sym = FindSomehowDesiredSymbol()
' user method to choose the symbol
Dim ell As GwxEllipse, rect As GwxRectangle
Set ell= sym.GetVisibleObjectFromName("MyRect_")
Set rect= sym.GetVisibleObjectFromName("MyEll_")
' do something with these objects
' release references
Set ell = nothing
Set rect = nothing
Set sym = nothing

```

Primjer 2

Slijedeći primjer demonstrira gornju tehniku.

1. U GraphWorX32 , kreirati pravougaonik, i dati mu ime objekta **Rect_1**.
2. Duplicirati ovaj pravougaonik da se kreira još jedan. Primjetimo da će on automatski dobiti ime **Rect_2**.
3. Grupirati Rect_1 i Rect_2.
4. Duplicirati ovu grupu koristeći taster za dupliciranje na **Draw** toolbaru.
5. Degrupirati duplicirani objekat.
6. Isčitati imena objekata koristeći dijalog Property Inspectora. Primjetimo da imena objekata od dva pravougaonika (treći i četvrti) su **Rect_3** i **Rect_4**.

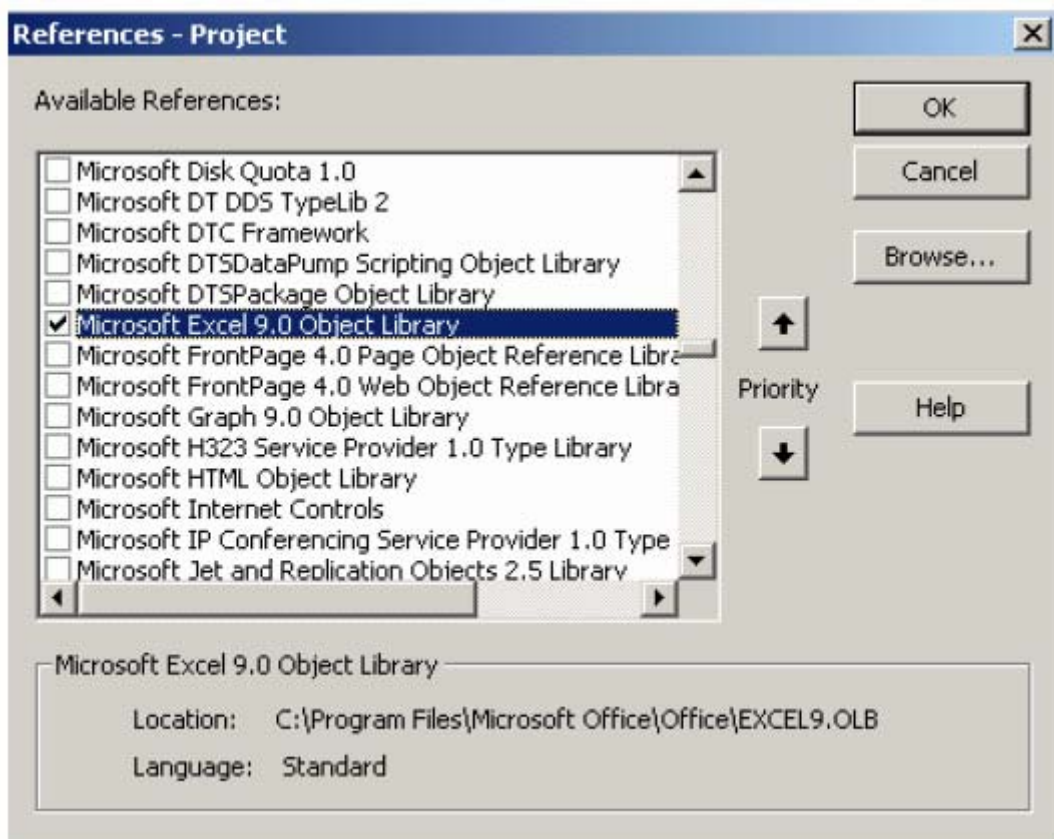
Korištenje VBA da se spojimo sa drugim aplikacijama.

Jedna moćna osobina VBA je da omogućava da se povežemo sa drugim Windows aplikacijama i razmjenjujemo podatke. U narednom primjeru, možemo vidjeti kako GraphWorX32 može slati podatke , i primiti ih od Excela, koristeći VBA.

Primjer

Dizajniraćemo displej i spreadsheet, svaki sa dvije tačke podataka, i učinićemo da komuniciraju pomoću VBA.

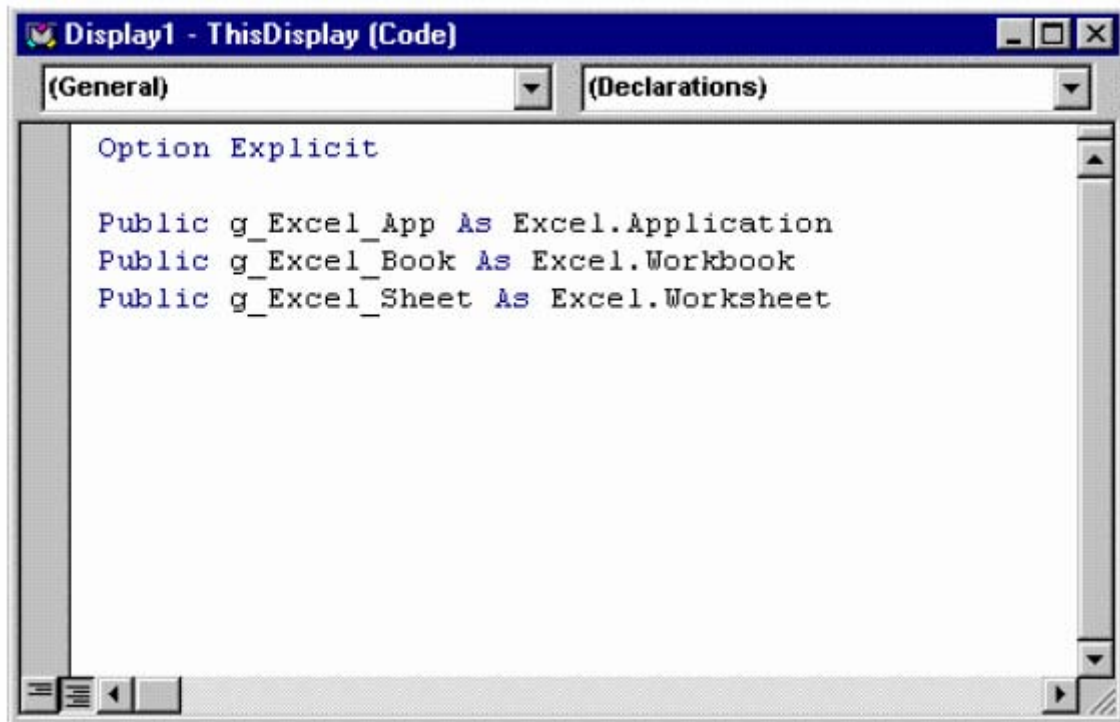
1. Otvoriti novi GraphWorX32 i ući u VBA Editor pomoću **ALT+F11**.
2. Prva stvar koju treba da uradimo je da dobijemo da VBA i GraphWorX32 prepoznaju MS Excel tipove podataka. U editoru Visual Basica, izabrati **References** iz **Tools** menija. Ovo otvara listu raspoloživih References na aplikacije. Čekirati boks pored "Microsoft Excel Object library" kao što se može vidjeti sa slijedeće slike:



3. U prozoru Project Viewera , dvaput kliknuti na modul ThisDisplay. Trebamo vidjeti prozor za kod sa slijedećim tekstom: "Open Explicit". U tom prozoru, treba da unesemo neke globalne deklaracije koje će biti korištene kasnije, kako je pokazano u nastavku:

Option Explicit

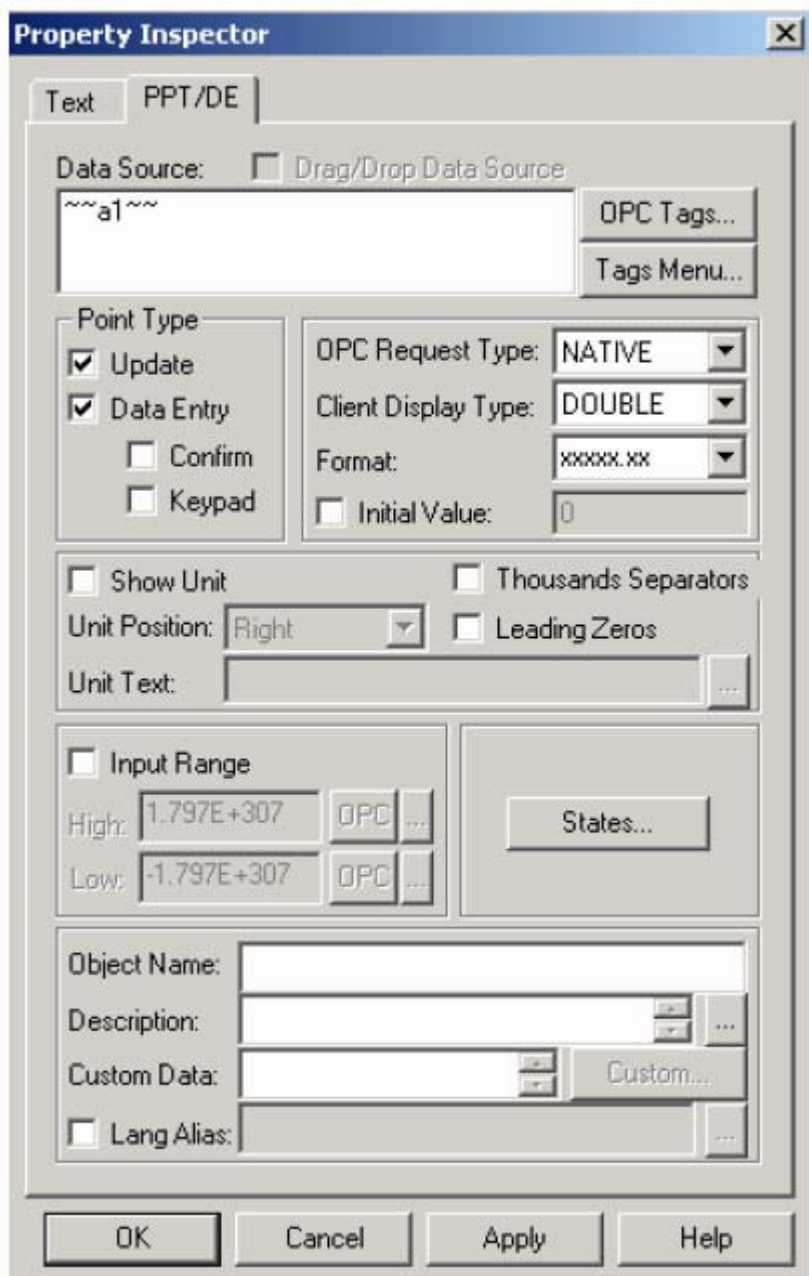
```
Public g_Excel_App As Excel.Application
Public g_Excel_Book As Excel.Workbook
Public g_Excel_Sheet As Excel.Worksheet
```



4. U lijevom kombo boksu, na vrhu prozora za kod, izabrati **GwxDisplay**. U desnom kombo prozoru, izabrati **PreRuntimeStart**. Unošenjem slijedećeg koda, učinimo da se pokrene Excel prije nego što GraphWorX32 udje u runtime mod.

```
Private Sub GwxDisplay_PreRuntimeStart()
' Open up Excel and make it visible
Set g_Excel_App = CreateObject("Excel.application")
g_Excel_App.Visible = True
' Open up a sheet
Set g_Excel_Book = g_Excel_App.Workbooks.Add
Set g_Excel_Sheet = g_Excel_Book.Worksheets(1)
' Initialize the two cells you will be using
g_Excel_Sheet.Range("a1") = 0
g_Excel_Sheet.Range("a2") = 0
End Sub
```

5. Kreirati dvije procesne tačke u GraphWorX32 displeju. provjeriti da je za obadvije procesne tačke, čekiran **Data Entry** ček boks, kako je pokazano na narednoj slici. Spojiti jednu na lokalnu varijablu `~~a1~~` a drugu na lokalnu varijablu `~~a2~~`.



6. Sada ćemo se vratiti u Editor Visual Basica. Napravićemo novu proceduru koja će čitati i upisivati u Excel. Izabrati **Procedure** iz **Insert** menija. Dati ime proceduri "Read_Write" i nakon toga unjeti slijedeći kod:

```

Public Sub Read_Write(ByVal Co As Integer)
' This procedure reads from or writes to excel
' based on the condition, Co, passed as a
' parameter

Dim Point As GwxPoint
Dim St As String 'Used to store cell name
Dim St2 As String 'Used to store variable name
Dim X As Integer
For X = 1 To 2
  ' add "a" to the value X converted to string

  St = "a" + Mid(Str(X), 2, 1)
  St2 = "~~" + St + "~~"
  Set Point=ThisDisplay.GetPointObjectFromName(St2)
  If Co = 1 Then 'Write to Excel
    ThisDisplay.g_Excel_Sheet.Range(St)=Point.Value
  Else 'Read From Excel
    Point.Value=ThisDisplay.g_Excel_Sheet.Range(St)
  End If
Next X
End Sub

```

7. Vratiti se u GraphWorX32 displej i kreirati dva radio tastera. Dati ime jednom od njih "Read from Excel" a drugom "Write to Excel". Konfigurirati obadva tako da izvršavaju VBA skript. Trigger izvršavanja treba biti "while down". Kreirati novi makro, za svaki od dva radio dugmeta. Naimenovati jedan makro kao "Rd" a drugi "Wr". Obadva ova makroa će pozivati "Read_Write" proceduru. Kod za svaki makro treba otprilike izgledati kao:

```

Sub Wr(o As GwxPick)
  Call ThisDisplay.Read_Write(1)
End Sub

Sub Rd(o As GwxPick)
  Call ThisDisplay.Read_Write(0)
End Sub

```

8. Vratiti se u GraphWorX32 displej i ući u runtime. Primjetićemo da se starta Excel automatski sa vrijednošću nula u ćelijama 'A1' i "A2". Ako kliknemo na "Write to Excel" taster, i promijenimo vrijednost jedne od procesnih tačaka u GraphWorX32, primjetićemo da vrijednost u jednoj od ćelija Excela se mijenja. Ako kliknemo "Read from Excel" taster i promijenimo jednu od vrijednosti u Excelu, procesna tačka će biti ažurirana na vrijednost one u Excelu.