



ADMINISTRATOR'S GUIDE

MANAGEMENT AND CONFIGURATION

Copyright © Vector Networks Limited. All rights reserved.

The information in this document is subject to change without notice and should not be construed as a commitment by Vector Networks Limited, Vector Networks Inc., MetaQuest Software Inc..

Vector Networks Limited assume no responsibility for errors in this document. The software described in this document is supplied under a license and may be used or copied only in accordance with the terms of such license.

PART NUMBER: DH500/00

CONTENTS

Administrator's Guide	1
Contents	3
Chapter 1 - Software And Assets Concepts	34
Assets, People and Organizations/Locations.....	34
Asset Classes, Asset Types and Assets (Instances)	35
Asset Classes	36
Asset Types	36
Asset Pools	37
Software Assets.....	39
Chapter 2 - Service Desk / HelpDesk	40
Terminology	40
Logging On.....	40
Assigning Issues.....	43
Claiming Issues.....	43
Working with Issues	43
Closing Issues	43
Related Issues and Child Issues	43
More about Workflow	44
Monitoring HelpDesk Activities	46
Revision History	46
Chapter 3 - Architecture	47
Architecture Diagram	47
Single Tier Installations	47
Multi-Server Setup.....	48
Chapter 4 - People and Organizations.....	49
Users.....	49

Managers of Employees.....	49
Importing Managers of Employees	50
Contacts.....	50
Organizations or Companies	51
User Groups.....	51
Organizational Roles and Organizational Units.....	54
Additional Fields.....	54
Chapter 5 - Asset and Software Allocation.....	55
What happens when an Asset is assigned to a person?	55
Allocations to Departments, Locations, Projects and Pools.....	56
Allocation to a Department	56
Pool of Assets	56
Asset Deallocation	57
Chapter 6 - Life Cycle of Assets and Services.....	58
Overview.....	58
RequestS and Request Portal.....	59
Customizing the Request	60
My Pending Requests	61
Approving Requests.....	63
Fulfilling Requests.....	64
Advanced Notes on the Request-Fulfillment process.....	66
Chapter 7 - Software Licenses.....	68
Expirations and Renewals.....	68
License Compliance State	69
Licensing Calculations	70
Fields.....	71
License Purchase calculations.....	74
Chapter 8 - Software Installations.....	75
Allocations vs Installations.....	75
Software License Keys Discovery.....	77

Chapter 9 - Approvals	78
Dynamic Approvers	78
<Employee Manager>	78
<Employee Supervisor>	78
<Asset Responsible>	79
Approval Exceptions.....	79
Delegates.....	79
Chapter 10 - Administering Vizor.....	80
Creating Projects and Generating Web Views	80
Administering Projects and Web Views.....	81
Related Topics	82
Managing Users, Contacts, and Groups.....	83
Users	83
Groups	83
Contacts	83
Related Topics	84
Active Directory and Email Integration.....	85
Email Integration	85
Active Directory Integration.....	88
Vizor Admin.....	92
Related Topics	92
Web View Editor.....	93
Related Topics	93
Vizor Web Admin.....	94
Related Topics	94
Troubleshooting Vizor Web Admin	95
To manually create the CensusAdminsGroup after Vizor is installed:	95
Chapter 11 - Searching and Querying.....	96
About Queries	96
Notes.....	97
Searching by Dates.....	98
Searching for Unassigned Assets or Issues	98

Finding assets or Issues Owned by the Current User	98
Finding New Assets or Issues	98
Finding Parent Issues	99
Searching Choices Lists with Relational Operators	99
Saving Queries	100
Inserting Parentheses Around Search Conditions	100
Searching for Strings.....	100
Searching for Null Values.....	101
Searching with Wildcards.....	102
Using Macros in queries.....	103
<Is Accessible Department>	103
<Is Accessible Location>	104
Chapter 12 - The Summary List.....	106
About Sorts.....	106
Sort Orders	107
Saving Sorts.....	108
Related Topics	108
About Layouts	109
Notes.....	109
Related Topics	109
Saving Layouts	110
Related Topics	110
Chapter 13 - Reporting.....	111
About Reports	112
Viewing Reports.....	112
Types of Reports.....	112
Standard Reports	113
Custom Reports	114
Creating new Custom Reports	115
Types of Crystal Reports in VIZOR.....	116
1. Dynamic Query Report.....	117

2. Dynamic Connection Report.....	119
3. Stand-alone Report	120
Customizing the Report Used by the Print Button.....	121
Exporting to CSV	122
Advanced Setting for the Report Engines: Using Crystal Reports for all Listing Reports.....	122
Time Reports.....	123
Defining the Time Axis	123
Including All Dates	124
Showing Interval Totals.....	124
Defining Reporting Periods	124
Including Totals	124
Controlling Column Width in Tabular Time Reports	125
Defining the Field Axis	125
Text (cross-tab) reports	125
Line charts.....	125
Bar charts.....	125
Defining Report Pages	126
To open the Page Editor:	126
Chapter 14 - Computers.....	128
Discovery of Computers - Bringing Computers Into Vizor.....	129
Vector Discovery.....	129
Microsoft System Center Configuration Manager (SCCM)	129
SolarWinds	129
LANsweeper.....	130
Excel / Worksheet.....	130
In-house Inventory	130
Needs Attention - Not Seen Recently.....	130
Chromebooks	131
Import Chromebooks from Google's G-Suite.....	131
Chapter 15 - Managing Contacts, Users and Groups	133
About Users and Contacts	134

Importing Users and Contacts from Windows	134
Administering Users and Contacts	135
User Security	135
User Groups.....	135
Notes.....	136
Importing User Accounts	137
To import user accounts from an Active Directory or Windows domain:	137
To import user accounts from Microsoft Dynamics	138
Needs Attention - Not Seen Recently	138
Defining Import Rules	138
Integrating with Active Directory.....	139
Integrating with Active Directory When VIZOR is in the Cloud	140
Real-Time Integration	140
Import.....	140
Filtering Users.....	140
Import Settings	142
Creating User Accounts	143
Enabling and Disabling Projects.....	144
Work Teams	145
Assigning Users to Work Teams	145
Adding Work Team Support to Web Views	146
Building Work Team Queries	146
User Categories.....	147
Assigning Users to Categories	147
Hiding Users in Projects	148
Disabling User Accounts.....	150
Editing User Information	151
Passwords.....	151
Changing the Authentication Method.....	152
Defining User Groups.....	153
Groups: Adding and Removing Users	154
Users: Changing Group Membership.....	155
Organizations & Companies.....	156

Departments	157
VIZOR Features	157
Web View Features.....	158
VIZOR Web Admin Features.....	160
Admin.....	162
Web View Editor Features	163
Chapter 16 - Fields	164
The Field Editor	165
Editing Fields	165
Destination Tab	165
Labels	165
Data	166
Field Is.....	166
Default Value.....	166
Show in Choice Lists.....	168
Maintain Revision History	168
About Adding Fields.....	169
Making Fields Read-Only	171
Using Fields to Define Styles	172
Tracking Changes.....	173
Choosing a Destination Tab	174
Setting Field Labels	175
Choosing a Field Type	176
Setting the Field Size.....	179
About Removing Fields	180
Deleting Fields	181
Hiding Fields in all Web Views.....	182
Notes.....	182
Removing Fields from Web Views.....	183
Choice Lists.....	184
Editing Choice Tables	184
Editing Choice Text	185
Renaming Choice List Tables	186

Sorting Choice Lists.....	186
Editing the Progress and State Lists.....	187
Global Choice Lists.....	189
Working with Tabs.....	189
Chapter 17 - E-mail Integration	190
E-mail Integration	191
Incoming E-mail Integration Process.....	191
Specifying the E-mail Integration Account.....	192
Incoming E-Mail Rules	193
Using Regular Expressions in the Email Rule Conditions	195
Outgoing E-mail Settings.....	197
To configure the Mail Settings:	198
Mapping Message Fields to Issue Fields.....	201
Setting Default Values for Issues Submitted by E-mail	203
Sending Automatic E-mails	205
To send an automatic e-mail as an SLA Escalation Rule Action:	205
To send an automatic e-mail as an E-mail Integration Rule Action:	205
Sending E-mails From the Web Views	206
Service Level Agreements (SLA) and E-mails:.....	206
Using the E-mail Editor	207
Sending E-mails	207
Editing the To, CC, and From Fields.....	207
Notes.....	208
Editing E-mail Content	208
Attaching Files	209
E-mail Conversation and Queue.....	209
The E-mail Conversation	209
The E-mail Queue.....	210
Processing E-mails in the WebView.....	210
Sending New E-mails.....	210
Replying to E-mails.....	211
Forwarding E-mails.....	211

Linking an E-mail to an Existing Issue	211
Creating a New Issue From an E-mail	212
E-mail Macros.....	213
using macros inside emails	214
E-mail Templates.....	215
To create an Email template	215
Links within notification template.....	216
To use an Email template.....	217
Personal E-mail Options.....	217
The "From Address" E-mail Option	218
The "Read Receipts" E-mail Option	219
Chapter 18 - Defining Workflow Rules.....	220
About Workflow Rules	221
Related Topics	221
What Can You Do with Workflow Rules?.....	221
What Can You Do with Workflow Rules?	222
Automate Workflow.....	222
Examples	224
Example 1:.....	224
Example 2:.....	225
Example 3:.....	225
Example 4:.....	226
Create Related Choice Lists	226
Examples	226
Depending on the Type (Software Defect, Hardware Defect, or Documentation Defect) you can update the Functional Area list.	227
Set Field Values.....	227
Examples	228
Related Topics	228
What You Should Know about Workflow Rules.....	228
What You Should Know about Workflow Rules.....	229
Related Topics	229

What Can You Do with Workflow Rules?	229
What is a Workflow Rule?	231
Related Topics	231
What Can You Do with Workflow Rules?	231
Creating Rule Templates	232
Related Topics	233
What Can You Do with Workflow Rules?	233
Defining Conditions	235
Defining Conditions based on User Groups	235
Defining Conditions for New Issues	235
Defining Rules	236
Notes	236
Related Topics	237
What Can You Do with Workflow Rules?	237
Deleting Rules	238
Related Topics	238
Defining Rules	238
Renaming Rules	239
Related Topics	239
Defining Rules	239
Using Macros in Rules	240
<Empty>	240
<User>	240
<Value of Field??>	240
Related Topics	241
What Can You Do with Workflow Rules?	241
Setting Possible Values	242
Substate = To Be Verified	242

Related Topics	242
What Can You Do with Workflow Rules?.....	242
Changing When Rules Are Evaluated.....	243
Related Topics	243
What Can You Do with Workflow Rules?.....	243
Changing the Order of Evaluation.....	244
Related Topics	244
What Can You Do with Workflow Rules?.....	244
Applying Workflow Rules.....	245
Notes.....	245
Related Topics	245
Defining Rules	245
Disabling Workflow Rules.....	246
Related Topics	246
Defining Rules	246
Chapter 19 - Defining Service Level Agreements	247
About Service Level Agreements	248
Setting Up Service Agreements.....	249
Target Times.....	250
Escalation Levels.....	251
About Escalation Rules.....	252
Order of Evaluation	253
Conditions.....	254
Target Time conditions	254
<Field> = <Value>	254
Meets the query <Value>.....	254
Actions.....	255
Sending E-mail Messages.....	256
Editing the To, CC, and From Fields.....	256
Notes.....	256

Editing E-mail Content	257
Attaching Files	257
Running Programs	258
Notes	258
Specifying Operational Hours	259
Defining Service Levels	260
Creating Service Agreements	261
Global Service Level Agreements	261
Applying Service Agreements to Issues	262
Disabling Service Agreements	263
Service Level Reporting	264
Service Types	265
Chapter 20 - E-mail Notifications	266
About Notifications	267
Related Topics	267
What Can You Do With Notifications?	267
What Can You Do With Notifications?	268
Related Topics	269
About Notifications	269
Editing Notifications	270
Related Topics	270
About Notifications	270
Specifying Notification Recipients	271
Related Topics	271
About Notifications	271
Defining When Conditions	272
Sending Notifications of Field Updates	272
Basing Notifications on Users	272
Related Topics	273
About Notifications	273

Specifying Notification Contents	274
Related Topics	274
About Notifications.....	274
Managing Workflow with Notifications.....	276
The tester receives the Substate update notification and reviews the fix. After the fix is verified, the tester sets the Substate to Fixed and the issue is closed.	276
Related Topics	276
Setting Up Notifications	278
Setting Up the Notification Service.....	278
To set up the notification service on Windows 2003 and Windows XP:	278
To set up the notification service on Windows 2000:	279
To set up the notification service on Windows:.....	279
Setting the Mail Server Options	279
To set the mail server options in Vizor Admin:	280
Related Topics	281
About Notifications.....	281
Handling Notification Failures.....	282
Chapter 21 - Provisioning Actions.....	285
Approval	285
Active Directory	285
HelpDesk Ticket.....	286
Execute Command	286
Change Request (Change Management Ticket)	286
Send Email.....	286
VIZOR Action.....	287
Chapter 22 - Generating Web Views.....	288
What is a Web View?	289
Related Topics	289
Generating Web Views - Issues to Consider.....	289
About the Web View Editor.....	291
Related Topics	291

Creating Web Views	291
Creating Web Views	292
Notes.....	292
Related Topics	292
Generating Web Views - Issues to Consider.....	292
Generating Web Views.....	294
Notes.....	294
Related Topics	294
Starting the Web View Editor	294
Copying Web Views	295
Related Topics	295
Starting the Web View Editor	295
Deleting Web Views	296
Related Topics	296
Starting the Web View Editor	296
Renaming Web Views.....	297
Related Topics	297
Starting the Web View Editor	297
Types of Web Views	298
Normal Views	298
Read-only Views.....	298
Knowledge Base Views	298
Submit-only Views.....	298
Self-Service Views.....	299
How licenses are used depending on the type of Web view	299
Setting Web View Attributes	301
Auto Refresh Summary List.....	301
Contact Tab in Record Details	301

Description.....	301
Display All Read-Only Fields As Text.....	301
Include VSS	301
Issues per Page.....	301
Timeout	301
Load Issues Not In Query.....	301
Web View Text	301
Web View Text CSS Class.....	301
Web View Theme	302
Related Topics	302
Creating Web Views	302
Setting Group Access Permissions	303
Related Topics	303
Creating Web Views	303
Knowledge Base Views.....	304
Using Knowledge Base Views.....	305
To use Knowledge Base views:	305
Creating and Updating a Knowledge Base View.....	305
To create or update a Knowledge Base view:	305
Create a user account to be used as the KB view logon account (see Creating User Accounts).	305
The Add to Knowledge Base Field.....	306
Types of Web Views	306
Self-Service Views	307
Using Self-Service Views	307
To use Self-Service views:.....	307
Creating and Updating a Self-Service View	307

To create or update a Self-Service view:	307
Exporting Fields	310
Notes	310
Related Topics	310
Creating Web Views	310
Exporting Queries	311
Notes	311
Related Topics	311
Creating Web Views	311
Adding the Child Issues Tab	312
Editing Field Attributes	313
Related Topics	313
Using Field Variables	313
Using Field Variables	315
Related Topics	315
Adding Pop-up Editors	316
Aligning Fields	317
Alignment	317
Vertical Alignment	317
Related Topics	317
Editing Field Attributes	317
Automatically Updating Choice Lists	318
Automatically Update List	318
Notes	318
Related Topics	318
Editing Field Attributes	318
Changing Field Captions	319
Spanning Columns	320
Fields	320

Captions.....	320
Related Topics	320
Editing Field Attributes.....	320
Applying CSS Styles.....	321
CSS Class.....	321
Displaying Read-Only Fields.....	322
Notes.....	322
Related Topics	323
Editing Field Attributes.....	323
Width	324
Related Topics	324
Applying CSS Styles.....	324
Inserting Custom HTML Code.....	325
Html code before/after control.....	325
Notes.....	325
Related Topics	326
Using Field Variables	326
Adding URL Buttons.....	327
Related Topics	327
Editing Field Attributes.....	327
Text Boxes.....	328
Number of Characters.....	328
Related Topics	328
Editing Field Attributes.....	328
Multi-Choice Lists.....	329
Number of Visible Items.....	329
Show Selected Items In Text.....	329

Show Selected Items In Tooltip	329
Configuring Choice Lists.....	330
Related Topics	330
Editing Field Attributes.....	330
Linking User Details.....	331
Related Topics	331
Editing Field Attributes.....	331
Hiding Fields	332
Related Topics	332
Editing Field Attributes.....	332
Pointing Users to Web Views	333
Related Topics	333
Changing URLs and Directories.....	333
Changing URLs and Directories	335
Changing the Windows Account.....	337
Notes.....	337
To change the Windows account used by Vizor:.....	338
Notes.....	338
Related Topics	338
Windows Permissions	338
Setting Default Values.....	339
Notes.....	339
Templates and the CustomizedFiles Folder	340
Related Topics	340
Customizing Web Views	340
Customizing Web View Files.....	341
To customize files for all Web views of all projects, create this folder structure:.....	341
To customize files for all Web views of a specific project, create a folder structure like this:	341

To customize files for all Web views accessed through a given URL, create a folder structure like this:	341
Related Topics	342
Templates and the CustomizedFiles Folder	342
Customizing Shared Web View Files.....	343
To customize the files in CensusWebVD for all Web views:	343
Put the files in the folder	343
To customize files for all Web views accessed through a given URL:	343
Put the files in the folder	343
Related Topics	343
Templates and the CustomizedFiles Folder	343
Overriding Customizations.....	344
Related Topics	344
Templates and the CustomizedFiles Folder	344
Customizing the Web View Interface	345
CensusMain.css	345
TmplCensusMain.htm	345
TmplToolbar.htm	345
TmplSummaryList.htm	345
TmplSQE.htm	345
Tabs.asp	345
TmplCmdBar.htm	345
Changing the Date and Time Formats for Web Views.....	346
Administering Vizor on the Web.....	346
Logging On.....	346
Generating Vizor Web Admin	347
Windows Permissions.....	347

Improving Web View Performance	347
Chapter 23 - Maintaining Your Vizor System	348
Relocating Databases to SQL Server.....	349
Notes.....	350
Related Topics	350
Troubleshooting Relocation	350
Troubleshooting Relocations.....	351
Related Topics	351
Relocating Databases to SQL Server.....	351
Relocating Attachments	352
Notes.....	353
Related Topics	353
Relocating Databases to SQL Server.....	353
Deleting Issues	354
To delete issues from within the Vizor Views:	354
About the Logons Editor	355
Name.....	355
Date/Time	355
Workstation	355
Project.....	355
Virtual Directory.....	355
Application.....	355
Related Topics	355
Logging Off Users	355
Logging Off Users	356
Related Topics	356
About the Logons Editor	356

Setting the Monitor Interval.....	357
About the Integrity Editor.....	358
Related Topics	358
Remove Locks.....	358
Remove Locks.....	359
Locks on Project Definitions.....	359
Locks on Users and Logon Sessions	359
Related Topics	359
About the Logons Editor	359
Validate Integrity of Fields.....	360
Validate Relationships	361
Related Topics	361
About the Integrity Editor.....	361
Validate Integrity of Issues	362
Related Topics	362
About the Integrity Editor.....	362
Enabling Auto Repair.....	363
Releasing Databases	364
Related Topics	364
About the Logons Editor	364
Uploading Large Attachments	365
Related Topics	365
Fixing 404 Errors	365
Fixing 404 Errors.....	366
Notes.....	366
Related Topics	366
Uploading Large Attachments.....	366
Chapter 24 - Importing Data	367
About Importing Data.....	368

Source table	368
Source fields	368
Source fields are the fields imported from the source table	368
Destination fields	368
Mapped fields	368
Related Topics	368
What Can You Import?	369
What Can You Import?	370
How Records are Created	371
Issue Number	371
Owner (nUserID)	371
Submitter (nSubmitterID)	371
Contact (nOriginatorID)	371
Progress	371
State	371
Time and Date fields	371
Choice lists	372
Related Topics	372
About Importing Issues	372
Before You Import	373
Add fields for imported data	373
Create a Number field to hold imported IDs or record numbers.....	373
Create users and contacts.....	373
Customize Vizor choice lists to match the imported data	373
If you are importing text files.....	374
Back Up Projects	374
Log off users	374
Related Topics	374

Backing Up Projects.....	374
Importing Issues into a Source Table.....	376
Related Topics	377
About Importing Issues	377
Combining Multiple Source Tables.....	378
Related Topics	378
About Importing Issues	378
Importing Issues from Projects.....	379
Related Topics	379
About Importing Issues	379
Creating Queries to Merge Source Tables.....	381
Related Topics	381
About Importing Issues	381
Mapping Import Fields.....	383
Mapping Text Fields.....	383
Mapping Memo Fields	383
Importing Choice Lists	383
Importing Users, Submitters, and Contacts	384
Chapter 25 - VIZOR Licenses	385
Entering Your License Information.....	386
Entering License Keys	386
Number of Licenses	387
License Type	387
Expiration Date	387
Reserving Licenses.....	387
Chapter 26 - Integrating Remote Control.....	388
About Remote Control.....	389
Related Topics	389
Setting Up Remote Control.....	389

Setting Up Remote Control.....	390
Related Topics	390
About Remote Control.....	390
Adding Remote Control Buttons.....	391
Related Topics	391
About Remote Control.....	391
Configuring ActiveX for Remote Control.....	393
After you customize RCViewer.asp, copy the file to.....	393
DEFAULT_SECURITY_KEY	393
DEFAULT_COMPRESS	393
DEFAULT_ENCRYPT.....	393
DEFAULT_SCALE_TO_FIT	394
DEFAULT_SCROLL_BARS	394
DEFAULT_VIEW_TYPE	394
DEFAULT_INIT	394
Related Topics	394
Setting Up Remote Control.....	394
Checking Client Security Key Settings.....	395
Related Topics	395
About Remote Control.....	395
Removing Remote Control Buttons.....	396
Related Topics	396
About Remote Control.....	396
Chapter 27 - Projects	397
What is a Project?.....	398
Project Structure	398
Related Topics	398
Creating New Projects.....	398
Creating Projects	400

Related Topics	401
What is a Project?	401
Copying Definitions and Styles	402
Why copy styles and notifications?	402
Related Topics	402
Creating New Projects.....	402
Building a Project Template	403
Related Topics	403
What is a Project?	403
Deleting Projects	404
To delete a project:	404
Related Topics	404
What is a Project?	404
Editing Project Properties	405
Project Location	405
Definitions Database	405
Database Engine	405
Groups Allowed to Open Project	405
Current Users	406
Name	406
Date/Time	406
Workstation	406
Project	406
Virtual Directory	406
Chapter 28 - Planning and Deploying a Custom Project.....	407
Planning	408
Implementation Roadmap.....	409
Step 1. Create a project	409
Step 2. Define user groups and create user accounts	409

Step 3. Define and generate Web views.....	409
Step 4. Define field dependencies (optional).....	410
Step 5. Define a workflow (optional).....	410
Step 6. Define Service Level Agreements	410
Step 7. Import existing data (optional)	410
Step 8. Define and set up notifications.....	411
Step 9. Go live	411
Step 10. System maintenance (ongoing)	411
Creating a Project.....	412
How Many Projects Do You Need?.....	412
Choosing a Base Project.....	412
Customizing Fields	413
Notes.....	414
What Happens to Styles.....	414
Notes.....	414
When you rename a field.....	414
When you edit a choice list	415
When you delete a field.....	415
Creating Groups and Granting Permissions.....	416
The Built-in Groups	416
Users	417
Admins	417
Guests	417
Features.....	417
Projects.....	417
Views.....	418
User Accounts.....	419
Submit-only Views.....	419
Importing User Accounts from Windows.....	419
Predefined User Accounts.....	419
Generating Web Views.....	421
Parent-child relationships between issues?	421
Customized files from the base project?.....	421

What if you don't export a field?	421
Queries.....	422
Layouts.....	422
Sorts.....	422
Listing Reports.....	422
Custom Reports.....	422
Should you automatically update choice lists?.....	423
Changing field labels	423
When do you have to regenerate?.....	423
Notes.....	424
How to control the layout of fields	424
Setting Up Notifications	425
Chapter 29 - Customizing Vizor.....	427
Power Customizations.....	428
About the Databases	428
Issue database	428
Project definitions database	428
Users database	428
Advanced Web View Settings	429
Logon Page	430
Simplify the URL	430
If Not using Integrated Authentication - SSO.....	430
If mostly using Integrated Authentication	431
Total Effort Time.....	432
Displaying the Total Effort Time in another tab.....	433
Creating an Attachments Field.....	435
To create an attachments field:	435
Customizing URL Buttons	436
Customizing Standard HTML Reports.....	438
Changing the CSS Styles	439
Notes.....	439
PlaceHolders.....	439

Removing Disabled Users from Choice Lists	440
Adding a Choice List of Users or Contacts	441
Optional behaviours In HelpDesk	442
Enabling Timestamping in New Projects	442
Activity	443
Adding Timestamping	444
Customizing Timestamping	445
Timestamping with One Memo Field	446
Html code after control:	446
Html code before control:	446
New function in CustomCode.js:	446
New declaration in CustomCode.js:	447
Changing the Format of Attachments	448
nFormatType constants:	448
Attaching Information	450
Adding Custom Mail Contents	451
Example Macros	452
Function to return a string:	452
Function to return the contents of a text file:	452
Function that takes the revision history as an argument and extracts a field value:	452
Related Topics	453
Adding Notification Reports	454
To create an .rpt file for your notification report:	455
Submit-only Views Without Persistent Cookies	456
Chapter 30 - Branding Web Views	457
Replacing Logos	457
Summary List Toolbar	459
Summary List Header	460
Summary List	461
Tabs	462
Form	463
Field Captions	464

Form Elements.....	465
Command Bar.....	466
About Branding Dialogs	467
Common Styles for All Dialogs	468
Attachments Dialog.....	469
Reports Dialog.....	470
Revision History Dialog.....	471
Options Dialog	472
Contacts Dialog.....	473
Password Dialog.....	474
About Branding the Logon Page	475
Validate Logon Page.....	476
Adding Rows.....	477
Changing the Timesheet Starting Day	479
Creating a Timesheet.....	480
Chapter 31 - Databases	482
Overview.....	483
DAT Database.....	484
Main Tables	484
Definitions Database.....	486
Main Tables	486
Users Database	488
Main Tables	488
Global Databases	489
Advanced Topic.....	489
Related Topics	490
Chapter 32 - Backups.....	493
About Backing Up.....	494
Backing Up Jet/MS Access databases	494
Backing Up SQL Server databases	494
Locating databases	494

Attachments.....	495
Web views.....	495
Related Topics	495
Backing Up Issues	497
What to backup	497
Notes.....	497
Related Topics	497
Backing Up Projects	499
What to backup	499
Related Topics	500
Backing Up the Users Database.....	501
What to backup	501
Related Topics	501
Backing Up System Files.....	502
What files to backup	502
Related Topics	502
Backing Up Web Views	504
What to backup	504
Related Topics	504
Automatic Backups.....	506
Related Topics	506
Performing Hot Backups	507
Related Topics	507
Restoring Backups	508
Related Topics	508
About Backing Up.....	508
Chapter 33 - Relocating A Vizor Installation	509
About the Relocation Wizard	510
Related Topics	510
Before You Relocate.....	511
Check Permissions	511
Close all copies of Vizor Admin and the Web View Editor	511

Relocate Projects	511
Back Up	511
Log Off Users	511
Related Topics	512
Relocating Vizor.....	513
Related Topics	513
If Something Goes Wrong	515
Updating Client Programs	516
Enabling Vizor after Relocation	517
To enable Vizor after a relocation:	517
Index	518

CHAPTER 1 - SOFTWARE AND ASSETS CONCEPTS

ASSETS, PEOPLE AND ORGANIZATIONS/LOCATIONS

Vizor is designed around the concepts of People, Assets and Organizations/Locations.

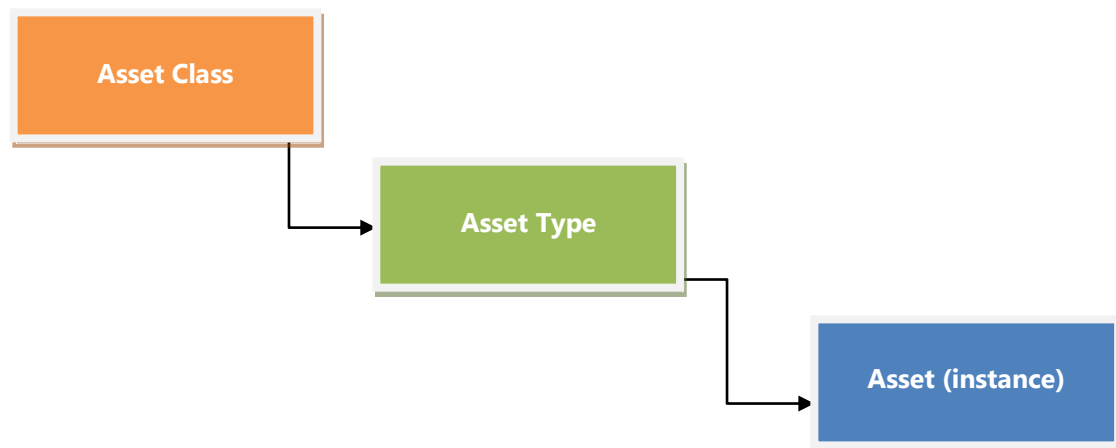
Vizor helps you manage your assets and how they are used by people and the organization throughout time.

Organizations and locations are related concepts that are managed separately depending on your needs.

If you use Vizor for managing your organization's internal assets, you may have a single Organization with one or many locations.

Alternatively, if you use Vizor for managing your customer's assets, you may have multiple organizations each one with one or many locations.

ASSET CLASSES, ASSET TYPES AND ASSETS (INSTANCES)



In Vizor, an asset represents any type of resource assigned to a person or organization. An asset can be an instance of an installation of a Software Application, a physical device such as a laptop or desktop, it can also be equipment like a projector, or even non-physical assets like a virtual machine or access to a Cloud Software Application.

Assets in Vizor are often but not exclusively IT Assets. Due to being highly configurable, Organizations use Vizor to track all types of assets that are valuable to them.

An Asset, also known as an instance of an asset, has a unique life-cycle that can be traced and managed with Vizor. Each asset has a specific life cycle that often looks like the following:

- start with a **request** to get the asset
- follow with the workflows and **approvals** to purchase the specific asset
- the **purchase** process to obtain it
- the arrival of the asset to the organization
- the **allocation** of the asset to a person, location, department or project
- the **provisioning** of the asset to the intended user or location
- the **de-provisioning** of the asset after used or when **reallocated**
- some possible cycles of **configuration** or **repair** of the asset
- and usually finished with the **retirement** or **disposal** of the asset

ASSET CLASSES

Assets are first categorized into Asset Classes. Asset Classes define the top-level type of assets and allow your organization to manage asset types in a structured way, with general categories such as "software" and "computers".

Typical examples of Asset Classes are:

- Software
- Computer
- Consumables
- Document
- Hardware
- Network Share
- Telecom

Can I customize the Asset Classes?

You can hide asset classes that are not in use by your organization.

You can rename the existing classes to align to the needs of your organization or industry, and you can also create your own asset classes.

Unlike Asset Types, Asset Classes do not have any specific life-cycle, behaviours or business rules and they serve more to categorize or group the Asset Types they contain.

ASSET TYPES

The next level of categorization is the Asset Type. An asset is always created as an instance of an Asset Type. Every Asset must come from an Asset Type.

The Asset Type defines the exact kind of asset it is and establishes the main characteristics of the asset such as:

- The Life-Cycle of that asset, from procurement to retirement
- The Actions to execute when provisioning or de-provisioning the asset
- The Approvals and workflows required along the life-cycle of the asset
- Costs and other purchase information
- Depreciation calculations

An Asset Type is similar to a template of an asset, and in fact many attributes of an asset (instance) are inherited from the Asset Type.

Typical examples of Asset Types are:

- Microsoft Office Software
- Desktop computers
- Laptops
- USB Keys

An Asset Type always belongs to an Asset Class. You cannot have an Asset Type that does not belong to an Asset Class. However, you can move an Asset Type from one class to another.

Examples of Asset Class → Asset Type → Asset hierarchies:

Asset Class	Asset Type	Asset
Computer	Desktop	HP Envy 750 SN:4536-45645
	Laptop	Dell Inspiron SN:RT36-43242
Hardware	Projector	Epson PowerLite SN:JHFY-4578
	External Drive	Seagate Backup Plus SN:48-JHG-7856
	Monitor	Dell 55 E5515H SN: OHH987IH

Asset Pools are a special type of assets that contain other assets. Most of the times, a Pool is like a container and it represents a real, physical item that contains the included assets, like a box, a pallet or a bin. However, a pool can also be virtual or immaterial, useful to create a set or logical group of the assets.

For example, you can organize assets of various types into a container, and then allocate the container to someone for transportation or for assigning the included assets to others.

Example scenario with Chromebooks.

A school is receiving Chromebooks for the students and the supplier is sending them in containers. A Chromebook Container is a type of Asset Pool and it has all the Chromebook devices for the students of a given class (school class).

The Pools or Containers can be created manually or imported using the pre-created template for importing Chromebook Containers. An administrator can log into Assets and select the query "Containers" to see all the containers in VIZOR, regardless of who is responsible for each one. If not already done, the administrator can assign a Chromebook Container to a teacher.

SOFTWARE ASSETS

Vizor Software focuses on managing your assets of class **Software**. Software Assets behave in general like any other asset, but they also have unique, extended characteristics and workflows that are exclusive to Software. For example, Software Assets have the following additional functionality:

- Software Licenses records
- Software Installations
- License Compliance
- Special Allocation/Deallocation

CHAPTER 2 - SERVICE DESK / HELPDESK

Vizor ServiceDesk (also known as Vizor HelpDesk) is a flexible, Web-based system for centrally tracking and managing help desk issues and calls. Because it is Web-based, it enables users to access the issue database from any browser, anywhere, any time. It also minimizes deployment effort because there is nothing to install on the PCs it supports.

Vizor ServiceDesk works like a Windows application rather than a set of static HTML pages that must be constantly updated and reloaded. This enables you to easily customize HelpDesk to suit the way you want to work. When you run a query, view an issue, or submit a new issue, HelpDesk dynamically updates its interface. It uses pop-up dialogs for tasks like generating reports and attaching files to records.

TERMINOLOGY

The following terms are used to refer to HelpDesk components:

Projects are databases or knowledge bases. All Projects share user accounts, user group privileges, and the administration utilities, but each Project has a separate database.

Web views are customized views of Project databases. Each Web view is designed to meet the needs of a class of users. For example, employees who do not belong to the support department only need to see the issues they submitted, not all the issues in the Project database or the notes added by the support staff. Conversely, support staff need access to all the issue information in the Project database.

LOGGING ON

Vizor HelpDesk contains three different views of the default HelpDesk Project. Each view shows a different set of issues and fields. For example, the Employee view shows only the issues and information entered by the user.

When you log on, HelpDesk displays the views you have permission to access. Access to views is controlled by user permissions, which are set up by the Vizor administrator.

In the default example, the **admin** and **demo** user accounts have administrative-level permissions and can open any view. The other accounts are role-based, and have restricted permissions and access to views. Use the role-based accounts to see how permissions and workflow rules control the issue tracking process:

- **HelpDesk** shows all issues and issue information. It enforces workflow rules and defines an issue lifecycle, and is intended for use by help desk staff. Use the following accounts to access the view:

Logon accounts:	admin	(Password = admin)
	demo	(No password)
	analyst	(No password)
	group leader	(No password)

- **Employee** enables employees to submit new issues and to check the status of existing issues. It shows only the issues and information submitted to the help desk by an employee. Internal information added by help desk staff is hidden.

Logon accounts:	employee	(No password)
-----------------	-----------------	---------------

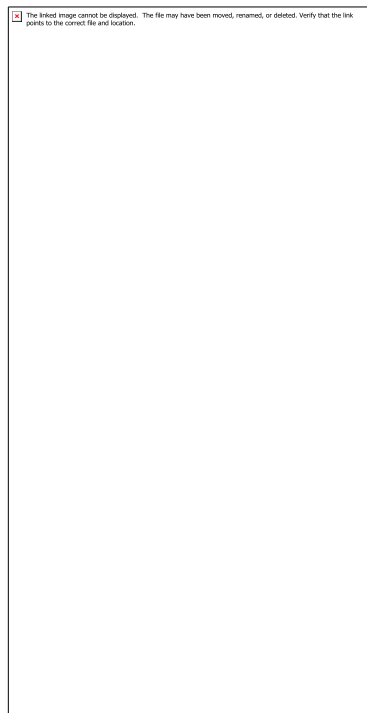
- **Report Issue** is a submit-only view that enables users to submit new issues to the help desk. The Report Issue view does not require a logon account. You can access it directly through a URL.

THE ISSUE LIFECYCLE

When an issue or problem is reported using HelpDesk, it is passed on to support staff. The investigation of the issue, and any work required, is then performed in accordance with the organization's issue-handling process.

In Vizor ServiceDesk, the steps in the issue-handling process are implemented as a *Workflow*. Each step in the process corresponds to a Workflow value shown in the issue's **Progress** field or Status. Support staff advance an issue through the process by changing this value. Control of the work is maintained by assigning different permissions to change the **Progress** field to different types of user. For example, in the default process included in the standard HelpDesk installation, only group leaders are allowed to assign an issue to support staff.

This diagram summarizes the process required to solve a simple problem using the default HelpDesk installation:



ASSIGNING ISSUES

When an employee submits a new issue through HelpDesk, it is recorded with a progress status of New indicating it is awaiting assignment. It can then be viewed by a group leader and assigned to an analyst based on the information supplied.

CLAIMING ISSUES

While an issue is marked **Assigned**, it is waiting to be claimed by a help desk analyst. An analyst claims an issue by marking it **In Progress**, which indicates that the analyst is working on the problem.

WORKING WITH ISSUES

In its standard configuration HelpDesk gives analyst users direct access to the software and hardware information stored in the Vizor Assets database. This provides a quick and easy method of collecting the extra information analysts need to investigate an issue, even when the person who reported it is not available.

In addition, with the PC-Duo Remote Control application, analysts can view and take control of the PC where the problem occurs, so that they can help even the most inexperienced users.

CLOSING ISSUES

When a problem is fixed or work on an issue is complete, analysts change its **Progress** status to **Resolved** to indicate work is closed.

RELATED ISSUES AND CHILD ISSUES

Often, a single problem can give rise to several different issues. For example, a server going offline could produce issues on problems with e-mail, access to the Internet and users being unable to access their files. If you have multiple issues resulting from the same underlying cause, you can link and manage them using the Child Issues tab.

MORE ABOUT WORKFLOW

The preceding exercises demonstrated how the workflow controls the steps an issue follows as it progresses through the help desk process. They also showed how each step in the process is represented by a status value in the **Progress** field.

At each step in the process, an issue is also considered *open* or *closed*. Open issues are still active, or awaiting information. Closed issues have been resolved (fixed) or dropped (the problem can't be fixed or reproduced). The read-only **State** field indicates whether an issue is open or closed.

The default workflow implemented in Vizor ServiceDesk is slightly more complicated than the simple process shown previously.



MONITORING HELPDESK ACTIVITIES

Vizor ServiceDesk automatically tracks the progress of issues through the issue lifecycle. This enables users to check on the progress of issues and also provides reporting capabilities that allow managers to track and monitor support activities.

REVISION HISTORY

Vizor ServiceDesk maintains a Revision History for each issue. The Revision History is a list of all changes made to the issue after it was first submitted. Each time you save an issue, HelpDesk records the changes in the Revision History.

CHAPTER 3 - ARCHITECTURE

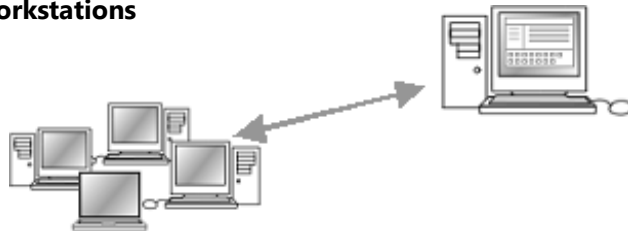
ARCHITECTURE DIAGRAM

Vizor is a Web based product, the architecture is relatively simple. All of the components below could be located on one Server.

SINGLE TIER INSTALLATIONS

Web Server + Database Server Mail Server

Workstations



- Web Browser (for accessing Vizor and Vizor Web Admin)

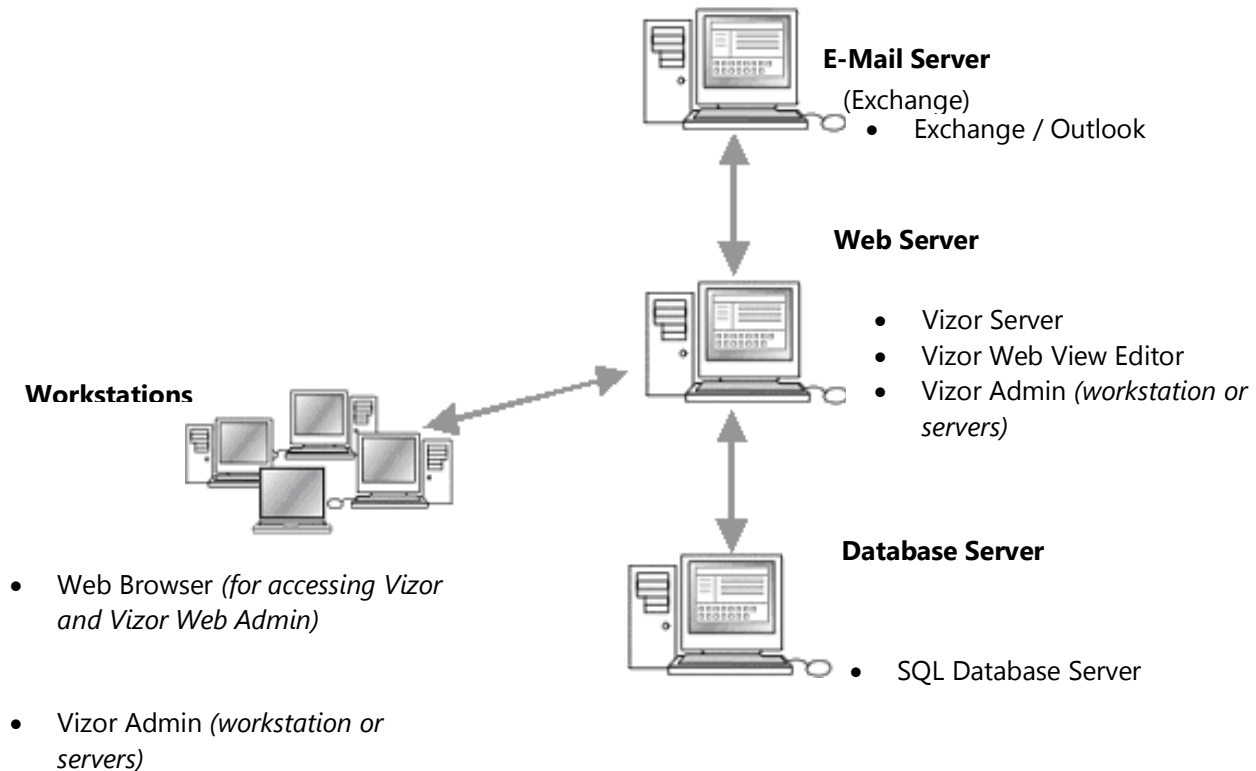
- Vizor Server
- Vizor Admin (workstation or server)
- Vizor Web View Editor
- SQL Database Server

- Vizor Admin (workstation or server)

MULTI-SERVER SETUP

The Vizor architecture supports the following multi-server architecture.

To install this configuration several items need to be manually configured on each server.



CHAPTER 4 - PEOPLE AND ORGANIZATIONS

In Vizor, people can be of two main types: Users and Contacts. People accounts can also belong to an organization or company.

USERS

Users can log into Vizor and to do so they have a logon account and credentials (user and password). The logon account may be native to Vizor or it may be via a 3rd party such as Active Directory or LDAP.

Users have a logon account and they can log into VIZOR to use any Self-Service area that has been allowed by the administrators of the system.

Employee is a type of user that is commonly used when managing assets internal to an organization. Employees are often imported from or synchronized with Active Directory or an HR system.

MANAGERS OF EMPLOYEES

Occasionally, some workflows make use of the concept of the “manager of an employee”. For example, requested assets often need to be approved by the manager of the employee requesting the asset, in which case it is not practical to add a fixed name in the list of approvers.

For these scenarios, Vizor has a special macro is dynamically resolved and that allows you to send the approval to the person that happens to be the manager of the employee at the time of the request.

WHO IS MY MANAGER?

The Manager of an employee is the person that is responsible for the Department, according to the employment information. Because the Manager is defined in the Department (Roles & Policies area), this means it is easy to make a change when the manager is changed.

To know the manager of an employee:

1. In the Employees area, select the person you want to know.
2. Click on the Employment tab and check what department the employee belongs to.
3. Go to the Roles & Policies area and select the Departments query.
4. Click the name of the department the employee belongs to from the list or search for it.
5. The name of the manager is displayed in the **Responsible** field.

IMPORTING MANAGERS OF EMPLOYEES

If you are importing the list of employees from an external source or synchronizing with a Human Resources (HR) system, it is possible to also import the manager of each employee. The actual mapping of fields depends on the type of source. Please contact support for more details on how to import this information.

CONTACTS

Contacts cannot log into the system but are referred to in processes, assets and tickets. They can receive emails and be used in fields of assets, licenses, purchases, tickets, etc. For example, a contact could be a person that sells software from a vendor and does not need to log on to Vizor.

ORGANIZATIONS OR COMPANIES

Organizations are also referred to as Companies. You can have one or many Internal Organizations and one or many external organizations. Examples of External Organizations are: Customers, Vendors and Resellers.

Customers are Companies or External Organizations in Vizor.

Each Company can have one or more associated people. A Company record can have both associated Contacts and Users.

USER GROUPS

User Groups in Vizor are like roles in the Vizor system, which allows easier administration of views, accesses, permissions and other rules to fit the needs of the organization.

Each roles is represented by a Security User Group, managed from the Web Admin.

These roles have a default combination of views and features that are typical for their use of the system.

The roles are very useful to profile the different functions of users in the Vizor system depending on their function or focus of interest (software licenses, it hardware assets, hr-it, helpdesk, etc.).

You can create any number of User Groups to fit the requirements of your organization.

Default User Groups or Vizor Roles:

User Group	Demo user	Function	Views
Software Managers	Daniel Ambler Logon=software	Manages Software and Software Licenses	Software Licenses Agreements & Purchases

			Employees
			My Actions
IT Asset Managers	Donald Nathan	Manages IT assets, primarily hardware assets	Assets
	Logon=assets		Employees
			Locations
			Agreements & Purchases
			My Actions
			Roles & Policies
IT Managers	Martin West	Manages general IT activities, budgets, assets, etc.	Assets
	Logon=itmanager		Software Licenses
		Manages policies for asset provisioning.	Employees
			Locations
			Agreements & Purchases
			My Actions
			Roles & Policies
			Approve Change Requests
			Help Desk Incidents
Employees	Employee User	Regular staff that uses	Request Assets

	Logon=employee	services from IT and HR, such as submitting tickets to HelpDesk and requesting assets	Search KB Submit a HelpDesk Ticket
Helpdesk Analyst	Help Desk Analyst Logon=analyst	Manage helpdesk incidents and problems. Edit knowledge base and FAQ articles	Help Desk Incidents & Problems Change Requests Search KB
HR Manager	Marianne Rosen Logon=mrosen	Monitor employee hires, terminations and turnover across the organization. Initiate employee Onboarding and termination processes.	New Employee Onboard Batch Onboard Terminate Employees Employees Locations Roles & Policies
Release Manager	Chris Hubbard Logon=chubbard	Uses the release manager project only. For select demos at the moment.	Release Management

ORGANIZATIONAL ROLES AND ORGANIZATIONAL UNITS

In order to manage better your assets and automate the provisioning or de-provisioning, Vizor allows you to define and organize roles within your organization, map users to roles, and map assets to roles.

For the assignment and organization of roles, Vizor uses the following types of entities:

- Organizational Units
- People/Employees
- Roles
- Assets

ADDITIONAL FIELDS

Organizations may need additional fields for their User, Contact, Customer and Company accounts. If you need an additional field, chances are Vizor already has it as a built-in field but it may not be visible in the user interface.

You can view the list of all fields regardless of whether they are published in a Web view or user interface by going to the Web Admin > Fields > Fields.

Some example fields are:

Field Name	Notes
Second Last Name	Some countries use a Second Last Name or Surname. This is the case for most Spanish-speaking countries, for example.

CHAPTER 5 - ASSET AND SOFTWARE ALLOCATION

You often need to track the allocation of Software Assets. Assets are typically assigned to a person, like an employee, but they can also be allocated to other entities like departments. Assets can be allocated to the following:

- Person (Employee or User)
- Location (room, building, etc.)
- Department
- Project
- Unit or Division
- Computer (in the case of Software)

Vizor keeps track of how many licenses you have and identifies any surplus inventory. It decrements your available licenses every time a license is allocated and increases it when the license is de-allocated. Hence your inventory stays current and accurate.

Tracking allocation with Vizor helps you manage costs and budgets, it helps you know if and when all your surplus licenses get redeployed. Vizor allows you to start the request for purchasing new software or stop taking orders depending on your organization's preference.

WHAT HAPPENS WHEN AN ASSET IS ASSIGNED TO A PERSON?

When an asset is assigned to a person, the Status changes to Assigned and the Assigned Date and Assigned Time fields are automatically set to that date and time. The assigned person is also known as the Owner of the asset.

ALLOCATIONS TO DEPARTMENTS, LOCATIONS, PROJECTS AND POOLS

In addition to assigning an asset to a person, it is often needed to allocate a number of assets to a Department, to a Location, to a Project or to an arbitrary pool of assets for which a person is responsible for.

ALLOCATION TO A DEPARTMENT

To allocate one or more assets to a specific department, set the field "Allocated to Department" in the details of the asset.

You can use the Mass Update function when allocating multiple assets to the same department. To do this, select all the assets you want to allocate by using the use check box on the last column of the summary list. Then click on the Mass Update button and set the field to the desired department.

When an asset is allocated to a department it can still be assigned to a person, at any time. This allows departments to receive pools of assets and assign them to people at a time when is needed.

POOL OF ASSETS

In some cases, the pool of assets is not based on a department, project or division. VIZOR allows you to create a type of asset called **Pool** and add arbitrary assets to it. You can then designate a person responsible for the pool of assets, regardless of the person's position, department or location.

For example, a team leader is given 10 licenses of software ABC, which she will later assign to the team members. In a similar way to allocating assets to a department, simply create a pool and add the 10 assets using the Mass Updates function (software in this example).

A person can view what assets she is responsible for by selecting "My Pools" in the query list of the Assets view.

ASSET DEALLOCATION

Vizor is a great tool to optimize the usage of your Software and other assets as people join the organization, leave the organization or change roles, thus having different needs with regards to assets.

The following are typical scenarios where Vizor is used:

- New Employee starts (On-boarding)
- Employee changes position
- Employee terminates job
- Employee requests a new or different asset in addition to existing ones
- Employee requests a new or different asset to replace an existing one
- Employee returns asset from a temporary loan
- One asset needs to be allocated to a different location
- IT person finds an asset that should no longer be allocated

CHAPTER 6 - LIFE CYCLE OF ASSETS AND SERVICES

OVERVIEW

Assets and Services have a life cycle. VIZOR helps organizations track these life cycles and make the associated processes efficient, optimized and collaborative.

The default life cycle of an asset or service follows this general sequence:

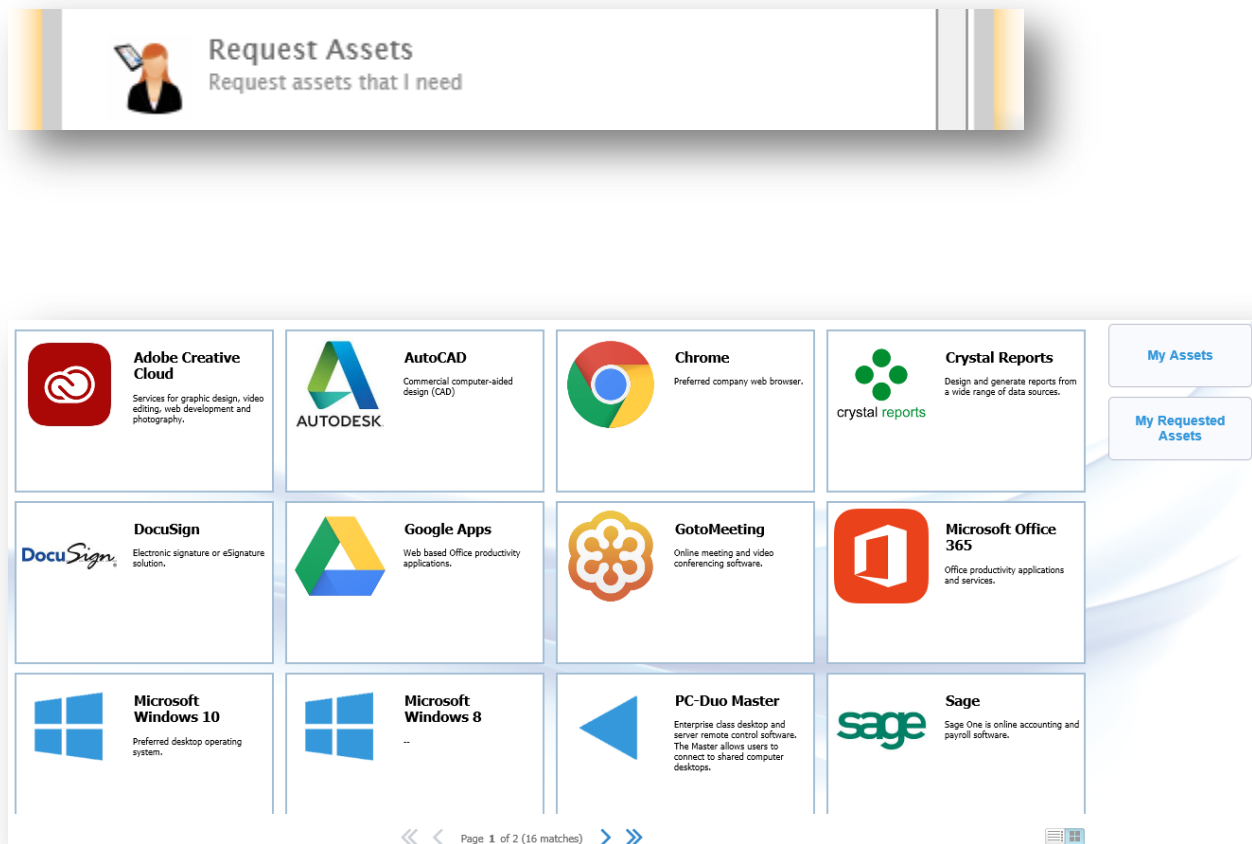
- Request
- Approval
- Fulfillment or Allocation
- Provisioning
- Use
- Deprovisioning



During the Use stage, the asset is assigned, typically to a person, and it is considered active, in production.

REQUESTS AND REQUEST PORTAL

The process usually starts with the Request of the asset by someone. Employees and users can request assets and services via the Request Portal.



The Request Portal is accessible to all employees by default. You can change this by modifying the groups that can access the Web View "Request Assets" in the Web View Editor.

By default, the Request Portal allows users to request assets and services for themselves.

It is also possible to allow users to request assets for someone else via the Request Portal. Please contact our support team if you need this configuration.

The administrator of Vizor can specify what Asset Types are visible or not in the Request Portal. This is defined per Asset Type. If you want a specific asset type to be visible in the Request Portal, open the definition of the asset type in question and set the field "Can be Requested" to Yes.

CUSTOMIZING THE REQUEST

The request of an asset presents the user with a form used to provide additional information needed to provision the asset. For example, Vizor can ask the user the date when the asset is required for, or the number of assets to provide.

Some asset types may require additional information, for example, if your organization has two types of models for computers and you want the requestor to decide, you can add a custom field to the form for that particular asset type.

Vizor comes with a number of commonly-used fields to choose from and you can decide if the field should be requested or not for that particular asset type.

To define the fields to include in the request, go to Assets > Manage Asset Types and select the asset type you want to change the request form for. Click on the Life Cycle tab and click on Request.

Overview Details **Life Cycle** Actions Assets Purchase Details Mapping

Click on the diagram to configure the life cycle stages.

Request Approval Fulfillment Provisioning Use De-provisioning

Showing Life Cycle Stage: Request

Select the fields that should be shown when someone requests an asset of this type.
The information entered in these fields will be accessible when approving the request.

Visible Request Fields:

- Associated Cost (Request Details Tab)
- Content Loaded on Device
- Deliver To
- For international travel
- For Travel
- Laptop Accessories Needed
- Litigation Software

Add and remove fields from the list as you need.

Click OK to apply the changes and include the new fields in new request of that asset type.

DO YOU NEED A FIELD THAT IS NOT IN THE LIST?

If you need a field that is not in the list, you can create your own customized field using the Field Editor.

Go to Web Admin > Fields and select the Resources project. Click on Add to create a new field.

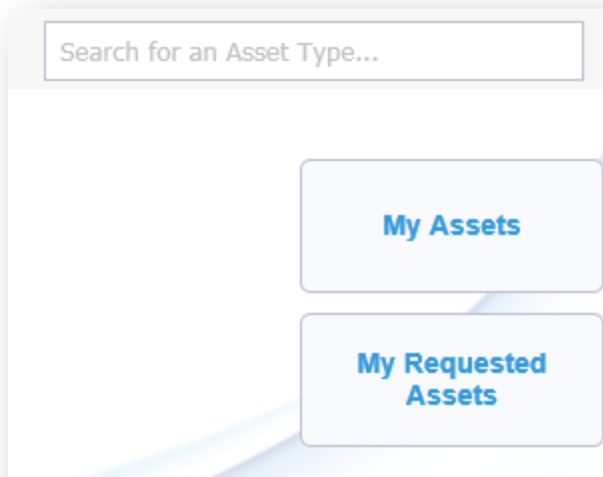
In order for this field to be used in the Request Portal, you need to:

- Create it in the **Request Details** tab
- Set **Is Visible** to No (clear the checkbox). This is so that the field is not visible for all asset types in general, but only the ones you explicitly define it to be.

See Chapter 16 - Fields for more information on customizing fields.

MY PENDING REQUESTS

Users can also check their pending requests inside the Request Portal by clicking on the button My Requested Assets.



My Requested Assets		Vizor			
Assets Requested By Me					
	ID	Requested Asset Type	Requested By	Required By Date	Required By Time
	352	Laptop	Employee User	--	--
	353	\\server\ISO_DOCUMENTS	Employee User	--	--
	354	Adobe Photoshop CS6	Employee User	--	--

From this area, users can also modify an existing request after submission. The user can select the specific request and change information like the required date or even cancelling the request.

 **My Requested Assets** ✕

Request Details

Requested Asset Type:	Laptop
Required By Date:	9/29/2016
Required By Time:	10:22:00 AM
Requested By:	Employee User
Progress:	Awaiting Approval - Level 1 

[View Asset Request Details...](#)

* This color indicates a required field

Save

Close

APPROVING REQUESTS

Once a Request for an asset is submitted, you may want to have it approved before fulfilling it.

To define if a particular asset type requires an approval and to configure the type of approval, go to Assets > Manage Asset Types and select the asset type you want to configure the approvals for. Click on the Life Cycle tab and click on Approval.

In this area you can define up to 3 levels of approvals, exceptions to the approvals and the minimum number of approvers.

Click on the diagram to configure the life cycle stages.

Request → Approval → Fulfillment → Provisioning → Use → De-provisioning

Showing Life Cycle Stage:

Approval Needed For:

Level 1 Approvers:

Level 2 Approvers:

Minimum Approvers Required:

No Approval Needed Level 1:

No Approval Needed Level 2:

Vizor supports an unlimited number of levels of approval. Please contact support if you need more than the default 3 levels.

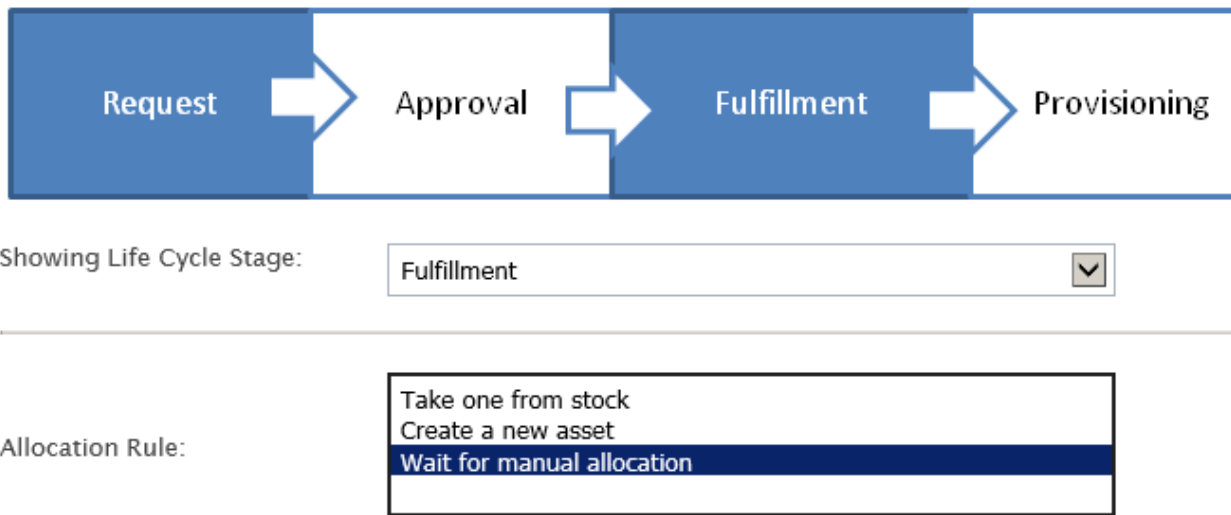
FULFILLING REQUESTS

Once a Request for an asset is approved, or if no approval is required, the request goes to Fulfillment. During the fulfillment stage an action to fulfill the request is created and assigned to a person or to a team.

The assigned person will get an email notification with a link to view the request.

Once the assigned person logs into Vizor, she will be able to see the request for asset fulfillment action inside **My Actions** area.

You can control how the fulfillment of the asset occurs in the Life Cycle tab of an asset type definition.



Configure the fulfillment by selecting how Vizor should choose the asset to assign to the person requesting the asset:

- **Take one from stock**

Vizor will take a random asset instance of the requested type from the pool of available assets in stock (storage) and assign it to the person the request is for.

Typical examples are software licenses, in which all instances in stock are essentially the same.

This method does not require a person to fulfill the asset and therefore the process moves directly to the provisioning stage.
- **Create a new asset**

Vizor will create a new asset instance of the requested type and assign it to the person the request is for.

Typical examples are Active Directory accounts, access to network shares, unlimited enterprise agreement licenses, etc.

This method does not require a person to fulfill the asset and therefore the process moves directly to the provisioning stage.
- **Wait for manual allocation**

Vizor will wait for a person to select what asset instance of the requested type to assign to the person the request is for.

Typical examples are computers and special hardware for which it is desirable to decide the specific item.

With this method, Vizor puts the request action in the status "Approved – Pending Fulfillment". It then requires the person assigned with the fulfillment action to go to **My Actions**, click on **Fulfill an Asset** and select the asset from the list before moving to the next provisioning stage.

When the fulfilling is manual, fulfilling the Asset from the request action is more efficient than allocating it in a different way because:

- You do not need to assign it to a person, it is automatically set to the person indicated in the request.
- You do not have to specify the return date if there is one, it is automatically set from the request.
- It automatically copies other requirements of the asset from the request into the actual provisioned asset.
- It automatically set the status of the selected asset to "Assigned".
- It only displays assets that are available and of the requested asset type.
- It changes the status of the request action to "Completed".

ADVANCED NOTES ON THE REQUEST-FULFILLMENT PROCESS

Requesting an Asset from the Request Portal creates:

1. An **Asset Specification** (Asset with UsageType = "Asset Specification")
 - a. Submitted Date & Time are set to the request time.
 - b. The Contact (originator) and Submitter fields default to the logged in user, unless requesting for someone else, in which case Contact (originator) will be different.
 - c. State=Open
 - d. Asset_Status is set to Requested
 - e. Asset_Sub_status is set to Requested
 - f. The asset specification is an instance copied from the asset type template, e.g. Laptop.
2. An **Action** for requesting, approving and fulfilling the asset
 - a. Submitted Date & Time are set to the request time.
 - b. Contact (originator) and Submitter default to the logged in user

- c. "Requested By" defaults to the logged in user
- d. State=Open
- e. Substate is set to Awaiting Approval 1
- f. Action Type= Request Asset
- g. Event Type= Request
- h. UsageType= Employee

The action is used for the process of requesting, approving and fulfilling the asset.

The **Asset Specification** contains the needs of the requester, which if approved will eventually be fulfilled with an asset instance.

Once Approved, the Action is passed to the Action Owner for fulfillment.

The owner of the action sees the fulfillment action in "My Actions" and the Fulfillment tab is visible.

After the fulfiller selects an available asset:

- The following fields are copied from the Asset Specification to the selected Asset Instance:
 - o Owner
 - o All the fields from the Visible Request Fields list for that asset type
- The following fields are set on the allocated asset instance
 - o The status is set to "Assigned"
 - o The Allocated to Person (Owner) is set with the value of Requested For or Contact

CHAPTER 7 - SOFTWARE LICENSES

Vizor helps you manage Software Licenses associated to Software Assets.

Each Software Licenses must be associated to a Software Assets, such as Microsoft Office.

If you plan to include multiple versions in the same license, do not add the version to the name your Software Asset, e.g. "Microsoft Office". However, in some cases, the license is restricted to a specific version, in which case is appropriate to name the Software Asset with the version, e.g. "Microsoft Office 2016".

In either case, you can further specify the Version, Edition and Language of the software covered by the Software License record.

EXPIRATIONS AND RENEWALS

You can get reminders via email when software licenses or subscriptions are about to expire or to be renewed.

By default, email notifications are sent 31 days before the End Date of a Purchase associated to a Software License. Emails are sent to the Contact and the Business Owner of the Software License in question, as designated in the Contact and Business Owner fields. The Software License must be active and the Purchase not Permanent (the duration) for the notification to be sent.

You can change the default time interval (31 days) and the recipients (Contact and the Business Owner) in the Notification Editor.

You can also create additional email notifications to meet the needs of your organization.

LICENSE COMPLIANCE STATE

License compliance is performed automatically by Vizor on a regular basis (configurable).

Each Software License can have a Compliance State, which can be one of the following:

Compliant – Optimal	Less licenses are used than entitlements, within a configurable threshold (%) that accommodates for near future growth
----------------------------	--

Non-Compliant	More licenses used than entitlements
----------------------	--------------------------------------

Compliant - Over Licensed	Less licenses are used than entitlements, but there are many unused licenses, more than the expected near future growth
----------------------------------	---

N/A	For the cases where either our calculations cannot set the compliance or the concept does not apply
------------	---

A software license can be Compliant but exceeding a threshold (%). The threshold is

- Defined per software license
- The suggested action would be to sell licenses or renew fewer in the next contract with the vendor
- You can also define the threshold based on employee hires/terminations/job changes

LICENSING CALCULATIONS

The license calculation will:

- Calculate the total number of licenses used, according to the specified method
- It will take into account matching entitlements for:

**For each allocated User or
Computer, the field...**

Needs to Match the field in SW License
record...

Allocated to Organization

Entitled to Organization

Entitled to Location

Entitled to Department

Entitled to Project

Licensing Calculations (choice list)

Name

**Function Name
(Customizable)**

**One License = One
Computer installation**

Calculate...

**One License = One Named
User**

One License = One

Computer CPU

One License = X Computer
CPU Cores

FIELDS

Some important fields of the Software License record are:

	Data Type	Choice List	Notes
Entitled Licenses / Number of Licenses	Single-Choice		
Licensing Model	Single-Choice		
Licensing Calculation	Single-choice		
Licenses/Seats Installed or in Use			Set by the licensing calculation

Last Licensing Calculation

Timestamp of last time the licensing calculation process executed

License Calculation
Parameter

Over-Licensed Threshold

Allocated to Organization

Single-Choice

Companies

(USERS.tblUser
WHERE ...)

Allocated to Location

Single-Choice

Locations

(Roles.tblIDts
WHERE ...)

Allocated to Department

Single-Choice

Departments

(Roles.tblIDts
WHERE ...)

Allocated to Project

Single-

Projects

Choice	(Roles.tblDts WHERE ...)
--------	-----------------------------

There are also other fields that belong to different entities and are used in the context of Software Licenses. For example, the following are fields of an Asset, a Computer Asset specifically, and are used for licensing calculations:

- CPU Count
- CPU Cores Count
- Virtual Machine

LICENSE PURCHASE CALCULATIONS

Vizor allows you to track your licenses purchases, renewals, upgrades and other agreements and automatically keep your licenses calculations up to date.

Vizor calculates these whenever a purchase is added, updated or removed in the Purchases tab.

- Number of Licenses are calculated using only the Active purchases.
- Costs over time (total cost, licenses costs, maintenance costs, and services costs) are calculated using the total of all the purchases, active or not.
- Average costs per unit (avg total cost, avg license cost, and avg maintenance cost) are calculated using just the active purchases.

CHAPTER 8 - SOFTWARE INSTALLATIONS

Vizor helps you manage the installations of software across your computers, for the software assets that are important to you.

A Software Asset is typically associated to multiple installations, so you can see where and how that software has been deployed. You can also join installations of different software titles or packages as one Software Asset Type. In many cases, there are many software installations of different versions, editions and minor builds found by the Discovery layer that represent the same Software Asset type.

In order to do this, Vizor synchronizes with your Microsoft System Center Configuration manager (SCCM), or Vector's own Inventory and Discovery component, and it automatically brings the relevant software inventory information discovered by them.

Vizor allows you to focus on the software that is important to you by:

- Viewing only the Software Installations that you need to manage.

You may need to manage specific software because of costs, size of deployment, license compliance, etc. About 80% of the installations found by discovery tools are not useful to be managed by organizations.

- Normalizing the Software Installations into Software Assets
Normalizing or grouping all the multiple different installations and flavours of a software title as one helps you be more efficient and optimize its use.

Allocations are normally managed by Vizor using approval workflows and the App/Asset Request Portal. They represent the “should be”.

Installations are found by a discovery layer, such as Vector’s own or Microsoft SCCM, and represent what “it is found to be”.

Part of the optimization process is to ensure the Software “found to be” is closest to the “should be”.

There are some reasons why the allocations may not be the same as the installations for a given software license:

- The allocation has not been defined yet for that software.
This is especially true at the beginning of using Vizor. As new requests and approvals for software are processed, Vizor automatically handles them as allocations (the “should be”), but at the beginning allocations would be empty.

Note: You can also import the allocations if you have the allocations or policies already defined elsewhere.

- The provisioning is not complete.
Maybe Vizor has just allocated the license to the user or computer, and therefore it is reserved, but it is not installed yet. The actual installation can be automated via Vizor and Microsoft SCCM, AD GPOs or other software distribution tools.
- There has been installations performed outside Vizor.
You want to know if it happens. Ensure you use Vizor email notifications to alert you of this situation.
- The de-provisioning is not complete.
Similar to deploying software, it is possible that Vizor has just removed the license from the user or computer, as when an employee leaves, but the uninstallation is in progress.
- The Discovery of Installed Software is not up-to-date.
The discovery tool may not be fully updated yet. This may or may not be a problem to attend. For example, if the discovery tool is configured to update the software inventory every 2 days, then it is expected to have up to a 2 day delay. This situation should be temporary.

SOFTWARE LICENSE KEYS DISCOVERY

Vizor integrates with third party tools such as Recover Keys that discover software keys installed in computers. You can view the discovered keys in Vizor in the Software Installations area. Each discovered key is automatically assigned to the computer installation of the software title, which may have version and edition information.

Discovered keys are 'found' keys, and are different from 'should be' keys. That is, the keys used to install software in your network may not be the same keys you have as part of your entitlements. Entitlement keys are stored in the Keys tab in the Software License record (Software Licenses).

You can easily check what software titles you have discovered keys for by running the query **License Key Discovered** (set the Summary List Layout to **Key**) or any of the reports **Discovered License Keys by Application** and **Discovered License Keys by Computer**.

You can also view the discovered keys in the Installations tab of the Software License record and the definition of the Software Asset type.

INTEGRATION WITH RECOVERKEYS

What types of keys are integrated into Vizor?

All the keys, serial numbers and license keys found by the tool RecoverKeys are imported into Vizor and associated to a Software Installation.

CHAPTER 9 - APPROVALS

Approvals can be used during various processes in Vizor. You can approve a Request for an Asset or a Service, you use approvals before a Change Request. You can also approve the request for onboarding a new employee or a new account in your organization.

You can set an approval workflow for each individual type of asset, and you can also have different approvers depending on the requestor.

You can also set multiple levels of approval. Multiple levels of approvals are used when you only want certain users to approve a request after specific users have already approved it. For example, you may want a Director to approve a request but only after it has been previously validated and approved by a Manager.

Vizor comes with three levels out of the box, but you can add any number of levels of approval to a given process.

DYNAMIC APPROVERS

There are some built-in macros that can be used for implementing dynamic approvers.

<EMPLOYEE MANAGER>

This person is the manager of the user for which the request is submitted for.

<EMPLOYEE SUPERVISOR>

This person is the supervisor of the user for which the request is submitted for.

<ASSET RESPONSIBLE>

This person is the person responsible of the asset type being requested. The name of the person with this role is indicated by the field Responsible in the details of the specific asset type.

These macros can make approvers be conditional by department or by the work team. For example, you may only want HR Manager of Department A to approve assets assigned to Department A employees, and may only want HR Manager of Department B to approve assets assigned to Department B employees.

APPROVAL EXCEPTIONS

You can add exceptions to specific levels of approvals for certain users. For instance, if all requests require a manager to approve, you may want managers to not require that level of approval. To define these exceptions, set the list of user groups where no approval is needed.

DELEGATES

Vizor has the ability to grant another employee (or delegate) the ability to make approvals on your behalf.

You can do so by logging into the Employees web view and editing the employee you want to give approval rights to. In the Overview tab, find the "Delegate For" field and click the "Make XX My Delegate" button. This should add your name to the field. The employee will now be able to see any requests awaiting your approval.

NOTE:

We currently don't have a way to change delegation for users other than the one you're currently logged in as. So, in order to change the delegations for anyone that may have left the organization, you will need to do the following:

- Login to ConnectAdmin
- Click on Security tab -> Users button
- Find the user you want to change delegation for and click on them
- Update the "Authenticate With" field to use "Vizor Authentication"
- Update the Password for the user (may need to save changes first before the fields will show up)
- Save the user
- Login to the Connect views and make the necessary changes to delegation

CHAPTER 10 - ADMINISTERING VIZOR

CREATING PROJECTS AND GENERATING WEB VIEWS

A Vizor project includes:

- A database of issues (and their revision histories).
- A database of definitions, which includes the queries, sorts, layouts, reports, notifications, fields, and tabs

available in the project.

Each project can have multiple Web-based views.

Each Web view is customized to meet the needs of a class of users. For example, employees only need to see the issues they submitted; not all the issues in the project database. On the other hand, support staff need access to all the issue information in the project database.

With Vizor, you can provide separate views for employees and for help desk staff. Each view is accessed through the same Web-based interface.

Each Web view is customized to meet the needs of a class of users. For example, customers on a beta list don't need to see all the issues in a project, just the ones submitted by the beta list. And customers don't need to see any fix-related information added to the issues by the development staff. Internal development staff, on the other hand, needs access to all the issue information in the project.

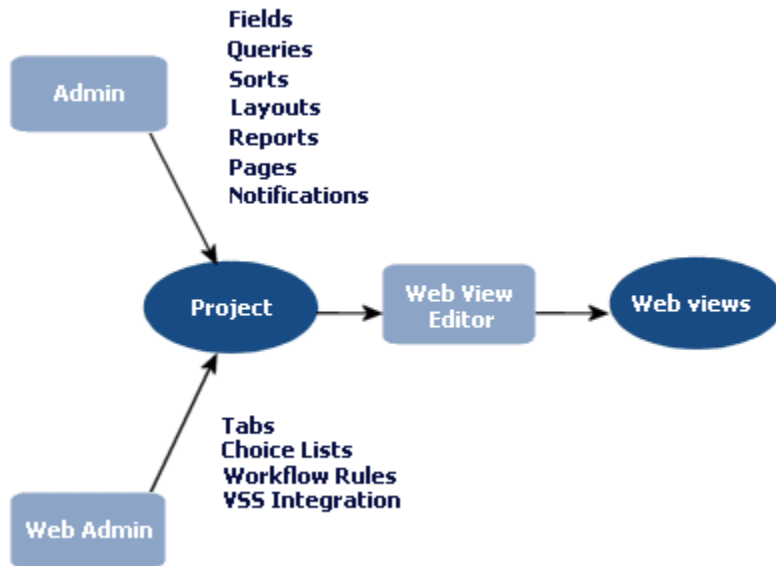
With Vizor, you can provide separate views for external customers and for internal staff. Each view is accessed through the same web-based interface.

Views are easy to define using the Web view Editor. Just choose the queries and fields that you want users to see:

- Queries determine which issues users can retrieve from the project database.
- Fields determine what information users can view and modify for each issue.

To create projects and edit definitions (of fields, queries, reports, sorts, layouts, and notifications), use Vizor Admin, a Windows program. Vizor Web Admin is a Web-based tool that allows you to remotely edit some of the project definitions (such as tabs and choice lists).

ADMINISTERING PROJECTS AND WEB VIEWS



RELATED TOPICS

[Creating New Projects](#)

[Editing Fields](#)

[Defining Reports](#)

[Defining Queries](#)

[Defining Sorts](#)

[Defining Layouts](#)

[About Workflow Rules](#)

[Working with Tabs](#)

[Editing Choice Tables](#)

[Setting Up Notifications](#)

[Creating Web Views](#)

[Vizor Admin](#)

[Web View Editor](#)

[Vizor Web Admin](#)

MANAGING USERS, CONTACTS, AND GROUPS

USERS

All users must have a Vizor user account, which consists of a user name and an optional password. Use Vizor Web Admin to create and manage user accounts.

GROUPS

By assigning users to groups, you can control:

- Access to features in Vizor Admin, Vizor Web Admin, and Web views.
- Access to projects.
- Access to Web views.
- Workflow permissions.

Use Vizor Web Admin to create groups, edit group privileges, and assign users to groups.

CONTACTS

A contact is a person who reports an issue. For example, a guest from another company may report an issue with a computer while they are working on-site. In this case, the guest is the contact, and the help desk analyst who enters the report is the submitter.

To create and manage a list of contacts, use Vizor Web Admin. Web views can also include a Contact button that opens the Users and Contacts dialog. Access to this dialog from a Web view is controlled by the group permissions defined in Vizor Web Admin.

Vizor provides a set of tools for administering projects and Web views:

- Vizor Admin for administering projects.
- Web View Editor for defining and generating Web views for projects.
- Vizor Web Admin for administering projects and users through the Web.

RELATED TOPICS

[Defining User Groups](#)

[Creating User Accounts](#)

[Adding and Removing Users](#)

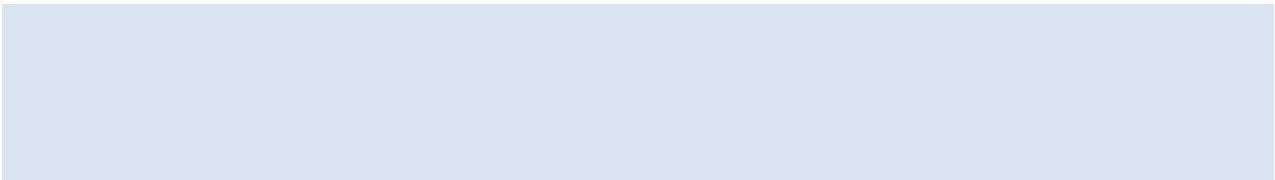
[Defining Conditions based on User Groups](#)

[Setting Group Access Permissions for Web views](#)

[Vizor Admin](#)

[Web View Editor](#)

[Vizor Web Admin](#)



ACTIVE DIRECTORY AND EMAIL INTEGRATION

EMAIL INTEGRATION

Vizor email integration requires both incoming and outgoing settings to be completed.

INCOMING SETTINGS

Access Vizor Web Admin via a link such as: **<https://mycompany.vizor.cloud/connectadmin/>**.

If a subdomain is not yet set up for your Vizor server, the link will be an IP address such as **<http://10.2.123.45/connectadmin/>**

Login using an account that has access to admin features. In most cases, the user will be:

Username: admin

Password: admin

This account is the default one available during initial configuration, but you may use any other administrator account. It is suggested to change the password of the admin user once all the configuration is completed.

Click the Email tab (Figure 1.1).

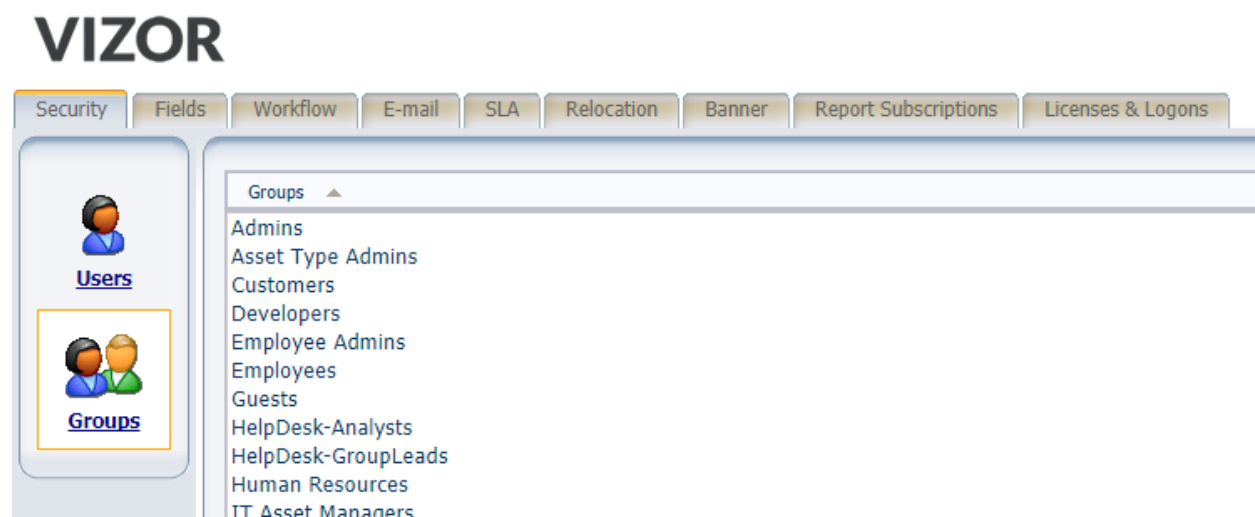


Figure 1.1

From the dropdown list, select the HelpDesk project (Figure 1.2).

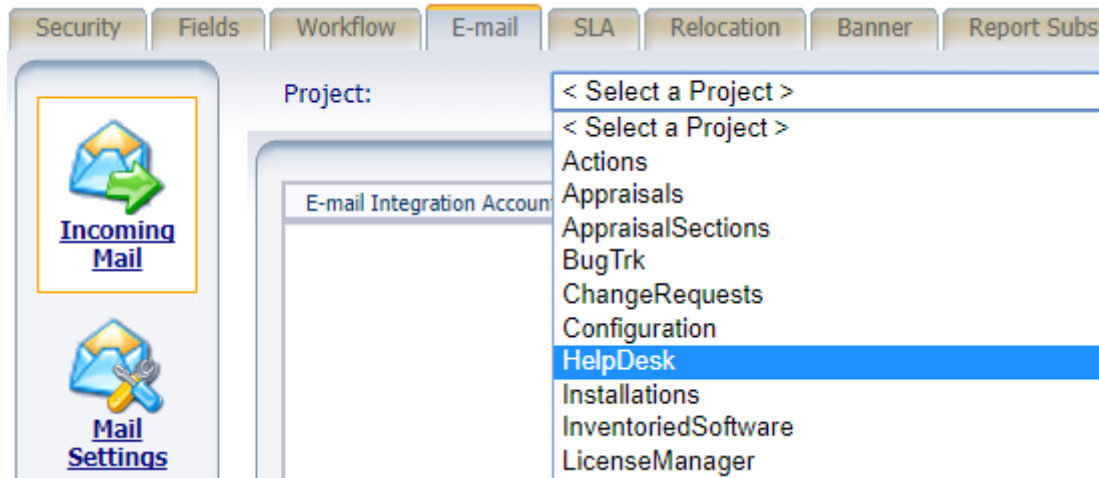


Figure 1.2

On the far right, click Add a new Integration. A window will appear with multiple fields to fill out (Figure 1.3).

Add E-mail Integration Account

Incoming E-mail Server (POP)
Vizor can be configured to monitor specific E-mail addresses and automatically create issues from the e-mails sent to it. Enter the details for an address you want Vizor to monitor.

Account Information
E-mail Address:

Server Information
Hostname:
Port: SSL: ☐

Logon Information
Logon:
Password:

Mail Handling
Mail-handling rules determine how messages sent to this account are converted to issues. The rules are processed in order. If all the conditions of the rule are met, the actions of the rule are executed, and no more rules are evaluated. If all conditions are not met, the next rule is evaluated.

Rule Name	Action Type
Send Status Report	None
Send Undeliverable to Queue	Queue
Link to Issue	Link
Create new Issue	Create
Add E-mail to Queue	Queue

Issue Completion
Issue-completion rules determine the values that will be given to each field in an issue. Click Edit Mapping to specify the issue fields that can be copied from the e-mail. Click Edit Defaults to specify default values for non-mapped fields.

☐ Enable this integration.

Figure 1.3

E-Mail Address: The e-mail address of this monitored account, such as **support@company.com**. It will appear by default as the “From” address of automated and manual replies to process e-mails.

Hostname: The hostname or IP address of the POP server.

Port: The port through which the POP server communicates. The default is 110. The default SSL port is 995. SSL: If the POP server requires SSL, enable this by checking the box.

Logon: The logon of the account used to access the POP server.

Password: The password of the account used to access the POP server.

Once all the information is entered, click on Test (Next to the logon and password fields). If the settings are correct, you will get a confirmation message as a browser alert. Otherwise, you will be alerted of an error, meaning one of the fields in the settings are incorrect.

The **Mail Handling** and **Issue Completion** sections of this window are typically done by the main contact of your Vizor implementation project and are to be discussed, so you may skip these parts and click OK in the bottom right. The Integration will be disabled by default, and it is turned on once configuration is complete, and you are ready to go live with your new Vizor system.

OUTGOING SETTINGS

Click on **Mail Settings** (Figure 1.2). The SMTP settings are usually the same the incoming settings. You can test these outgoing settings by clicking the Test button (Under the password field) and entering an email address to which a test email will be sent to verify the connection (Figure 2.1). Like the incoming settings, you will see a success or error message depending on the validity of your settings.

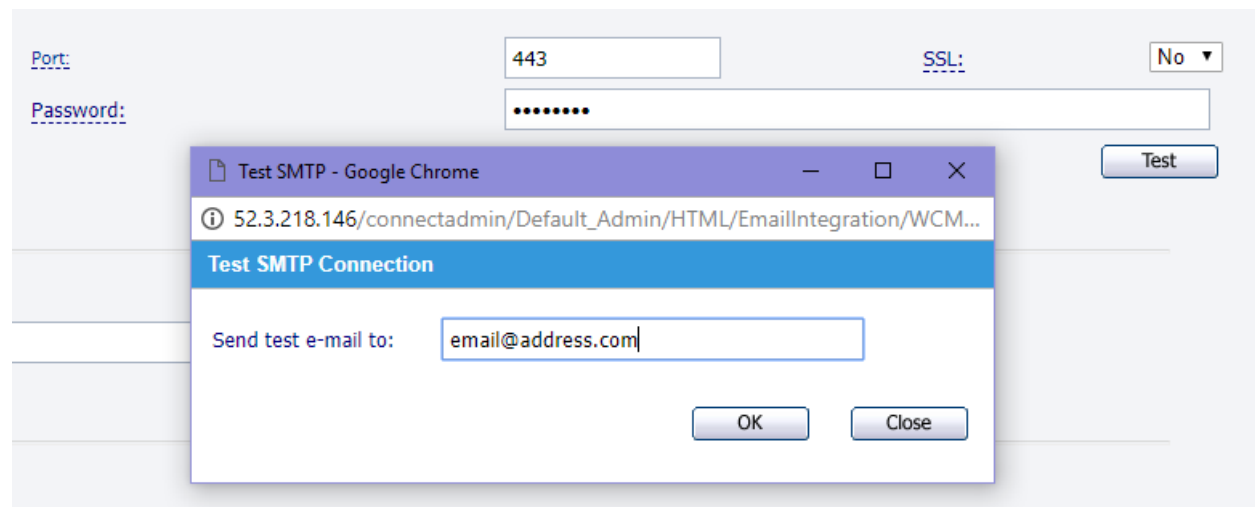
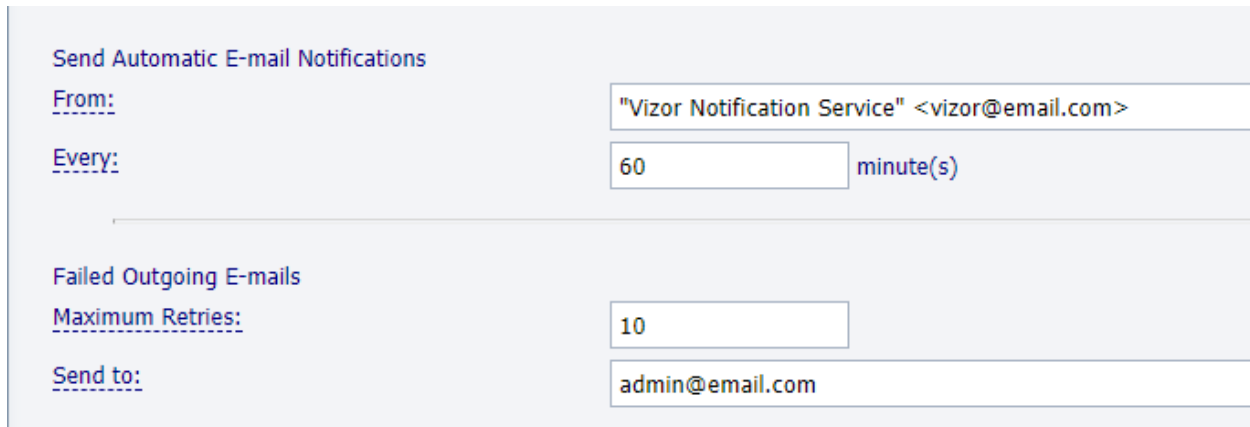


Figure 2.1

In the next section under the SMTP settings, add the email addresses that will be used (Figure 2.2).



The screenshot shows a configuration window for SMTP settings. It is divided into two main sections. The first section, titled "Send Automatic E-mail Notifications", contains three fields: "From:" with the value "Vizor Notification Service" <vizor@email.com>, "Every:" with the value "60" and the unit "minute(s)", and "Failed Outgoing E-mails". The second section, titled "Failed Outgoing E-mails", contains two fields: "Maximum Retries:" with the value "10" and "Send to:" with the value "admin@email.com".

Figure 2.2

The first email address, just like the incoming settings, is used in the "From" of emails being sent. Also, emails can be sent at specific intervals. In this example, it would be every hour. The second email address is used for failed outgoing emails. Typically, an administrator would receive the notification. A maximum number of retries can be set, meaning the amount of tries before the failing email sends an error notification to the administrator. Once all the settings are filled out, click Apply in the bottom right. Your Vizor email integration is now complete.

ACTIVE DIRECTORY INTEGRATION

The Active Directory integration will require a remote desktop connection to the server. You may do so by using the Windows desktop app (Figure 3.1). Enter the IP address of the server along with a username and password. These should be provided to you by your main contact for the implementation project.

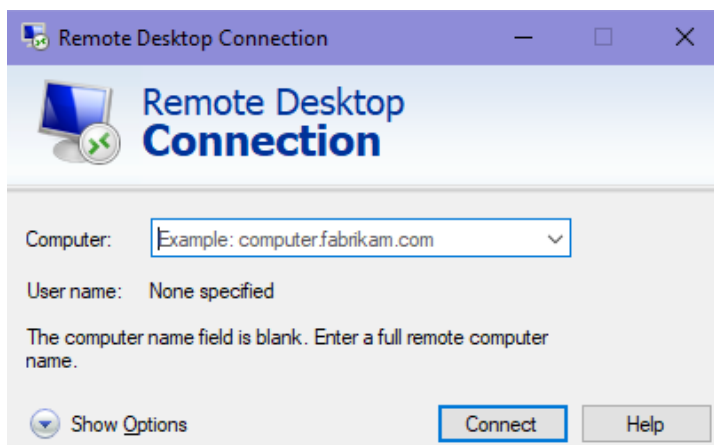


Figure 3.1

Once you have access to the server, click Start, then select **VIZOR Import Manager** (Figure 3.2). You will be prompted to enter credentials. Enter the credentials of an administrator such as the default admin account.

Username: admin

Password: admin

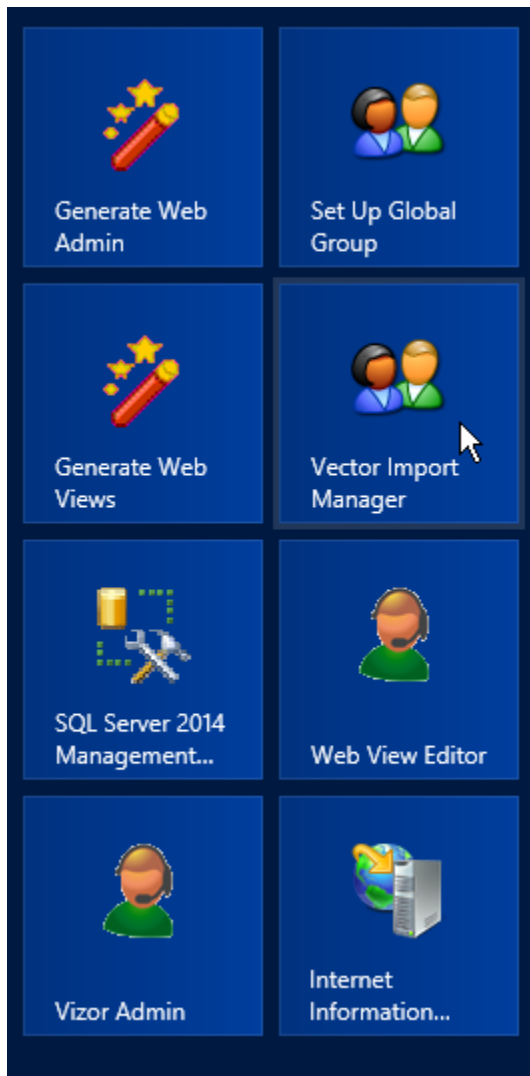


Figure 3.2

A list of imports will be shown. Select Import Users from AD, then click Edit (Figure 3.3). Click Edit again in the next window.

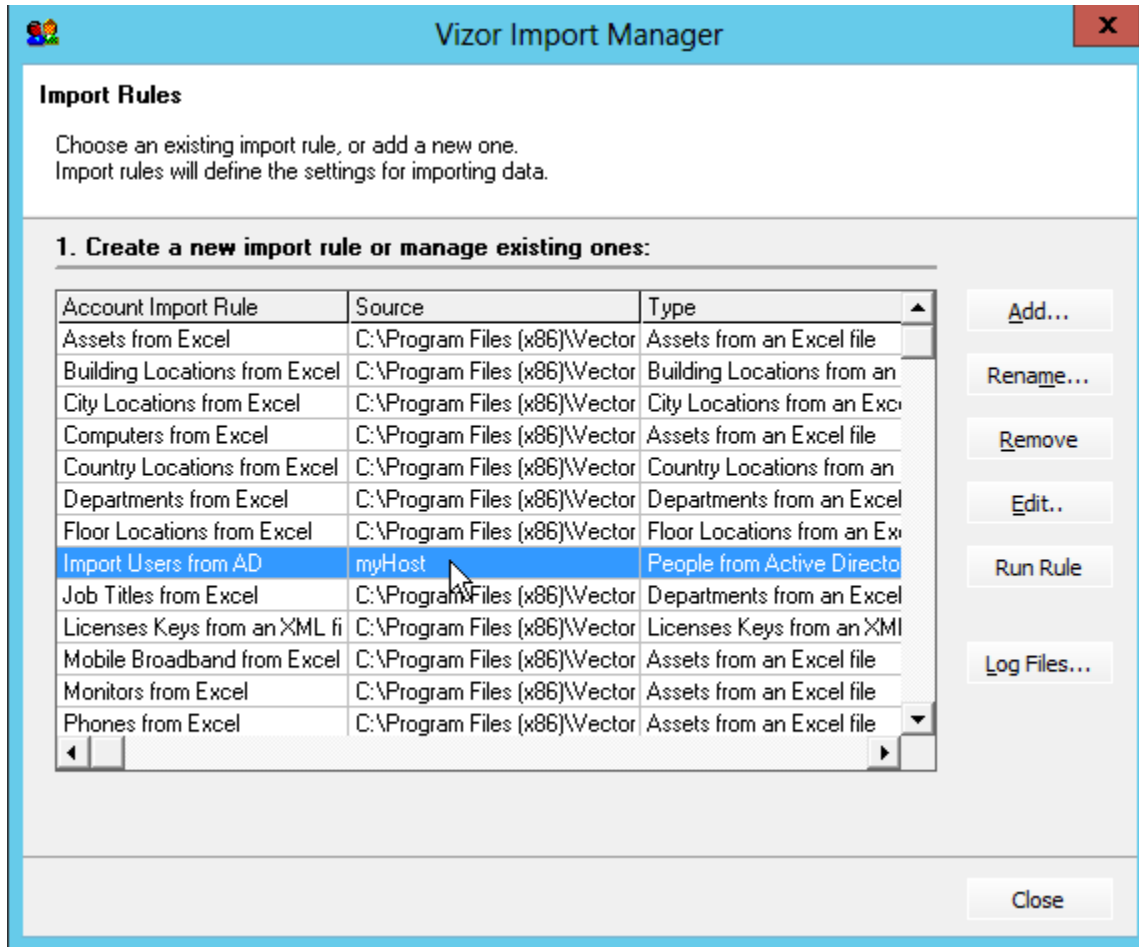
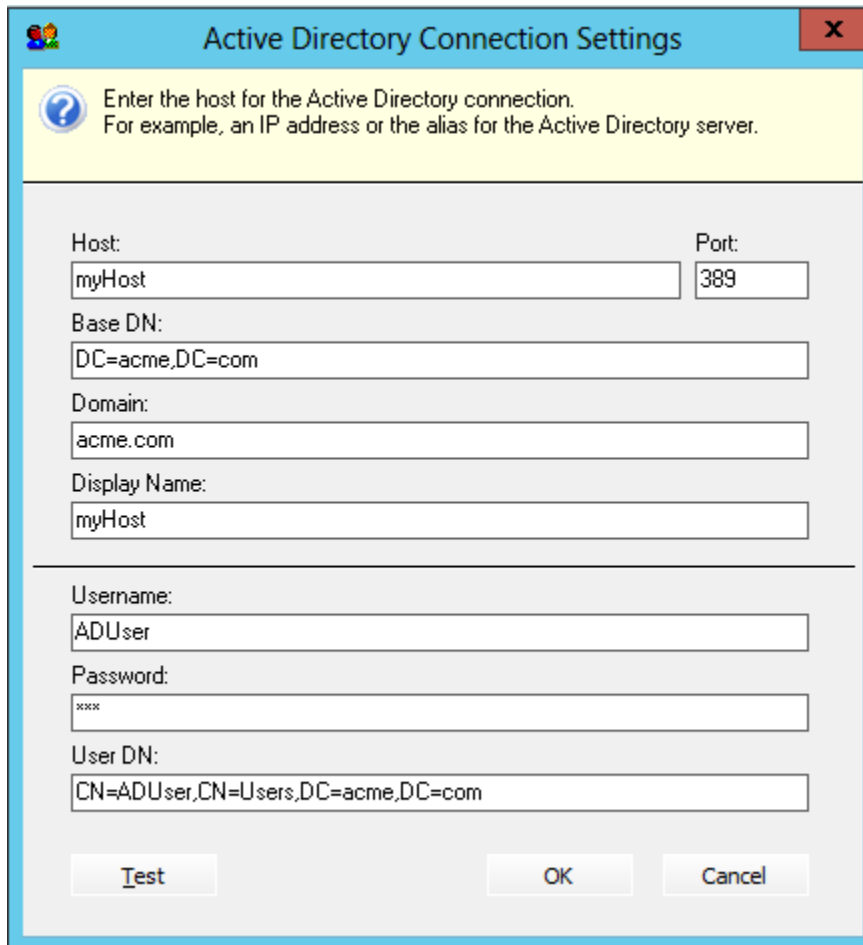


Figure 3.3

Clicking Edit the second time will display a window with fields to fill in (Figure 3.4). Sample values have been provided in the figure.



The image shows a Windows-style dialog box titled "Active Directory Connection Settings". It has a blue header bar with a question mark icon on the left and a red close button (X) on the right. Below the header is a yellow instruction box that says: "Enter the host for the Active Directory connection. For example, an IP address or the alias for the Active Directory server." The main area of the dialog contains several text input fields arranged in two sections. The first section includes fields for "Host:" (containing "myHost"), "Port:" (containing "389"), "Base DN:" (containing "DC=acme,DC=com"), "Domain:" (containing "acme.com"), and "Display Name:" (containing "myHost"). The second section includes fields for "Username:" (containing "ADUser"), "Password:" (containing "xxxx"), and "User DN:" (containing "CN=ADUser,CN=Users,DC=acme,DC=com"). At the bottom of the dialog are three buttons: "Test", "OK", and "Cancel".

Field	Value
Host:	myHost
Port:	389
Base DN:	DC=acme,DC=com
Domain:	acme.com
Display Name:	myHost
Username:	ADUser
Password:	xxxx
User DN:	CN=ADUser,CN=Users,DC=acme,DC=com

Figure 3.4

After filling in these fields, click Test to verify your settings. If your settings are correct or incorrect, you will get an appropriate alert. If your settings are correct but you still receive an error, it is possibly a firewall issue. In that case, you may need to contact your network administrator to solve the issue.

If the test was successful, click OK to close the settings window, then click OK again to go back to the list of imports (Figure 1.3). To run the AD rule that was just edited, click Run Rule on the left side under the Edit button. Your users from AD will now be imported into Vizor.

VIZOR ADMIN

VIZOR Admin is an application for administering Vizor projects. For example, you can use it to:

- Create projects
- Customize fields
- Define queries, sorts, and layouts
- Define reports
- Define and set up e-mail notifications

To start Vizor Admin:

On the **Start** menu, select **Programs, Vector, Vizor, Vizor Admin**.

On the **Start** menu, point to **Programs**, click **Vizor**, and then click **Vizor Admin**.

RELATED TOPICS

[Creating New Projects](#)

[Editing Fields](#)

[Defining Reports](#)

[Defining Queries](#)

[Defining Sorts](#)

[Defining Layouts](#)

WEB VIEW EDITOR

Web View Editor is a Microsoft Windows application that creates Web views for your Vizor projects. Web views are Web-based applications that provide access to projects through a Web browser.

To start the Web View Editor from Vizor Admin:

On the **Tools** menu, click **Web** and then click **Web View Editor**.

The **Tools > Web > Web View Editor** command is available only when the Web View editor is installed on the computer running Vizor Admin.

To start the Web View Editor from the Start menu:

On the **Start** menu, select **Programs, Vector, Vizor, Web View Editor**.

On the **Start** menu, point to **Programs**, click **Vizor**, and then click **Web View Editor**.

Notes

- Vizor Admin and Web View Editor are installed on the Web server, by default. You can install additional copies of these tools on other computers, so you can perform administration tasks locally. However, to create new Web views, you must use the editor on the Web server.

RELATED TOPICS

[Creating Web Views](#)

VIZOR WEB ADMIN

Vizor Web Admin is a Web-based administration tool. It allows you to perform the following administrative tasks over the Web:

- Manager users and user groups, and define group privileges (access to features and projects)
- Add and edit contacts
- Define workflow rules
- Create new tabs
- Set up choice fields
- Enable e-mail notifications
- Migrate projects to SQL Server and move projects to different SQL Server computers
- Move attachments to a different computer

To logon to Vizor Web Admin, choose one of the following methods:

- In your Web browser, go to **//server/vitadmin**, where **server** is the name of your Web server.
- In Vizor Admin, click **Tools > Web > Web Admin**.
- At the Web server, click **Start > Programs > Vector > Vizor > Web AdminStart > Programs > Vector > Vizor > Vizor Web Admin**.

RELATED TOPICS

[Administering Vizor on the Web](#)

[Troubleshooting Vizor Web Admin](#)

[About Workflow Rules](#)

[Working with Tabs](#)

[Editing Choice Tables](#)

[Setting Up Notifications](#)

[Relocating Databases to SQL Server](#)

[Relocating Attachments](#)

TROUBLESHOOTING VIZOR WEB ADMIN

If users cannot successfully relocate databases or perform other tasks with Vizor Web Admin, check that there is a global group named YOURDOMAIN\CensusAdminsGroup, and that the user (especially if the user is a domain user, not a domain administrator) is a member of this group.

Vizor Web Admin uses your Windows credentials to run tasks if you are a member of the CensusAdminsGroup group or the local Administrators group on the Vizor Server computer. Otherwise, Vizor Web Admin runs tasks as CensusUser.

Members of the CensusAdminsGroup group (or of the local Administrators group on the Vizor Server computer) have the required permissions to perform any task with Vizor Web Admin. The CensusUser account, on the other hand, cannot perform IIS-related tasks such as unloading virtual directories (when logging off users), and may not be able to relocate databases to other computers.

During installation, the Setup program creates the CensusAdminsGroup group and adds all members of the domain Administrators group to the CensusAdminsGroup group. Setup also adds the CensusAdminsGroup to the local Administrators group on the Vizor Server computer. The purpose of the CensusAdminsGroup is to control access to Web Admin.

If Setup was unable to create the CensusAdminsGroup, you can create it manually.

For example, if you want to allow a domain user that is not a domain administrator (because domain administrators already belong to the local Admins group by default) to use the Relocation tab, you can create the CensusAdminsGroup and add the domain user to the group. This allows the domain user to perform tasks that are typically only available to local admins:

- Managing SQL Server databases and physical files.
- Logging off Vizor users (which requires permissions to control IIS).

TO MANUALLY CREATE THE CENSUSADMINSGROUP AFTER VIZOR IS INSTALLED:

- 1 Create a global group named **CensusAdminsGroup** in the domain.
- 2 Add CensusAdminsGroup as a member of the local Administrators group of the Web server.
- 3 On the **Start** menu, click **All Programs > Vector > Vizor > Vizor Tools > Set Up Global Group.All Programs > Vector > Vizor > Vizor Tools > Set Up Global Group.**

CHAPTER 11 - SEARCHING AND QUERYING

ABOUT QUERIES

To build queries in Vizor Admin, select the Project you want to create a query for on the left. Click **Query Editor** on the **Project** menu. In the Query editor, use **And** and **Or** operators to combine multiple search conditions. You can control the order in which search conditions are evaluated using parentheses.

The standard format of a condition is:

Field	Test	Value
-------	------	-------

For example:

NOTES

- To base a new query on an existing query, choose an entry in the **Query Name** list, click **Copy**, and name the new query.

SEARCHING BY DATES

Use the **<Today>** value to match the current date. For example, you can search for all issues submitted or updated today.

To search against specific dates, double-click the **Value** box, then double-click a date in the calendar.

SEARCHING FOR UNASSIGNED ASSETS OR ISSUES

By default, the **Owner** field is set to **<None>**. You can use the value **<None>** in queries to find unassigned issues.

FINDING ASSETS OR ISSUES OWNED BY THE CURRENT USER

Use the value **<User>** to match the current user. With **<User>**, you can build generic queries that work for any user. For example, the **My Assigned Issues** query finds all issues assigned to the current user.

FINDING NEW ASSETS OR ISSUES

To find all issues that have not been updated since they were first submitted, search against the revision number: **Revision Number = 1**.

FINDING PARENT ISSUES

You can find the parent issues of a child issue by searching against the **Child Issues** field. For example, to find the parent issues of the issue 22, use the Advanced Query editor to search for **Child Issues = 22**.

To find all parent issues, use the Advanced query editor to search for **Child Issues <> 0**.

SEARCHING CHOICES LISTS WITH RELATIONAL OPERATORS

The relational operators are <, <=, >=, and >. When you search choice lists with a relational operator, the search is based on the order of the choices in the list.

For example, if the choices in the **Problem Area** list are ordered:

```
Apps - Goldmine
Apps - MS Excel
Apps - MS Outlook
Apps - MS Word
Apps - Other
HW - Disk
...
```

the query **Problem Area < HW - Disk** returns all the application-related issues.

When a choice list uses a non-alphabetical sort order, the sense of the relational operators is reversed. For example, if the **Priority** list is ordered:

```
Highest
High
Medium
Low
Lowest
```

the query **Priority > Medium** returns issues marked **High** or **Highest**, and the query **Priority < Medium** returns issues marked **Low** or **Lowest**.

SAVING QUERIES

To save changes and continue editing, click **Apply**.

To save changes and exit the Query editor, click **Close**.

INSERTING PARENTHESES AROUND SEARCH CONDITIONS

When you combine search conditions with the **And** and **Or** operators, you may not always get the results you expect. For example, suppose you want to find all new issues with priorities **ASAP** or **Today**.

```
Progress = New And  
Priority = ASAP Or  
Priority = Today
```

And has a higher precedence than **Or**, so this query finds all new issues with priority **ASAP**, plus all issues (not just the new ones) with priority **Today**.

To find only new issues with priorities **ASAP** and **Today**, click the **Priority = Today** search condition, then click  to place parentheses around the conditions joined by **Or**. For example:

```
Progress = New And  
(Priority = ASAP Or  
Priority = Today)
```

SEARCHING FOR STRINGS

The **Contains** operator performs a case-insensitive search for a string of text anywhere in a field. You can use **Contains** to search for keywords in the **Summary** or **Description** fields. This allows you to package common keyword searches such as predefined queries. For example:

```
Summary Contains printer
```

You can also use **Contains** to find all choices in a list that contain the same string of characters. For example, the choices in the default **Problem Area** list start with "HW - ", "Apps - ", or "System - ". So **Contains "HW -"** finds all hardware problems.

SEARCHING FOR NULL VALUES

It is possible to perform a search to determine whether or not a value has been set for a field. When a field has no value, it is considered to be **<Null>**; on the other hand, when a field has a value, it is **<Not Null>**.

The following steps describe how these macros can be used in queries:

- 1 Open the **Query Editor** for the desired project.
- 2 Create the query using **<Null>** and **<Not Null>** as desired (use **Like** for the **Test** column).

Example:

```
Priority Like <Null> AND Severity Like <Not Null>
```

NOTE: Searching for null values in *Date*, *Time*, and *Number* fields will require a few additional steps. Please contact support for more information.

- 3 Export the query to the Web view(s) and regenerate the view(s).

NOTE: Saved queries with **<Null>** and **<Not Null>** can only be used from the Web views, not from the Preview in the Admin editors.

It is also possible to use the **<Null>** and **<Not Null>** using the Ad-hoc Query editor of any given Web view. Simply select a Field, choose Like for the Test option, and type either **<Null>** or **<Not Null>** in the Value textbox.

SEARCHING WITH WILDCARDS

To provide flexibility in the specification of search patterns, the **Like** and **Not Like** operators support wildcard characters. Using wildcards, you can search for inexact patterns of text in any field.

Wildcard	Matches
_	Any single character. For example: run_time matches run time or run-time
%	Zero or more characters. For example: Contains run%time matches any string that contains runtime , run time , or run-time . Like Install% matches any string that starts with Install . Like %run%time% matches the same strings as Contains run%time .
[charlist]	Any single character in charlist. Can include spaces, but not the right bracket (]) character. For example: SP[56] matches SP5 or SP6 Use a hyphen to specify a range of characters. For example: [1-36-9] matches the digits 1, 2, 3, 6, 7, 8, or 9 To match the hyphen character, the hyphen must be either the first or last character in charlist. For example: [-0-9] or [0-9-] match any digit or a minus sign
[!charlist]	Any single character not in charlist. For example: [!tb]rash matches crash but not trash or brash

To search for the wildcard characters _ and %, you must enclose them in brackets. For example, to search for a percent sign, use the pattern [%].

USING MACROS IN QUERIES

A Macro in a query is part of a condition expression that is resolved at runtime, that is, its value is evaluated dynamically when executing the query.

For example, in a typical query condition expression:

Field	Operator	Value
Owner	=	Peter Parker

the value is fixed to the user Peter Parker. However, using a macro <User> in the value:

Field	Operator	Value
Owner	=	<User>

will cause the value to be the name of the person currently logged in to Vizor.

Most of the macros used in queries are used in the **Value** part of the condition but custom macros can be created for the **Field** or **Operator** parts as well.

<IS ACCESSIBLE DEPARTMENT>

This macro is used to filter items (assets, records, etc) based on the department of the current logged-in user. The macro is used in queries where the field part is a single choice with Choice values coming from Departments. For example:

Allocated to Department = <Is Accessible Department>

In the condition above, the query will return all records where the Allocated to Department is set to the same department as the one of the logged-in user, or any department if the user has the feature "Access all Departments" enabled.

The typical use of this macro is in named queries that are designated as the "base query" of a Web View.

For example, adding the condition expression

Allocated to Department = <Is Accessible Department>

to the base query of the Asset Library (Resources _ Library) Web view will ensure users can only see the assets allocated to the department they belong to, or any asset if they have enabled the feature "Access all Departments".

The macro <Is Accessible Department> distinguishes between two types of users:

1. Users who can view or manage items (assets, records, etc) of their own department only.
2. Users who can view or manage items (assets, records, etc) of any department, or no explicit department.

<IS ACCESSIBLE LOCATION>

This macro is used to filter items (assets, records, etc) based on the location of the currently logged-in user. The macro is used in queries where the field part is a single choice with Choice values coming from Locations. For example:

Allocated to Location = <Is Accessible Location>

In the condition above, the query will return all records where the Allocated to Location is set to the same location of the logged-in user or any location inside. That is, if the user location is "San Francisco", then items in locations "San Francisco \ Lombard St." and "San Francisco \ Bay St." will also be included.

Similar to <Is Accessible Department>, items in any location will be included in the results if the user has the feature "Access all Locations" enabled.

The typical use of this macro is in named queries that are designated as the "base query" of a Web View.

For example, adding the condition expression

Allocated to Location = <Is Accessible Location>

to the base query of the Asset Library (Resources _ Library) Web view will ensure users can only see the assets allocated to the Location they belong to, or any asset if they have enabled the feature "Access all Location".

The macro <Is Accessible Location> distinguishes between two types of users:

1. Users who can view or manage items (assets, records, etc) of their own Location only.
2. Users who can view or manage items (assets, records, etc) of any Location, or no explicit Location.

CHAPTER 12 - THE SUMMARY LIST

ABOUT SORTS

You can sort the Summary List by issue number, priority, state, progress, owner, or any other field or combination of fields. Vizor allows you to define sorts, which are named sets of sort criteria. Sorts appear in the Sort list on the Summary toolbar.

To define a sort in Vizor Admin:

- 1 Click **Sort Editor** on the **Project** menu to open the Sort editor.
- 2 In the Sort editor, choose the fields you want to use as sort keys. The first field is the primary sort key, the second field is the secondary key, and so on.

For example, if you sort by **Owner** and **Priority**, all issues belonging to the same person are grouped together, and the issues for each owner are sorted by priority.

To base a new sort on an existing sort, choose a sort from the **Name** list, click **Copy**, and name the new sort.

SORT ORDERS

Choice Order Ascending and **Choice Order Descending** sort the issues based on the order of the choices in the choice list, not on alphabetical order.

For example, a choice list may contain the values **High, Medium, Low** in that order. An alphabetical order for the sort list would be **High, Low, Medium** or **Medium, Low, High**, which doesn't make as much sense.

SAVING SORTS

To save changes and continue editing, click **Apply**.

To save changes and exit the Sort editor, click **Close**.

RELATED TOPICS

[Defining Sorts](#)

ABOUT LAYOUTS

Different queries can list different information in the **Summary List**. For example, a query that lists only the issues assigned to the user does not need to display the **Owner** column, unlike a query that lists all issues assigned to a group.

Vizor allows you to define layout styles that specify the column layout and contents of the **Summary List**. Layout styles appear in the **Layout** list on the **Summary Toolbar**.

To define a layout in Vizor Admin, click **Layout Editor** on the **Project** menu to open the **Layout Editor**.

Each column in the **Summary List** corresponds to a field. Choose the fields you want to list, and set the column widths and alignments. By default, field names are used as column headings, but you can change this by entering a new heading in the **Title** cell.

NOTES

To base a new layout on an existing layout, choose the layout from the **Name** list, click **Copy**, and name the new layout.

RELATED TOPICS

[Previewing Layouts](#)

[Saving Layouts](#)

[Exporting Layouts](#)

SAVING LAYOUTS

To save changes and continue editing, click **Apply**.

To save changes and exit the Layout editor, click **Close**.

RELATED TOPICS

[Defining Layouts](#)

[Previewing Layouts](#)

CHAPTER 13 - REPORTING

ABOUT REPORTS

Reports help you understand and assess the state of your project. A system like Vizor contains a lot of valuable information. Reports allow you to access this information, break it down, analyze it, and present it.

In addition to viewing the default reports included with Vizor, you can create new reports in Vizor Admin. To design reports in Vizor Admin, click **Reports** on the **Project** menu, and then click **Report Editor**.

VIEWING REPORTS

You can view reports inside Web views. After you click on a Web view, you can access the list of reports available to you in that Web view by clicking on the Reports menu item on the top toolbar.

By default, Web views use HTML to format and display Listing reports, and Crystal Reports for custom reports (Vizor includes the Crystal Reports Runtime Software).

Some reports have parameters that allow to customize the report at runtime. Depending on how the report is viewed, parameters can be accessible from the Report Viewer top menu, when using the Interactive Crystal Reports viewer, or prompted before viewing the report, as it is the case for PDF reports.

You can also view Summary, Time, and Listing reports in Vizor Admin, within the Report Editor.

TYPES OF REPORTS

In general, Vizor reports can be categorized as follows:

- Web or Listing Reports
 - Standard
 - Tabular
 - Multi-column
 - Custom
 - Vizor Query Report
 - Embedded Query Report
 - Standalone Report
- Summary and Time Reports

These reports are deprecated and can only be viewed in the Report Editor (Vizor Admin).

STANDARD REPORTS

Listing reports are listings of information entered in records, such as tickets, issues, assets, people, etc. Use Listing reports to extract specific information from records.

For example, you can use a Listing report to print a summary of the resolution details for each resolved issue, or to print only the issue number, owner, and summary for each issue. With Listing reports, you choose a set of issue fields, and only those fields are included in the report.

Standard reports are generated by Vizor without the use of a third party reporting engine. Standard reports are viewed in the Web UI as HTML reports.

QUERY

In the **Query** list, choose a query to extract the records from the database and include them in the report. For example, if you want a report that breaks down the open issue count, then use the Open Issues query.

SORT

Sorts the issues.

PAGE

Header, footer, and title to apply to the report.

TABULAR

Tabular listings list issues in the columns of a table. A tabular report is like the Summary List, except that you can use a report template to add things like a report title or page numbering.

Issue Number	Priority	Submitted Date	Submitted Time	Progress	Assigned Date
1	ASAP	12/11/2001	12:46:43 PM	Resolved	12/11/2001
2	This Week	12/11/2001	12:48:00 PM	In Progress	12/12/2001
3	This Week	12/11/2001	1:05:45 PM	Assigned	12/13/2001
4	ASAP	12/11/2001	4:36:33 PM	Assigned	12/13/2001
5	Today	12/11/2001	4:38:20 PM	New	
6	Today	12/11/2001	4:59:59 PM	New	

MULTICOLUMN

Multicolumn listings arrange issue information into blocks; these are repeated down the page for each issue.

Issue Number	1
Priority	ASAP
Submitted Date	12/11/2001
Submitted Time	12:46:43 PM
Progress	Resolved
Assigned Date	12/11/2001
Assigned Time	4:15:17 PM
Problem Area	System - Printing
Owner	Help Desk Analyst
Summary	Cannot print from my office PC.

Files

Standard reports do not have an associated file, they are generated at runtime based on the definition of the report, using the Vizor Admin > Report Editor (the definition of reports are stored in the DEF database).

CUSTOM REPORTS

Custom reports use Crystal Reports and can have features that are not available with Standard HTML reports. Using Crystal Reports, you can add features such as parameters, charts, formulas, field highlighting, and running totals to a listing report. You can also import graphics (such as company logos)

and completely reformat a listing report. To build custom reports, you need a version of Crystal Reports that is version 8.5 or beyond.

Custom listing reports appear in the same list together with Standard reports in the Vizor Web Report Viewer. After you customize a listing report, you cannot edit it in the Report Editor. You must use Crystal Reports.

Custom reports also have an additional option for viewing: the Interactive Report Viewer, which allows the user to zoom in/out, page and export to other formats.

Files

All custom reports have an associated rpt file that is not generated at runtime, it must be created in advance.

CREATING NEW CUSTOM REPORTS

While Vizor comes with numerous custom reports out of the box, it may be desirable to create additional custom reports. The process of creating such a report is as follows:

- 1 Open the **Report Editor** for the desired project.
- 2 Select the **Listing** tab.
- 3 Locate a report that has the **Custom Report** checkbox checked. If checked, this indicates that the report will appear in the **Custom** tab of **Reports** section of the Web views.
- 4 Click the **Copy Report** button. Choose a name for the new report, type it in, and press **OK**.
- 5 Press **Apply** to save your changes.

Note: After doing this, Vizor will have already saved two .rpt files in the **CustomizedFiles** folder allocated to this project. In order to view the reports in the Web views, they must be moved to the Web view folder. This can be achieved by regenerating the Web views that are related to the project.

Example:

CUSTOMIZEDFILES\#Project#HelpDesk\#AllWebViews#\

6 Regenerate the Web views that are related to the project.

7 Navigate to the Web view's **Reports** folder, and locate the two .rpt files. From here you can use Crystal Reports to edit the .rpt files. If you wish to add, change, or remove fields, create cross tabs, or add new charts, you will need advanced knowledge of Crystal Reports. Please contact support for more information.

If the report name is "Open Issues - Age by Priority" then the filenames will be:

Open\$Issue\$-\$Age\$by\$Priority**landscape**.rpt and

Open\$Issue\$-\$Age\$by\$Priority**portrait**.rpt

or

rptListingGOpen\$Issue\$-\$Age\$by\$Priority**landscape**.rpt

rptListingGOpen\$Issue\$-\$Age\$by\$Priority**portrait**.rpt

Once the modifications have been made to both files, they must now be copied to the **CustomizedFiles** folder to ensure they will be included the next time the Web views are generated.

See [Customizing Web View Files](#) for more information on the CustomizedFiles folder.

TYPES OF CRYSTAL REPORTS IN VIZOR

Depending on how the data is retrieved, there are three types of crystal reports in VIZOR:

1. Dynamic Query
2. Dynamic Connection
3. Stand-alone

	Query can be modified in VIZOR	Report works automatically for new VIZOR projects and relocated databases
Dynamic Query	Y	Y
Dynamic Connection	N	Y

Stand-alone	N	N
-------------	---	---

	Report knows about the Database Connections?	Report knows about the Query and SQL Statements?
Dynamic Query	N	N
Dynamic Connection	N	Y
Stand-alone	Y	Y

1. DYNAMIC QUERY REPORT

For this type of report, Vizor uses the rpt Crystal Report and injects the data from a query executed by Vizor, not inside the Crystal Report. The data is controlled by the query defined in the Report Editor just as it would be for Standard reports.

The actual SQL statement is not known to the report and the VIZOR engine passes the actual data to the crystal report inside a virtual table called 'ado'.

Selecting the data to include in a report

In the **Query** list, choose a query to extract the records from the database and include them in the report. For example, if you want a report that breaks down the open issue count, then use the Open Issues query.

Selecting the fields to include in a report

In the **Fields** list, choose any of the fields available for that Vizor project or database. Adding or removing fields from the list cause Vizor to pass these fields as the data input provided to the Crystal Reports. How

the field is used or displayed is controlled by the Crystal Report and can be changed in the usual way as for standard Crystal Reports.

From within Crystal Reports, fields passed to this type of report appear in Database Fields as a virtual single table called "ado", regardless of where they actually exist in a database.

ADDING FIELDS TO A DYNAMIC QUERY REPORT

- 1** In Vizor Admin, select a project and open the Report Editor. On the Listing tab, add one or more fields to a custom report. You can insert the new fields anywhere (for example, in between two existing report fields).
- 2** Log on to a Web view of the project and view the custom report.

Vizor dynamically adds the field when you view the report BUT Vizor does not save the changed report. You have to do that yourself (see step 3).
- 3** From the report viewer, save the report in the Reports folder. Do this for both the Portrait and Landscape versions of the custom report.

The Reports folder is located in the ConnectServer folder. For example:
C:\Program Files (x86)\Vector
\ConnectServer\CensusWeb\Views\CensusWebVD\HelpDesk_HelpDesk\Reports
- 4** In Crystal Reports, open one of the .rpt file you just saved. The report includes the new field (to check: in the Field Explorer, expand Database Fields and then expand ado).

Note that if you insert the new field into the report and then try to preview the report, Crystal Reports will display an OLE DB Provider dialog box and prompt you to select a data source.

When this happens, just click Cancel. Vizor sets the data source at runtime.

If you want to be able to preview changes in Crystal Reports:

 - Save the report. If you get a "Unable to Save Data With Report" message box, click No.
 - In the Web view, view the report and save it again (this time the saved report will include the data required by Crystal Reports for previewing).
 - Open the report in Crystal Reports: now you can preview the report without being prompted for a data source.
- 5** Copy the modified .rpt files to the CUSTOMIZEDFILES folder. If you don't, your modified custom reports will be overwritten the next time you generate the Web view.

Note: In general, any time Crystal Reports asks you to either use saved data or refresh data, use the saved data. Refreshing the data requires that you select a data source, and the data source should be set only by Vizor at run time (when a user views the reports).

REMOVING FIELDS FROM A DYNAMIC QUERY REPORT

- 6 In Vizor Admin, select a project and open the Report Editor. On the Listing tab, remove a field from a custom report.
- 7 Log on to a Web view of the project and view the custom report.

When you view the report, Vizor dynamically updates the list of fields included in the report BUT Vizor does not save the changed report. You have to do that yourself (see step 3).

If you get a “This field name is not known” message, then the field is used in a formula somewhere in the report. If this happens, you must open the .rpt file in Crystal Reports, find the formulas that use the field, and remove all references to the field.

If the field was displayed in the report, or used to define a chart, you’ll see a blank space in the report where the field or chart were supposed to be.
- 8 From the report viewer, save the report in the Reports folder. Do this for both the Portrait and Landscape versions of the custom report.

The Reports folder is located in the ConnectServer folder. For example:

```
C:\Program Files (x86)\Vector  
\ConnectServer\CensusWeb\Views\CensusWebVD\HelpDesk_HelpDesk\Reports
```
- 9 In Crystal Reports, open one of the .rpt file you just saved. The report no longer includes the field you removed (to check: in the Field Explorer, expand Database Fields and then expand ado).
- 10 Update the custom report as required. For example, if the removed field was used to define a chart, you’ll have to create a new chart.
- 11 Copy the modified .rpt files to the CUSTOMIZEDFILES folder. If you don’t, your modified custom reports will be overwritten the next time you generate the Web view.

Note: In general, any time Crystal Reports asks you to either used saved data or refresh data, use the saved data. Refreshing the data requires that you select a data source, and the data source should be set only by Vizor at run time (when a user views the reports).

2. DYNAMIC CONNECTION REPORT

For this type of report, Vizor uses the rpt Crystal Report and dynamically provides the necessary connection information for the Crystal Report to execute a query. Vizor provides the database parameters including user credentials. The data is retrieved inside the Crystal Report from the query stored in the rpt file.

Selecting the data to include in a report

The data is not controlled by the query defined in the Report Editor, so you need to query directly the tables and create the necessary joins within Crystal Reports. This type of report requires that the author of the report understands the structure of the database underneath but other conditions or filters that may not be supported by the Vizor query editor may be used.

Selecting the fields to include in a report

The **Fields** list is not used in this type of report. Adding or removing fields from the list in the Report Editor will not change anything in the report since the fields are completely controlled by the Crystal Report. How the field is used or displayed is also controlled by the Crystal Report and can be changed in the usual ways. This type of report requires that the author of the report understands the structure of the database underneath but fields that are not Vizor fields can be used.

From within Crystal Reports, fields appear in Database Fields with the real table and field names and they point to a specific database (e.g. HelpDesk01_DAT).

To add fields to a report of this type, you need to use the Database Expert in Crystal Reports and connect to a valid database, select the table where the field is and add it. Depending on your report, you may need to create a JOIN as well.

3. STAND-ALONE REPORT

This type of report is loaded by Vizor from the rpt Crystal Report file but it does not change it in any way or pass any information to it. Vizor assumes all the necessary connection information, query and data are stored within the Crystal Report. A report of this type can be a report where the database connection information and credentials are stored inside the Crystal Report or it can be a static report that does not query any database (it already has the data inside).

CUSTOMIZING THE REPORT USED BY THE PRINT BUTTON

The report used to print record is named Current Issue - Detailed. If you edit this report in the Report Editor (of Vizor Admin), the changes apply to all Web views of the project.

By default the report is a Standard Report and the field can be defined directly in the Report Editor.

However, you can choose to make it a Custom Report, in which case you need to click on the checkbox Custom and provide the two required rpt files.

Each view has two versions this report:

```
rptListingGCurrent$Record$-$DetailedPortrait.rpt
rptListingGCurrent$Record$-$DetailedLandscape.rpt
```

You can find examples of these files in the

```
CensusWebVD\<project>_<view>\Reports
```

folder, where <project> is the name of the project and <view> is the name of the Web view. The portrait version of the report is used to print issues.

To customize the report:

- 1 View the portrait version of the report in a Web view. You can either print an issue or view the report with the Report Viewer.

Viewing the report creates the .rpt file.

- 2 In the Report Editor (Vizor Admin), mark the report as a custom report (select the **Custom** check box).

This prevents Vizor from overwriting the customized report if the report definition changes.

- 3 Using Crystal Reports, customize the report (add graphics, change fonts, rearrange and align fields).

- 4 Put a copy of the customized report in:

```
CUSTOMIZEDFILES
  \#Project#\<project>
    \#WebView#\<view>
      \Reports
```

When you regenerate the Web view, the customized version of the report is copied to the Web view folder.

EXPORTING TO CSV

Custom reports can be exported to CSV from the Web.

By default the separator is the comma “,”. If you need the CSV separator to be different, for example “;” , you can do it by adding a custom field inside the report.

This is defined per report and inside the Crystal Report.

Inside the Crystal Reports Editor, create a Formula Field called “**CSV-CharFieldDelimiter**”.

In the formula editor, simply add the character to be used as the delimiter or separator, like “;”, but not surrounded by quotes.

If not present or empty, the default will be used.

Internally, the format of this type is CharSV=7. You can also make a specific report open by default in CSV by customizing the Reports.js file.

Advanced Setting for the Report Engines: USING CRYSTAL REPORTS FOR ALL

LISTING REPORTS

It is also possible to use the Crystal Reports engine for all the Listing reports, rather than the HTML report engine. In this case, a Creation API for Crystal Reports would need to be purchased from a third party. Once purchased, this Creation API, after having been installed on the web server, will enable Vizor to display all existing and future Listing reports using the Crystal Reports Engine. This is done automatically without the need to use the Crystal Reports Editor. Since this involves the automatic generation of .rpt files, Crystal Reports requires that the Creation API be present.

Options control which report viewers are used by the Web views:

Standard Report Engine specifies the report viewer used for Listing reports.

Custom Report Engine specifies the report viewer used for Custom reports.

To set these options, log on to Vizor Admin, and on the **Tools** menu, click **Options**.

Note: If you have a version of Crystal Reports that is newer than version 9, you must still select **Crystal Reports 9** as the report viewer.

By default, Vizor uses the included Crystal Reports Runtime Software to display custom reports, and a built-in HTML report engine to display listing reports, print issues from the Web Views, and format notification contents such as attachments.

TIME REPORTS

Time reports show trends over time. Use Time reports to answer questions like: "What is the issue arrival rate?", or "How fast are issues being resolved?"

DEFINING THE TIME AXIS

The time axis divides the reporting period into intervals of days, weeks, months, quarters, or years (this is the granularity of the report). The Time report then sorts the issues into the time intervals, and counts the number of issues in each interval.

To sort issues into time intervals, a Time report uses a master date. The master date is a specified in an issue field such as **Submitted Date**, **Closed Date**, **Assigned Date**, or **Update Date**.

Master Date	Description
Assigned Date	The date an issue was assigned. For a report that shows the number of open issues per analyst or developer, you could use the Assigned Date instead of the Submitted Date .
Closed Date	The date an issue was closed. This happens when Progress is set to a value that maps to the Closed state. Use this date for reports that show the number of issues fixed

over time (fix rate reports).

Submitted Date The date an issue was submitted.

Use this date for Time reports that show the number of submitted, open, or closed issues over time.

Update Date The date an issue was last modified.

INCLUDING ALL DATES

Select the **Show dates with no values** check box to show days, weeks, months, quarters, and years when there were no issues. For example, in a report that shows the daily arrival rate of new issues during the last week, you probably want to see a value for each day in the week, even if that value is 0.

SHOWING INTERVAL TOTALS

When the **Cumulative** check box is selected, a Time report accumulates the number of issues from one interval to the next. For example, suppose you have a Time report that shows the number of submitted issues per week. To see how the total number of submitted issues grows week by week, select the **Cumulative** check box. To see the number of issues submitted each week, clear the **Cumulative** check box.

DEFINING REPORTING PERIODS

Enter specific **From** and **To** dates, or choose one of the When conditions: **This Month**, **This Quarter**, or **This Year**. You can also combine **From** or **To** with **When**. For example, if the year is 2006 and you set **From** to **3/3/06** and **When** to **This Year**, your reporting period would be 3/3/06 to 12/31/06.

INCLUDING TOTALS

Select the **Total** check box. For each time interval, the Time report shows the total number of issues included in the report.

CONTROLLING COLUMN WIDTH IN TABULAR TIME REPORTS

Use the **Column Width** box to set a maximum width for all columns in a report, including the column headings.

DEFINING THE FIELD AXIS

The field axis determines which issues a report counts. For example, to include the number of **ASAP** and **Today** priority issues, choose these **Priority** field values as the columns. The Time report then counts the number of **ASAP** and **Today** priority issues in each interval.

Each column corresponds to a possible value of a choice-list field (single or multi). For example, you can add columns for all possible priority values, or just for the highest priorities (ASAP, Next 4 Hours, Today).

TEXT (CROSS-TAB) REPORTS

Each column is a column in the cross-tab table.

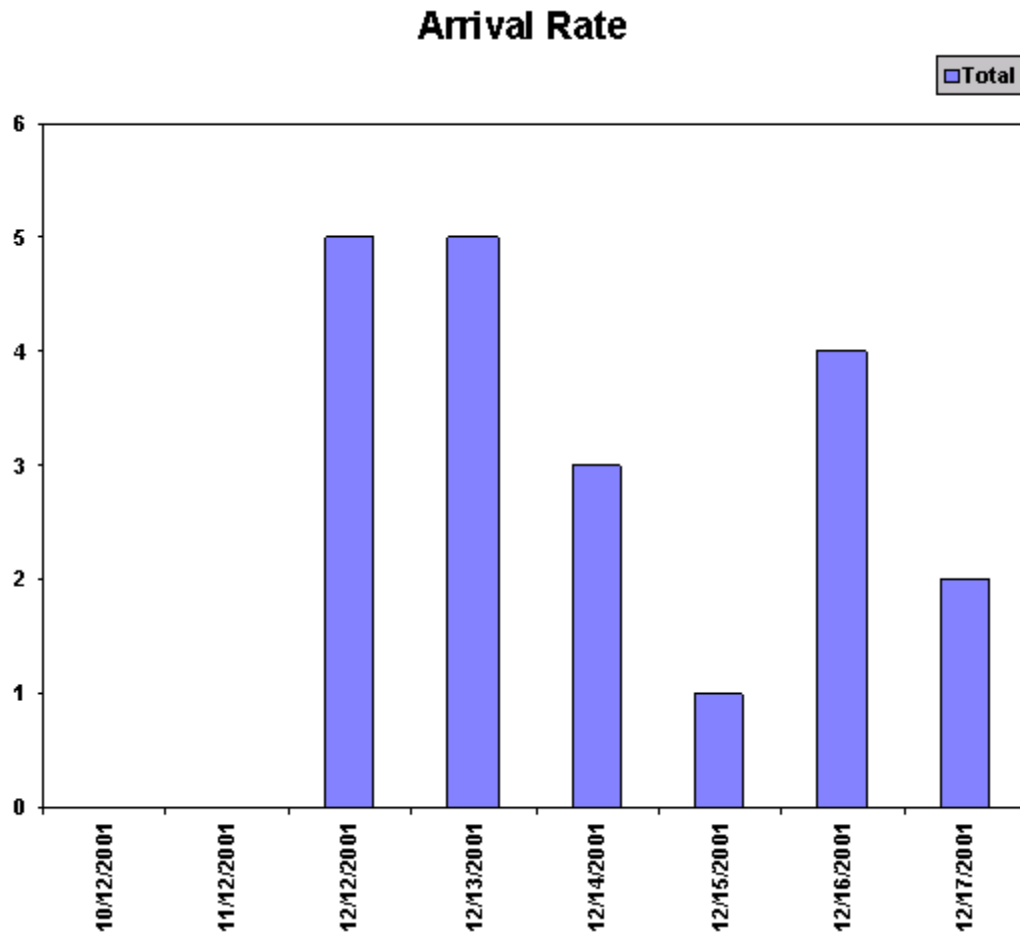
LINE CHARTS

Each column is a line in the line chart.

BAR CHARTS

The column specifies what bars to display for each row.

If you set **Field Name** to **<None>**, then the Time report provides a total of all issues for each interval.



DEFINING REPORT PAGES

Report pages define formatting elements common to every page of a report, such as headers, footers, titles, and column layout. Report pages also define report titles. Pages can be shared by many reports, allowing you to define a standard look for all reports generated by Vizor. Report Pages are only used in Standard HTML Reports.

Vizor comes with a set of predefined pages. You can adapt these templates to your specific requirements, or define entirely new pages.

TO OPEN THE PAGE EDITOR:

On the **Tools** menu, click **Report**, then click **Page Editor**.

- Choose a page from the **Name** list. You can then edit, delete, or rename the page.
- To use one page as a starting point for a new page, choose a page from the **Name** list, click **Copy**, and give the new page a name.
- To create a page from scratch, click **New**.

You can use the Page editor to define headers, footers, and titles for your reports. In addition to typing and formatting plain text, you can insert placeholders for the current date, time, and page number. When you generate your report, these placeholders are replaced with values.

To do this	Click
Insert the current date	
Insert the current time	
Insert the current page number	
Left-align text	
Center text	
Right-align text	
Format text by changing fonts and point sizes, and applying bold, italic, or underline styles.	

CHAPTER 14 - COMPUTERS

DISCOVERY OF COMPUTERS - BRINGING COMPUTERS INTO VIZOR

Computers are special types of assets managed in Vizor. Computers have special attributes or fields such as CPU, CPU Speed, Memory, Operating System, etc.

Before you can manage your computer assets in Vizor, you can import them from one or several of these sources:

VECTOR DISCOVERY

Vizor can discover the computers in your network via our own Vector Discovery tool. If you use our discovery tool, computers are automatically imported into Vizor.

MICROSOFT SYSTEM CENTER CONFIGURATION MANAGER (SCCM)

If you have deployed Microsoft's SCCM across your network, Vizor's SCCM Connector imports and synchronizes the computers information collected by SCCM. The SCCM Connector also imports the software installations per computer as discovered by SCCM.

SOLARWINDS

With Vizor's SolarWinds Connector, import and keep synchronized the list of servers managed by SolarWinds, including computer details.

LANSWEEPER

Similar to the connector for SCCM, you can import and synchronize the list of computers discovered by LanSweeper.

EXCEL / WORKSHEET

Some times the best source for the import is an Excel file. Maybe it was created manually or maybe a 3rd party system can easily export the inventory to Excel. Vizor's native adapter easily imports and synchronizes from a list of computers in Excel.

IN-HOUSE INVENTORY

If your organization has deployed a custom computer discovery or inventory system, you can still use Vizor's flexible adapter technology to import and synchronize the list of computers.

NEEDS ATTENTION - NOT SEEN RECENTLY

Vizor automatically sets the field flag Needs Attention – Not Seen Recently to Yes if an asset that was imported from discovery (SCCM, Vector Discovery, etc.) has not been seen recently.

For example, if a computer is decommissioned outside VIZOR or deleted in SCCM/Vector Discovery, it will not be updated in VIZOR. Computers are updated daily by default.

How much time passes before the flag needs attention is raised?

The time is configurable in the Admin > Options.

Look for the option **Days before asset marked as Not Seen Recently**

The default value is **60** (days)

CHROMEBOOKS

Chromebooks have special characteristics.

IMPORT CHROMEBOOKS FROM GOOGLE'S G-SUITE

You can use the app **chromebookInventory** to export Chromebooks from Google Apps domains with Chrome Device Management Console.

The following is the default mapping when using the default import rule:

Internal Vizor Field	Google Console Field
Domain_Name	Org Unit Path
MqAssetOwnerLookup (Owner)	Annotated User
Location	Annotated Location
Asset Tag	Annotated Asset Id
Notes	Notes

Primary_Owner	Recent Users
Serial_Number	Serial Number
MAC_Address	Mac Address
ComputerName	Device Id
Model	Model
OS_Version	Firmware Version
Source_Timestamp	Last Sync
tBriefDescription	Device Id
tSID	Device Id

Defaults

Internal Vizor Field	Default Value
Asset Type	Chromebook
OS_Name	Chrome OS
Item_Level	3
nUsageType	1

CHAPTER 15 - MANAGING CONTACTS, USERS AND GROUPS

ABOUT USERS AND CONTACTS

In Vizor, you work with four types of accounts:

- Users, who have logon names and passwords and can log on to Web views, Vizor Web Admin, Vizor Admin, or the Web View Editor.
- Contacts, who cannot log on. Contacts are people who, for example, report issues by telephone, e-mail, fax, or through submit-only views.
- Companies, who cannot log on. Companies are related to either a Contact, or User of the Vizor system.
- Value-Added Resellers (VARs), who cannot log on. Like Companies, VARs are related to either a Contact, or User of the Vizor system.

Users are not specific to projects. For example, a user can use the same logon name and password to log on to any Web view of any project. However, each project can store different information for users. When you add a field to the **Contact** tab of a project, that field is a project-specific user information field.

Companies and VARs can be used to better organize Users and Contacts. For example, when multiple Contacts are associated to the same Company, it becomes easier to search for all issues related to the Company - the same is true for VARs. However, there are specific features in Vizor that only work with Companies, for example:

- Service level agreements can be created to apply to all users of one specific company. For more information, see [Companies](#).
- When Contacts are created through E-mail Integration, Vizor will attempt to automatically associate a Company. For more information, see [Setting Default Values for Issues Submitted by E-mail](#).

IMPORTING USERS AND CONTACTS FROM WINDOWS

You can import user accounts from Active Directory (or from a Windows domain), so users can log on to Vizor with their Windows user names and passwords. Use the User Account Import Manager to import user accounts and keep Vizor synchronized with Active Directory/Windows.

ADMINISTERING USERS AND CONTACTS

To create and manage users and contacts in Vizor, use Vizor Web Admin. For example, you can use Web Admin to disable users, create contacts, and to create user accounts.

Users with the required permissions can also create contacts in Web views.

USER SECURITY

VIZOR security enables you to manage user access to issue data and features. In VIZOR, security is based on the users and groups.

When you install Vizor, the Setup program creates sample users, such as **Admin**, and **demo**, along with a number of role-based user accounts.

To secure your projects, you must import or create user accounts for users. You then assign the users to groups, and use group privileges to control their access to tools, such as Vizor Admin and Vizor Web Admin and for individual features such as the Report editor.

USER GROUPS

Each user is a member of one or more groups. Access to Vizor features and projects is based on group membership. You can enable and disable features and projects on a group-by-group basis. A user can only access a feature or project if the user is a member of a group where the feature or project is enabled.

Vizor has several default groups.

- The **Admins** group has all features enabled
- The **Users** group enables features that you want to make available to all users. All users are members of this group
- The **Analysts** and **Group Leaders** groups have access to all Web View features controlled by security permissions
- The **Employees** group cannot access any of the Web view features controlled by security permissions

To control access to Vizor features, you assign users to the groups based on their roles and responsibilities in the issue tracking process. For example, a help desk group leader or analyst who is also responsible for administering Vizor must be a member of the **Admins** group. For example, a member of the QA staff who is also responsible for administering Vizor would be a member of both the **QA** and the **Admins** group.

NOTES

- Use the Web View Editor to set the [group access permissions](#) that control who can log on to the Web views of a project.
- Use Vizor Admin to control access to projects.

IMPORTING USER ACCOUNTS

You can import user accounts from an Active Directory domain, an LDAP repository, from Microsoft Dynamics or from other sources.

TO IMPORT USER ACCOUNTS FROM AN ACTIVE DIRECTORY OR WINDOWS DOMAIN:

- 1 Log on to the **User Account Import Manager**, and click **Next** in the **Welcome** dialog.
- 2 In the **Import Rules** dialog, click an import rule to specify the users you want to import.
- 3 In the **Active Directory Servers and Windows Domains** dialog, select an Active Directory LDAP server or a Windows domain.

To add an Active Directory server, click [Add LDAP](#). To add a Windows domain, click [Add Domain](#).

- 4 In the **Active Directory Servers and Windows Domains** dialog, click the Active Directory server or Windows domain from which you want to import users.

- 5 In the **Import Users** dialog, choose one of the following options:

- To import all users listed in the dialog, click **Next**.
- To filter the list based on Active Directory groups, click **Add Filter** and select the groups to filter, then click **Append** to add the group filter to the current search filter, or click **Replace** to replace the current filter with the group filter. Click **Refresh** to apply the search filter.

You can also type additional LDAP search filter conditions directly in the **Filter** box. For example, "(cn=e*)" selects all users with user names starting with "e", and "(&!cn=IUSR*) (!cn=CensusUser)" filters out all IUSR_<computername> accounts and the CensusUser account.

- To filter the list based on Windows Domains groups, select the groups you want to import from the list.
 - To import some of the users, click **Selected Users**, then click the users you want to import.
- 6 In the **Import Settings** dialog, select the **Import as Contacts** check box if you want to import contacts instead of users.
 - 7 To add the imported users to a specified user group, such as **Employees**, click the group to which you want to add the new users in the **Import to Group** list. (Note that imported users are always added to the **Users** group.)
 - 8 To schedule imports, select the **When Scheduled** check box, then click **Configure**.

TO IMPORT USER ACCOUNTS FROM MICROSOFT DYNAMICS

See [Importing Customers or Contacts from Microsoft Dynamics](#).

NEEDS ATTENTION - NOT SEEN RECENTLY

Vizor automatically sets the field flag Needs Attention – Not Seen Recently to Yes if a user account that was imported has not been seen recently in the last synchronizations with the source.

For example, if a user account is deleted in Active Directory and it was imported into VIZOR, it will not be updated in VIZOR on the daily synchronization.

How much time passes before the flag needs attention is raised?

The time is configurable in the Admin > Options.

Look for the option **Days before user marked as Not Seen Recently**

The default value is **7** (days)

DEFINING IMPORT RULES

An import rule remembers the settings from the last time you used the rule.

To define an import rule:

- 1 In the Import Rules dialog, click **Add** and type a name for the import rule.
- 2 Click **Next** and continue through the remaining steps of the User Account Import Manager.

All your subsequent choices and settings are saved in the import rule. For example, suppose you select an Active Directory server, specify an LDAP search filter, and change the default import settings. The next

time you run the User Account Import Manager, you can select the import rule to use the same choices and settings.

INTEGRATING WITH ACTIVE DIRECTORY

Before you can import user accounts, you must add a connection to an Active Directory server or Windows domain.

To add an Active Directory server:

- 1 In the Active Directory Servers and Windows Domains dialog, click **Manage**.
- 2 In the Manage Active Directory Servers and Windows Domains dialog, click **Add LDAP**.
- 3 In the **Display Name** box, type a name for the LDAP server. Vizor uses this name to identify the server.
- 4 In the **Host**, **Port**, and **Base DN** boxes, enter the connection information for the LDAP server:

Host is the IP address of the LDAP server.

Port is the port the LDAP server is running on. By default, LDAP servers run on port 389. LDAP over SSL uses port 636 by default.

Base DN is the top level of the LDAP directory tree. For example, if your organization's DNS domain name is *mycompany.local*, the Base DN is *dc=mycompany,dc=local*.

- 5 If you want to use anonymous authentication for the connection, select the **Anonymous Bind** check box.

Alternatively, clear the **Anonymous Bind** check box and enter the information for authenticating a user when connected to the server, where:

User DN is the user to authenticate. For example: *CN=Administrator,CN=Users,DC=mycompany,DC=local*

Password is the user's password

- 6 In the **LDAP Logon Name Field** box, type the name of the LDAP property that stores the user account name. Vizor uses this name to log on.
- 7 Click **Test** to test the connection.

INTEGRATING WITH ACTIVE DIRECTORY WHEN VIZOR IS IN THE **CLOUD**

Vizor can integrate with your Active Directory for synchronizing user accounts and allowing users to log in using the same user credentials.

REAL-TIME INTEGRATION

If you want the Vizor cloud server to communicate with your Active Directory in real time, an open IP channel needs to be allowed across your firewall.

Typically, the TCP and UDP 389 ports are needed, but depending on the configuration there may be more. You will need to consult with your firewall configuration expert.

Vizor uses the LDAP protocol on this port.

You can also use the LDAPS protocol.

IMPORT

As an alternative, you can export the users list from your AD and then import them into Vizor. This solution, however, does not bring the user passwords and therefore users would need to set the passwords in Vizor.

FILTERING USERS

You can use LDAP search filters to filter the list of users imported from Active Directory.

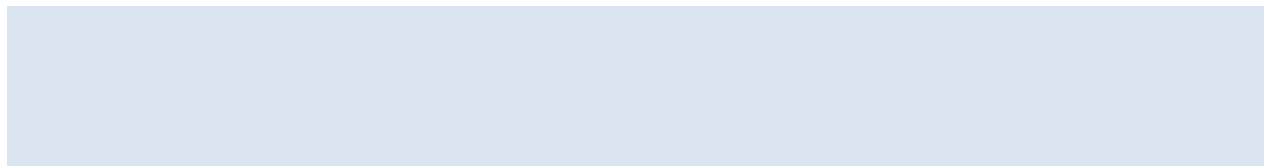
If you want to filter users based on Active Directory groups, click **Add Filter** and select the groups you want to filter. Then click **Append** to append the group filter to the search filter in the **Filter** box, or click **Replace** to replace any existing search filter.

If you want to filter on other conditions, you can also write your own LDAP search filters, and combine them with the group filters created with **Add Filter**. For example, you could type in filters such as:

- (sn=Mark)
selects users with a surname of Mark.
- (!cn=IUSR_*)
filters out all user names that start with IUSR_*.
- (&(!cn=IUSR_*) (!cn=IWAM_*))
filters out all IUSR_<computername> and IWAM_<computername> users.

For more information on the LDAP search filter syntax, see:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/adsisearch_filter_syntax.asp.



IMPORT SETTINGS

Import as Contacts

If you want to import the users as contacts, select the **Import as Contacts** check box.

Import to Groups

By default, imported users are added the Users group in Vizor. If you want to add the imported users to a different group (for example, Admins), click the group in the Import to Groups list.

Get E-mail Addresses From

Specifies where to get the e-mail addresses of imported users.

Scheduling Imports

Specifies when to import users. Select the **When Scheduled** check box and then click **Configure** if you want to use Window's Scheduled Tasks to schedule regular imports. Scheduling imports keeps Vizor synchronized with Active Directory and Windows domains.

Select the **Now** check box if you want to import users as soon as you finish with the User Account Import Manager.

Import Settings

These check boxes control whether the User Account Manager:

- Adds new user accounts for any new users found in the Active Directory or Windows domain.
- Updates existing user accounts with any changes (for example, to the user name or e-mail address) found the Active Directory or Windows domain.

CREATING USER ACCOUNTS

New users automatically belong to the Users group; however, administrators can assign the User to additional groups, if desired.

To create a user account:

1. In Vizor Web Admin, select the **Security** tab and click the **Users** button.

Alternatively, in Vizor Admin, click **Tools > Security > Users**.

2. Click **New** .
3. In the **Type** field, select **User**.
4. In the **Authenticate With** list, click Vizor **Authentication** to authenticate against the Vizor user database.
5. In the **Logon Name** box, type the user name.
6. In the **Password** and **Confirm Password** boxes, type the user's password.
7. Enter the rest of the user information and click **Apply**.

If you want to control the projects in which the user is visible, use the **User Category** list to assign the user to a category (for example, Internal User or External Contact).

Different projects can be configured to display different categories of users and contacts.

ENABLING AND DISABLING PROJECTS

When you disable a project for a group, members of that group cannot:

- Log on to Web views of the project.
- Edit the project in Vizor Admin.
- Edit views of that project in Web View editor, or generate new views of that project.

You can [disable specific Web views](#) for members of a group.

To enable or disable a project for a group:

- 1 In Vizor Web Admin, click the **Security** tab, then click the large **Groups** button on the left of the tab.
- 2 In the **Groups** list, click a group.
- 3 Click **Projects** 
- 4 If you want to enable a project:
 - a In the **Disabled** list, click the projects you want to disable. Use the Shift and Control keys to select more than one project.
 - b Click .
- 5 If you want to disable a project:
 - c In the **Enabled** list, click the projects you want to disable. Use the Shift and Control keys to select more than one project.
 - d Click .

You can also enable or disable projects from the **Groups Allowed to Open** list in the Project Properties dialog.

WORK TEAMS

You can use VIZOR to help manage and coordinate work teams. Setting up work team support involves the following tasks:

- Building a list of work teams. Manage Work Teams as Departments.
- Assigning users to work teams.
- Adding the Assigned Work Team field to Web views. This field is used to assign issues to specific work teams, either manually by users or automatically by workflow rules.

Define queries for finding work team issues. VIZOR includes a **My Work Team's Issues** query that gets all the issues assigned to a user's work team, but you may want to define additional queries for work teams (for example, a query that finds all open issues assigned to the user's work team).

Defining Work Teams

The list of available work teams is stored in a global choice list named Work Team. The HelpDesk project has an **Assigned Work Team** field that uses this global choice list.

You use Vizor Web Admin to edit the list of work teams.

To edit the list of Work Teams:

- 1 In VIZOR, go to the **Roles and Policies** area.
- 2 Click on the queries and select **Work Teams** in order to see the current list.
- 3 The list is also a Global Choice List, accessible for Single-Choice and Multi-Choice fields in all projects.
- 4 Edit the Work Teams:

Click on the appropriate button on the right to create a new Work Team.

To delete a Work Team, click on the Delete red icon on the row in the Summary List.

To rename a Work Team and edit other properties, click the Work Team on the Summary List. Make changes in the form that opens and save as usual.

ASSIGNING USERS TO WORK TEAMS

You can assign both users and contacts to work teams. No user or contact can belong to more than one work team.

To assign a user or contact to a work team:

1. In Vizor Web Admin, select the **Security** tab and click the **Users** button.
2. Select the Users query.
3. In the query results, click the user or contact you wish to edit.
4. In the **Work Team** list, click the work team to which the user belongs.

ADDING WORK TEAM SUPPORT TO WEB VIEWS

To allow users and workflow rules to assign issues to work teams, [export](#) the **Assigned Work Team** field (**Overview** tab) to the Web views.

BUILDING WORK TEAM QUERIES

You can query for issues assigned to a specific work team, for example, all issues assigned to the QA Team:

```
Assigned Work Team = QA Team
```

You can also use the **<My Work Team>** macro to query for issues assigned to the work team of the current user. For example, this query finds all open issues assigned to the user's work team:

```
Assigned Work Team = <My Work Team> AND  
State = Open
```

USER CATEGORIES

You can control which users and contacts are visible in a project. Each user (and each contact) can be assigned to a user category. For example, some users can be assigned to the Internal User category, and others to the External User category.

The list of available user categories is a global choice list, so you can use Vizor Web Admin to build the list of user categories you want to use.

To edit the list of user categories:

- 1 In Vizor Web Admin, click the **Choices** tab.
- 2 In the Project list, click < **Global Choice Lists** >.
- 3 In the **Choice List** list, click **User Category**.
- 4 In the **Choices** section:
 - To add a new user category, click  and then type the category name.
 - To rename a user category, click the category and then click .
 - To delete a user category, click the category and then click .

ASSIGNING USERS TO CATEGORIES

Assigning users and contacts to categories allows you to control whether they are visible in a project.

To assign users and contacts to categories:

- 5 In Vizor Web Admin, select the **Security** tab and click the **Users** button.

Alternatively, in Vizor Admin, click **Tools** > **Security** > **Users**.

- 6 Select the All Users & Contacts query.
 - For more information about using queries, see [About Queries](#).
- 7 In the query results, click the user or contact you wish to edit.

8. In the **User Category** list, click the category to which the user belongs.
9. Click **Save**.

HIDING USERS IN PROJECTS

To control which users, contacts, and companies are visible in a project:

10. In Vizor Web Admin, click the **Options** tab.
11. In the **Project** list, click a project.
12. Click Organize users per project.
13. Check the In this project, only display users, contacts, and companies in the following categories box.
14. Select the categories of users and contacts you want to be visible in the project.
15. Click **OK**.

DISABLING USER ACCOUNTS

Disabled accounts cannot be used to log on, but the names still appear in the **Owner** and **Contact** lists because existing issues may still refer to the users.

Disabling a contact, company, or VAR has no effect.

To disable a user account:

16. In Vizor Web Admin, select the **Security** tab and click the **Users** button.

Alternatively, in Vizor Admin, click **Tools > Security > Users**.

17. Select the **All Users** query.

- For more information about using queries, see [About Queries](#).

18. In the query results, click the user you wish to edit.

19. Check the **Is Disabled** box.

20. Click **Save**.

EDITING USER INFORMATION

You can edit user information for Users, Contacts, Companies, and VARs with Vizor Web Admin.

To edit user information:

21. In Vizor Web Admin, select the **Security** tab and click the **Users** button.

Alternatively, in Vizor Admin, click **Tools > Security > Users**.

22. Select the All Users, Contacts, Companies & VARs query.

- For more information about using queries, see [About Queries](#).
- To quickly find a user, use of the search feature is recommended. To make the search box appear, press the  button.

23. In the query results, click the user you wish to edit.

24. Make the modifications you desire.

25. Click **Save**.

PASSWORDS

You can change passwords for Vizor native user accounts. You cannot change passwords for user accounts imported from Active Directory or Windows.

To change a user's password:

26. In Vizor Web Admin, select the **Security** tab and click the **Users** button.

Alternatively, in Vizor Admin, click **Tools > Security > Users**.

27. Select the All Users, Contacts, Companies & VARs query.

- For more information about using queries, see [About Queries](#).
- To quickly find a user, use of the search feature is recommended. To make the search box appear, press the  button.

28. In the query results, click the user you wish to edit.
29. Change the value in the **Password** and **Confirm Password** fields.
30. Click **Save**.

CHANGING THE AUTHENTICATION METHOD

- 1 In the **Authenticate With** list:
 - Click Vizor **Authentication** to authenticate against the Vizor user database.
 - Click an Active Directory server to authenticate against Active Directory.
 - Click a Windows server to authenticate against a Windows domain.
- 2 If you authenticate with VIZOR, you may need to enter the logon name and password.

DEFINING USER GROUPS

Groups allow you to assign privileges to users. By enabling and disabling features for a group, you control the features (privileges) available to the members of that group.

Group membership determines which features are available to a user. Each group enables and disables different features. Users that belong to more than one group have access to any feature enabled in at least one of the groups.

Group membership also controls access to projects and Web views. To assign users to groups, use Vizor Admin .

In the **Users** view of the **Security** tab, you can select a user and then edit the list of groups to which the user belongs. In the **Groups** view, you can select a group and edit the list of users that belong to the group.

To add a group:

- 1 In Vizor Web Admin, click the **Security** tab.

If you are in Vizor Admin, click **Tools > Security > Groups**.

- 2 On the **Security** tab, click the large **Groups** button on the left side of the tab.

- 3 Click **Add**  and type a new for the group.

To delete a group:

- 1 In Vizor Web Admin, click the **Security** tab.

If you are in Vizor Admin, click **Tools > Security > Groups**.

- 2 On the **Security** tab, click the large **Groups** button on the left side of the tab.

- 3 In the **Groups** list, click the group you want to delete.

- 4 Click **Delete** .

GROUPS: ADDING AND REMOVING USERS

On the **Security** tab, the **Groups** view allows you to add or remove multiple users at the same time.

To add users to a group:

- 1 In Vizor Web Admin, click the **Security** tab.
- 2 On the **Security** tab, click the large **Groups** button on the left side of the tab.
- 3 In the **Groups** list, click a group.
- 4 Click **Users** 
- 5 To add users:
 - a In the **Non-Members** list, click the users you want to add to the group. Use the Shift and Control keys to select more than one user.
 - b Click .
- 6 To remove users:
 - c In the **Members** list, click the users you want to remove from the group. Use the Shift and Control keys to select more than one user.
 - d Click .

USERS: CHANGING GROUP MEMBERSHIP

To add a user to a group (in the Users view):

1. In Vizor Web Admin, select the **Security** tab and click the **Users** button.
2. Select the **All Users** query.
 - For more information about using queries, see [About Queries](#).
 - To quickly find a user, use of the search feature is recommended. To make the search box appear, press the  button.
3. In the query results, click the user you wish to edit.
4. Select the **Groups** tab.
5. Click the **Edit** button.
6. In the **Select Groups** dialog, select the check boxes for the groups to which you want to add the user.
 - To filter the list of groups, click **Advanced**  and type part of a group name. Click **Filter** to list only the groups that contain the text you typed.
 - To find all groups with names that start with a certain string of characters, click **Options** and in the **Search Type** list, click **Starting With**.
7. Click **OK**.
8. Click **Save**.

ORGANIZATIONS & COMPANIES

VIZOR maintains a list of companies. By assigning users to companies, you can create a service level agreement that applies to all users at a specific company.

To create a company:

9. In Vizor Web Admin, select the **Security** tab and click the **Users** button.
10. Click **New** .
11. In the **Type** field, select **Company**.
12. Provide all required information.
13. Press **Save**.

DEPARTMENTS

By assigning users to departments, you can create a service level agreement that applies to all users in a specific department.

To edit the list of departments:

- 1 In VIZOR, go to the **Roles and Policies** area.
- 2 Click on the queries and select **Departments** in order to see the current list.
- 3 The Departments list is also a Global Choice List, accessible for Single-Choice and Multi-Choice fields in all projects.
- 4 Edit the departments:

Click on the appropriate button on the right to create a new department.

To delete a department, click on the Delete red icon on the row in the Summary List.

To rename a department and edit other properties, click the department on the Summary List. Make changes in the form that opens and save as usual.

VIZOR FEATURES

Vizor includes features that can be enabled and disabled. Most VIZOR Features correspond directly to buttons or menu options in the user interface (UI), such as **Security** or **Report Editor**. When a user does not have permission to use a feature, the feature is not displayed in the UI.

To enable or disable a feature for a group:

- 1 In Vizor Web Admin, click the **Security** tab.
- 2 On the **Security** tab, click the large **Groups** button on the left of the tab.
- 3 In the **Groups** list, click a group.
- 4 Click **Features**.
- 5 To enable a feature:

a In the **Disabled** list, click the feature you want to disable. Use the Shift and Control keys to select more than one feature.

b Click .

6 To disable a feature:

c In the **Enabled** list, click the features you want to disable. Use the Shift and Control keys to select more than one feature.

d Click .

WEB VIEW FEATURES

Feature	Permits the user to
Ad hoc Queries	Access the Ad-hoc Query Editor
Add Contact Editor	Open the Users and Contacts dialog in a Web view by clicking the Contacts button
Add Contact	Create contacts
Add & Save Issues	Create new issues and save them
Copy & Save Issues	Create a new issue based on the current one in the Web view and save it
Delete Contact	Disable contacts
Update Contact	Edit contacts
Default Value Editor	Set default values for new issues
Email Conversation	View and manage the e-mails linked to the

	current issue in the Web view
E-mail Current Issue	Send an e-mail with a summary of the current issue from the Web views
Email Queue	View and manage the e-mails that are not linked to issues
Manage E-mail Templates	Create, update and delete e-mail templates that are used when composing new e-mails or in automated e-mail responses
My Queries	Run personal queries only available to the user from the Web views
Password Editor	Change password
Queries - View Source	View the SQL WHERE clause behind the named query
Report Viewer	View Vizor reports
Report Viewer - Hardware Inventory	View Hardware Inventory reports
Report Viewer - Software Inventory	View Software Inventory reports
Revision History	View the change history for an issue
Save & Remove User Queries	Create new personal queries and be able to delete them
Save Issues	Update and save issues from the Web views
Update Submit Only View Contact	Update the contact information from a Submit-Only View

Use E-mail Templates Apply saved e-mail templates when composing new e-mails

Example

The staff responsible for logging issues received from customers need to be able to add new contacts (each customer is a contact). To allow users to add new contacts in Web views, enable these features:

Add Contact Editor

Add Contact

VIZOR WEB ADMIN FEATURES

The following features enable and disable the different tabs in Vizor Web Admin.

Feature	Permits the user to
Web Admin - Banner	Create and edit banners.
Web Admin - Choices	Create and edit choices lists
Web Admin - Email Integration	Manage and configure integrations of Vizor and e-mail accounts for submitting issues by e-mail, linking e-mails to issues, automatic status by e-mail and more
Web Admin - Inventory Editor	Connect projects to Vector Asset Management or PC-Duo Enterprise site databases for linking computers and other assets to the issues as well as viewing hardware and software reports
Web Admin - Notifications	Set the user account used for notifications

Web Admin - Relocation	Relocate databases and attachments
Web Admin - Security	Edit users, contacts, and groups. Add, edit, and delete groups. Enable and disable features for groups
Web Admin - SLA - Manage Agreements & Levels	Manage Service Level Agreements and Service Levels, including working hours, target times and escalation rules
Web Admin - SLA - Report Viewer	View reports with summary and metrics of Service Level Agreements performance
Web Admin - SLA - View	View only the Service Level Agreement Information
Web Admin - Tabs	Create new tabs
Web Admin - Users - Add	Create new users
Web Admin - Users - Disable	Disable users therefore preventing them from logging into the system
Web Admin - Users - Update	Update user information such as e-mail address, postal address and telephone
Web Admin - Users - View	View only users and user information
Web Admin - Workflow	Edit workflow rules

ADMIN

To enable Admin features, you must enable Admin Utility. Otherwise, users cannot log on.

Feature	Permits the user to
Admin Utility	Log on to Vizor Admin
Delete Project	Delete a project
Delete Issues	Delete issues from a project
Import Issues	Import issues into a project
Integrity Editor	Open the Integrity Editor
Layout Editor	Open the Layout Editor and define custom layout styles for the Summary List
Licenses	Open the Licenses dialog
New Project	Create a new project
Notification Editor	Set up automatic e-mail notifications
Page Editor	Create and edit report pages
Project Properties	Edit project properties, such as which groups are allowed to open a project
Query Editor	Open the Query Editor and define custom queries
Report Editor	Define new reports, and edit existing reports

Sort Editor	Open the Sort Editor and define custom sorts
Field Editor	Add new fields, change labels and modify the values that appear in choice lists
Web Tools Menu	Start the Web View Editor and Vizor Web Admin from the Tools > Web menu
When Editor	Edit the when conditions for notifications
Windows Account Editor	Edit the Windows user account used for anonymous access to virtual directories and for running the notification service.

Example

To allow users to create new reports, enable, at a minimum, the Admin Utility and Report editor features. This allows users to log on to Vizor Admin and create new reports using existing queries, sorts, and report pages. The new reports are automatically made available in existing Web views.

Enabling the Query editor, Sort editor, and Page editor features allows users to create new queries, sorts, and report pages and use them in report definitions. New queries aren't available in Web views unless you export the queries and regenerate the views. All new sorts, however, are automatically made available in Web views.

WEB VIEW EDITOR FEATURES

You can prevent users from logging on to the Web View editor by disabling the Web View Editor feature.

CHAPTER 16 - FIELDS

THE FIELD EDITOR

Before you start tracking assets and issues, you may need to customize some of the fields used to collect and display information. For example, the **Type** and **Priority** fields should list choices that correspond to your organization's terminology and environment.

You may want to delete unused fields, add new fields to track additional information, or add new tabs. For example, you may want to add a tab for hardware configuration details.

Vizor Web Admin provides a Field Editor for editing fields. To open this editor, click on the **Fields** tab and then click on **Fields** on the left side menu. The Field Editor allows you to edit field definitions, but to export fields to Web views use the Web View Editor.

EDITING FIELDS

Click a field in the **Field** list to edit, copy, rename, or delete the field. The **Field** list contains all the fields defined in the project.

The buttons beside the **Field** list allow you to create new fields or copy, delete, or edit the selected field.

DESTINATION TAB

Tabs allow you to group sets of related fields.

LABELS

A field has two labels. One label is the field name, which appears in choice lists such as the Field list in any of the Vizor Admin editors. The other label is the caption that appears beside the field in a Web view.

DATA

Under **Data**, you specify what type of input the field accepts: text, dates, times, numbers, yes/no values, or choices from a list.

The choices displayed in a choice list field are defined in a choice table. To edit and create choice tables, click the browse  button adjacent to the **Table Containing Choices** list.

FIELD IS

The **Field Is** check boxes give you control over how the field behaves in a Web view. **Required** fields cannot be left blank. **Visible** fields appear in a Web view, instead of being hidden. **Enabled** fields accept user input, while disabled fields are read-only (they display data but do not accept user input).

The **Read-Only For** list specifies which user groups cannot edit the field.

DEFAULT VALUE

Prepopulate field values with defaults to save time and define standards.

Default field values automatically insert the value of the field when a new record is created.

You can use **exact values** or a **formula**.

Exact values can be:

- A number: 45
- Text: Enter text here

Formulas can use:

- Use the value of another field in the same record:

<F.ID:Owner>

This macro will set the field to the value of the Owner field

<F.ID:Asset Tag>

This macro will set the field to the value of the Asset Tag field

- Use a macro:

<Today>

- Use a combination of macros

<Asset Purchased Date><Asset Type Abbrev><Asset Type Count>

This will set the field to a concatenation of the 3 macros resulting in something like this:

21-11-2018LAPTOP1844

SHOW IN CHOICE LISTS

This check box controls whether you can use the field to build queries, reports, and other styles.

MAINTAIN REVISION HISTORY

This check box controls whether Vizor tracks revisions to the field, and whether you can use the field for update notifications (update notifications are generated when the value of a field changes).

ABOUT ADDING FIELDS

You can add fields to any of the tabs. New fields are automatically available in all Vizor editors, such as the Query Editor and Report Editor.

New fields are not automatically exported to Web views. You must use the Web View Editor to export the fields and regenerate the Web views.

To use a field as a starting point for a new field, click an entry in the **Field** list and click .

Making Fields Required

By default, users can leave the Web view fields blank when they submit an issue. To force users to complete a field before saving an issue, click the **Required** check box. The field label is highlighted in Web views to show that it is mandatory.

MAKING FIELDS READ-ONLY

To make a field read-only for all users, clear the **Enabled** check box.

To make a field read-only for a specific user group, click the **Enabled** check box, and then click the group in the **Read-Only For** list.

USING FIELDS TO DEFINE STYLES

If you don't need to use a field to build queries, sorts, layouts, reports, or notifications, clear the **Show in Choice Lists** check box. This removes the field from the **Field** lists of editors, such as the Query Editor and the Ad-hoc Query Editor of a Web view.

For example, the **Attachments** field is not needed for queries, sorts, layouts, reports, or notifications. How often would you expect users to search for or sort issues based on the files attached to an issue?

TRACKING CHANGES

If you don't want to track changes to a field, clear the **Maintain Revision History** check box. Vizor doesn't update the revision history when a user changes the field.

For a memo field, Vizor doesn't track the changes to the field value. Instead, it marks the field as modified.

Notes

- To send notifications when a field is updated, select the field's **Maintain Revision History** check box.

CHOOSING A DESTINATION TAB

The **Destination Tab** specifies the tab to which the new field is added.

Fields added to the **Contact tab** are treated in a slightly different manner. Depending on how the Web view is configured, the fields could appear in two locations:

- **Contact tab**
 - The Contact tab shows information about the current contact, while preventing the viewer from making modifications. It is functionally equivalent to a regular tab - the only difference is that all fields are automatically read-only.
- **View Contacts Details dialog**
 - For more information, see [Viewing Contact Details](#).

By default, fields assigned to the **Contact tab** will appear in the **View Contacts dialog**. To control whether or not they should also appear in a physical contact tab, modify the **Contact Tab in Record Details** Web View Attribute. For more information, see [Setting Web View Attributes](#).

SETTING FIELD LABELS

Field Caption is the label displayed by the field in a Web view. You can override this label in the Web View Editor by setting the **Caption** attribute.

Choice Lists is the label that appears in any choice list that allows a user to choose a field (for example, in the **Field** list of the Ad-hoc Query editor of a Web view). To change this label, click  to rename the field.

CHOOSING A FIELD TYPE

CURRENCY, DATE, NUMBER, TEXT, AND TIME

These are text entry fields. Vizor validates the text entered in these field types, and displays a warning if users enter invalid characters or uses the wrong format.

For example, **Number** fields accept only numeric values, and **Currency**, **Date**, and **Time** fields require the input values to follow the Windows Regional Settings.

MEMO

A **Memo** field is a text field that allows a large amount of characters to be input.

SINGLE CHOICE AND MULTI CHOICE

These fields are drop-down choice lists. Choice lists present fixed lists of choices to the user. **Multi Choice** fields allow a user to select one or more choices.

YES/NO

This is a check box.

Advanced

Internally, the Data Types are defined using the following IDs (for example, in tblDtsFields).

Data Type	ID	Notes
Boolean	1	
Currency	5	
Date	8	

Text	10	
Memo	12	
Time	101	
Number	102	Sub type (size): Byte 2 Integer 3 Long Integer 4 (default) Single 6 Double 7
SingleChoice	103	
MultiChoice	104	
SCC	105	
RelatedRecord	106	
DateTime	109	
SubRecordLink	110	The main field that links to a subrecords table
Password	115	
HTML	120	
SQLRowVersion	121	Binary, timestamp type in SQL Server

SETTING THE FIELD SIZE

For **Text** fields, the field size is the maximum number of characters that a user can type in the field. For **Number** fields, however, the size is determined by the selected data size:

Size	Description
Byte	Stores numbers from 0 to 255 (no fractions). Decimal precision: None Storage size: 1 bytes
Integer	Stores numbers from –32,768 to 32,767 (no fractions). Decimal precision: None Storage size: 2 bytes
Long Integer	Stores numbers from -2,147,483,648 to 2,147,483,647 (no fractions). Decimal precision: None Storage size: 4 bytes
Single	Stores numbers from –3.402823E38 to –1.401298E–45 for negative values and from 1.401298E–45 to 3.402823E38 for positive values. Decimal precision: 7 Storage size: 4 bytes
Double	Stores numbers from –1.79769313486231E308 to –4.94065645841247E–324 for negative values and from 1.79769313486231E308 to 4.94065645841247E–324 for positive values. Decimal precision: 15 Storage size: 8 bytes

ABOUT REMOVING FIELDS

To remove a field, you can either delete it or hide it. Alternatively, you can choose not export the field when you generate your Web views.

You can also disable fields by [making them read-only](#).

DELETING FIELDS

Deleting a field removes it and all stored data from the issue database. If you don't want to lose the data, you can disable the field, hide it, or remove it from the Web views.

You can delete a field only if it is not used in any query, sort, layout, report, or notification.

HIDING FIELDS IN ALL WEB VIEWS

Because you can remove a field from a Web view by not exporting the field, the ability to hide fields is not often required. However, you may want to hide a field if some of the queries you export use the field, but you don't want users to see it.

To hide a field in all Web views of a project, clear the [Visible](#) check box in the Field Editor. To hide a field in specific Web views, use the **Visible** attribute of the field in the Web View Editor.

NOTES

- Hidden fields are still available in editors unless you clear the **Show in Choice Lists** check box.
- Queries do not work if fields used in the query definition are not in the Web view.

REMOVING FIELDS FROM WEB VIEWS

Using the Web View Editor, you can remove fields from some Web views and leave the fields in other views.

CHOICE LISTS

A choice list is a list of choices displayed in a selection list on a form, in particular, they are used to select the choices for single choice fields and multi choice fields. The choices are defined in a choice table.

EDITING CHOICE TABLES

You can edit the choices in the list from Vizor Web Admin.

Go to <http://<server>/connectadmin/>

Click on the Fields tab.

Click on "Choice Lists" on the left.

In Project, select either the name of the project where your choice table or associated field is, or "Global Choice Lists" if it's a global choice list.

The Choice Lists editor allows you to add new choices, delete choices, change choice text, and reorder choices.

After your changes, we also recommend clicking on the button "Update Now" located on the bottom right. This will refresh the choice list in the views where it's being used right away.

Use Vizor Admin when you want to create new

Using the Field Editor, you can associate a choice table with a field.

Typical examples of choice lists are **Status** or **Progress** in HelpDesk.

You can also delete complete choice lists when they are not referenced by a field.

EDITING CHOICE TEXT

When you change the text of a choice, you must update any queries that test the choice value. For example, consider this query:

```
Progress = Assigned
```

If you change the choice text from **Assigned** to **InProgress** in the choice table, the query will not find any issues. Issues store the index of the choice, but queries store the text.

RENAMING CHOICE LIST TABLES

Vizor Web Admin (in the **Choice List** list on the **Choices** tab) displays a name associated with the choice table, not the name of the choice list field.

A choice table has two names: the name of the choice table (for example, **tblState**) and a friendly name (for example, **State**). Vizor Web Admin displays the friendly name. You can change the friendly name by renaming the choice table in Vizor Web Admin.

SORTING CHOICE LISTS

Web Views can list choices in any order. For example, a non-alphabetic choice order such as Highest, High, Medium, Low, Lowest is more appropriate for a list of relative values than the alphabetic order.

When you create a new choice list, you choose how you want to order the list. The **Allow Choice Order Sort** check box controls whether a choice list supports non-alphabetic choice orders and sorts.

By default, Vizor lists choices in alphabetic order. However, when you allow choice order sorts, you can define the sort order by assigning a numeric ID to each choice. This ID controls the position of the choice in the list (1st, 2nd, and so on).

Choice List:

ID	Choice ID	Choice
1	1	Highest
2	2	High
3	3	Medium
4	4	Low
5	5	Lowest

EDITING THE PROGRESS AND STATE LISTS

The **Status/Progress** choice list is a special type of choice list. When a user selects a choice from the **Progress** list, the **State** field is set to **Open** or **Closed**.

To change the mapping of Status values to State values, use the same editor. The choice table for the **Status/Progress** list contains an extra column that maps **Status** choices to **State** choices. You must complete this column when you add or edit the choices in the **Status** list.

Choice List:

Choices:

ID	Choice	Ticket Status
13	Call Next Day	Open
14	Client Called 2nd Time	Open
15	Client Called Back	Open
16	Closed	Closed
17	Closed - Client Resolved	Closed
18	Closed - Duplicate Call	Closed
19	Escalated	Open

When you edit the **State** choice table, remember that the **Closed** choice sets the **Closed Date** and **Time** fields. So, for example, changing the choice text from **Closed** may confuse users.

Notes

If you edit the **Status** choice list, check that workflow rules based on the **Status** field are still correct for your needs.

GLOBAL CHOICE LISTS

Global choice lists are shared by all projects.

TO DEFINE A GLOBAL CHOICE LIST

- 1 In Vizor Web Admin, click the **Choices** tab.
- 2 In the **Project** list, click **Global Choice Lists**.
- 3 Click  to create a new choice list.

Alternatively, to base the new list on an existing list, select an entry in **Choice List** and click .

- 4 In the Add Choice List dialog:
 - a In the **Choice List Name** box, type a name for the database table in which you want to store the list choices.
 - b In **Choice List Caption**, type a display name for the choice list (this is the name displayed in in Vizor Web Admin).
 - c Select the **Non-Alphabetical Choice Order** check box if you want to define a non-alphabetical order for the choices in the list (for example, **Applications, System, Hardware**).
 - d In the **Choices** section, click  to add a new choice to the list.

To associate the new choice list with a field, use the Field Editor in Vizor Admin, then regenerate the Web views that use the field.

WORKING WITH TABS

Vizor Web Admin allows you to add new tabs to a project. After you add a new tab, use the Field Editor to add new fields to the tab.

You can also reorder the tabs, change the tab names, and delete tabs. To delete tabs you must delete all the fields on the tab or move the fields to another tab.

CHAPTER 17 - E-MAIL INTEGRATION

E-MAIL INTEGRATION

In addition to enabling users to submit issues through Web views, you can configure Vizor to directly accept issues submitted by e-mail, as well as keep track of all e-mail conversations associated to the issue.

This provides an easy-to-use interface with which most users are already familiar, and enables support staff to manage e-mail exchanges with users without having to spend time entering the details of conversations.

To configure your system to accept e-mail issues:

Open Web Admin using one of the following methods:

In your Web browser, enter *//server/Connectadmin*, where *server* is the name of your Web server.

In Vizor Admin, click Tools > Web > Web Admin.

At the Web server, click Start > All Programs > Vector > Vizor > Vizor Web Admin.

Log on as a user that belongs to the Vizor **Administrators** group.

In Web Admin, click the **E-mail** tab and click **Incoming Mail**.

Select the **Project** for which you want to enable e-mail integration.

INCOMING E-MAIL INTEGRATION PROCESS

Configuring Vizor to accept issues submitted by e-mail is a multi-stage process:

Click **Add a new Integration** to enter the details of an e-mail address you want to use to submit issues to the project and keep the e-mail exchanges. The **Add E-mail Integration Account** window is displayed.

For each e-mail address, you want Vizor to monitor, you must specify:

The information that will enable Vizor to access the account. (See " Specifying the E-mail Integration Account".)

The rules that govern how incoming e-mails are handled. (See " Mail Handling".)

The information you want to copy from the e-mail to the associated issue, and default values for non-mapped fields. (See " Mapping E-mail Fields to Issue Fields".)

When you have completed entering the details of the **E-mail Integration Account**, select **Enable this integration** , and click **OK** to save your changes.

SPECIFYING THE E-MAIL INTEGRATION ACCOUNT

Before Vizor can convert e-mails to issues, you must specify the addresses that you want to use for submitting issues.

To specify the details of the e-mail account you want to integrate with Vizor:

- 1 In the Hostname field, enter the name or IP address of the POP e-mail server that handles incoming e-mail.**
- 2 In the Port field, enter the port through which the e-mail server communicates.** By default, POP uses port 110.

If the POP server requires an encrypted connection, check the **SSL** box, and change the port accordingly. By default, POP SSL uses port 995.

- 3 Enter a Logon and a Password for the account that Vizor will use to access the e-mail server.**

Add E-mail Integration Account

Incoming E-mail Server (POP)

Issue Tracker can be configured to monitor specific E-mail addresses and automatically create issues from the e-mails sent to it. Enter the details for an address you want Issue Tracker to monitor.

Account Information

E-mail Address:

Server Information

Hostname:

Port:

SSL: ☐

Logon Information

Logon:

Password:

Test

Mail Handling

Mail-handling rules determine how messages sent to this account are converted to issues. The rules are processed in order. If all the conditions of the rule are met, the actions of the rule are executed, and no more rules are evaluated. If all conditions are not met, the next rule is evaluated.

Rule Name	Action Type
Send Status Report	None
Link to Issue	Link
Create new Issue	Create
Add E-mail to Queue	Queue

Add...

Edit...

Copy...

Disable

Remove

▲

▼

Issue Completion

Issue-completion rules determine the values that will be given to each field in an issue. Click Edit Mapping to specify the issue fields that can be copied from the e-mail. Click Edit Defaults to specify default values for non-mapped fields.

Edit Mapping...

Edit Defaults...

☐ Enable this integration.

OK

Cancel

INCOMING E-MAIL RULES

E-mail-handling rules determine how e-mails sent to the address specified in the **Incoming E-mail Server** section are processed.

When an e-mail is received, Vizor compares the e-mail fields with the conditions specified in the first entry of the rules list. If the e-mail matches all conditions specified in the rule, Vizor executes the specified actions, and then processes the next e-mail. If any of the conditions in the rule are not met, Vizor tests the e-mail against the conditions specified in the next rule, until it reaches the end of the rule list.

It is recommended to place a rule that has an **Action Type** of **Queue** as the last rule in the list to cover the possibility that the e-mail does not meet the conditions of any rule in the list, in which case, the e-mail will be automatically added to the **E-mail Queue** for manual processing.

Edit Rule

Rule Name:

When all these conditions are met...

☒	Subject	Contains	New
☐	<Field>	Is Equal To	
☐	<Field>	Is Not Equal To	
☐	<Field>	Contains	
☐	<Field>	Doesn't Contain	
☐	<Field>	Is Empty	
☐	<Field>	Is Equal To	
☐	Has Attachments		
☐	E-mail contains number identifier (Issue ID: #) of existing issue in <Field> field		

Perform the following actions...

☒	Integrate e-mail with the Vector system:
☒	Create New Issue using these values
☐	Send to Queue
☐	Link to Issue
☒	Send an e-mail to <Reply To Sender>
☐	Send an e-mail to <None>
☐	Run program

Once an e-mail matches a Rule's conditions, that Rule's actions are executed. Afterwards, the next e-mail in the list is considered - without evaluating the remaining Rules.

OK Cancel

To create an e-mail rule:

In the **Mail Handling** section, click the **Add** button. The **Add Rule** window is displayed.

Enter a **Rule Name**.

Enter the conditions you want the rule to test for. For example, to test the *Body* of the e-mail for the word *problem*:

- 1 Enable the first empty condition by selecting its check box.
- 2 Click **<Field>** and select **Body**.
- 3 Click **Is Equal to** and select **Contains**.
- 4 Type *problem* in the entry field.
- 5 Next, enter the actions you want the system to perform when all the conditions are met.

For example, to configure the system to create an issue and send a reply to the person who sent the e-mail:

- a** Select the Integrate e-mail with Vizor check box.
- b** Select Create New Issue using these values.

*By default, Vizor completes the issue using the field mappings defined in the **Issue Completion** section of the **Edit E-mail Integration Account** window. To override these mappings for this rule only, click **these values**. For more information, see “Mapping E-mail Fields to Issue Fields”.*

- c** Select **Send an e-mail to <Reply to Sender>** to send an acknowledgement to the submitter. (For more information, see “Defining E-mails”) .

- 6** Click **OK** to save the rule. The rule is added to the bottom of the mail-handling list.

- 7** Select the rule and click the **Up** and **Down** arrows to specify the processing order for the rule.

In most systems, the first rule checks whether an e-mail refers to an existing issue to avoid the possibility of creating duplicate issues. The easiest way to create this rule is to specify a condition that checks whether the e-mail contains a reference to an existing issue. If there is no match, only then should the system go on to check whether the e-mail contains details of a new issue.

USING REGULAR EXPRESSIONS IN THE EMAIL RULE CONDITIONS

You can use Regular Expressions in the Email Rule Conditions. For example, you may want to have a rule that matches a subject with the word “hello” in multiple languages. The condition will be something like:

subject contains hello|hola|bonjour

where the character “|” is used as a regular expression OR.

Another example could be:

subject contains [nN]ew ticket

where the rule will match both “New ticket” and “new ticket” in the subject.

Regular Expressions are **only** enabled automatically in email rule conditions that use the operator **contains** , which then causes the value that follows to be processed like a regular expression.

OUTGOING E-MAIL SETTINGS

In order for the Vizor E-mail Integration to process incoming and outgoing e-mails, several Mail Settings must be configured in Vizor.

Outgoing E-mail Server (SMTP)

Hostname: Port: SSL:

Logon: Password:

Request Read Receipt?

Send Automatic E-mail Notifications

From:

Every: minute(s)

Failed Outgoing E-mails

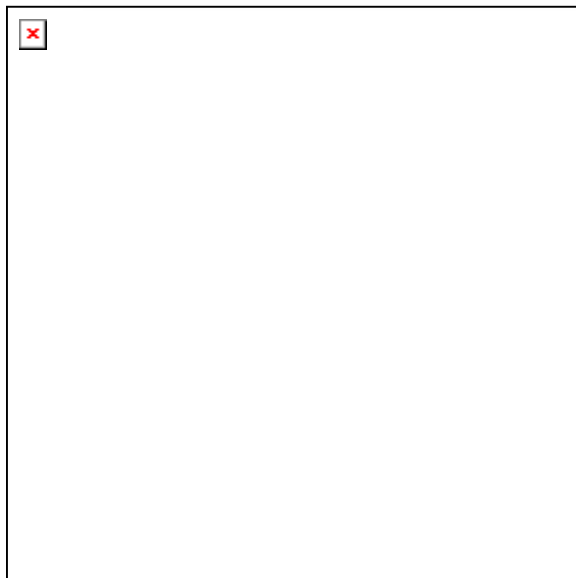
Maximum Retries:

Send to:

Incoming E-mail Server (POP)

Check Server For New Mail

Every: minute(s)



TO CONFIGURE THE MAIL SETTINGS:

Log on to Vizor Web Admin. Click the **E-mail** tab and the **Mail Settings** section.

Fill in the Outgoing E-mail Settings. These settings are used when sending e-mails.

Click **Test** to ensure Vizor is able to connect to the SMTP server using the settings you provided.

Option	Description
Hostname	The hostname or IP address of the SMTP server. For example: smtp.mycompany.com
Port	The port through which the SMTP server communicates. The default SMTP server port is 25.
SSL	Vizor supports SSL, enabling you to use secure connections to your email server. Selecting Yes from the drop down list will ensure that Vizor communicates utilising the SSL protocol.
Logon	The logon account used to gain access to the SMTP server. Use this only if your SMTP server requires authentication.
Password	The password for logging into the SMTP server. Use this only if your SMTP server requires authentication.
Request Read Receipt	Read receipts will appear in the conversation, confirming that the recipient has read the message. Note, however, that some recipients may choose to never send read receipts.
Send Automatic E-mail Notifications From	Name of the sender for SMTP e-mail. Some SMTP mail servers require the sender to be a valid e-mail address.Supported Formats: "Notification Service" <notificationsservice@yourcompany.com> notificationsservice@yourcompany.com

Send Automatic E-mail Notifications Every	Send e-mail notifications at the specified interval (in minutes).
Maximum Retries	The maximum number of retries for e-mail notifications. Once surpassed, the e-mail will be sent to the address specified in the Send Failed Outgoing E-mails To.
Send Failed Outgoing E-mails To	If the Maximum Retries is surpassed, the e-mail will be sent to this address. If there is no address specified, the e-mail service will continue trying to send to the original recipient.

Fill in the **Incoming E-mail Settings**. These settings are used when receiving e-mails, together with the **Account**, **Server** and **Logon Information** specified in the defined Mail Integrations. For more information, see [Setting Up E-mail Integration](#).

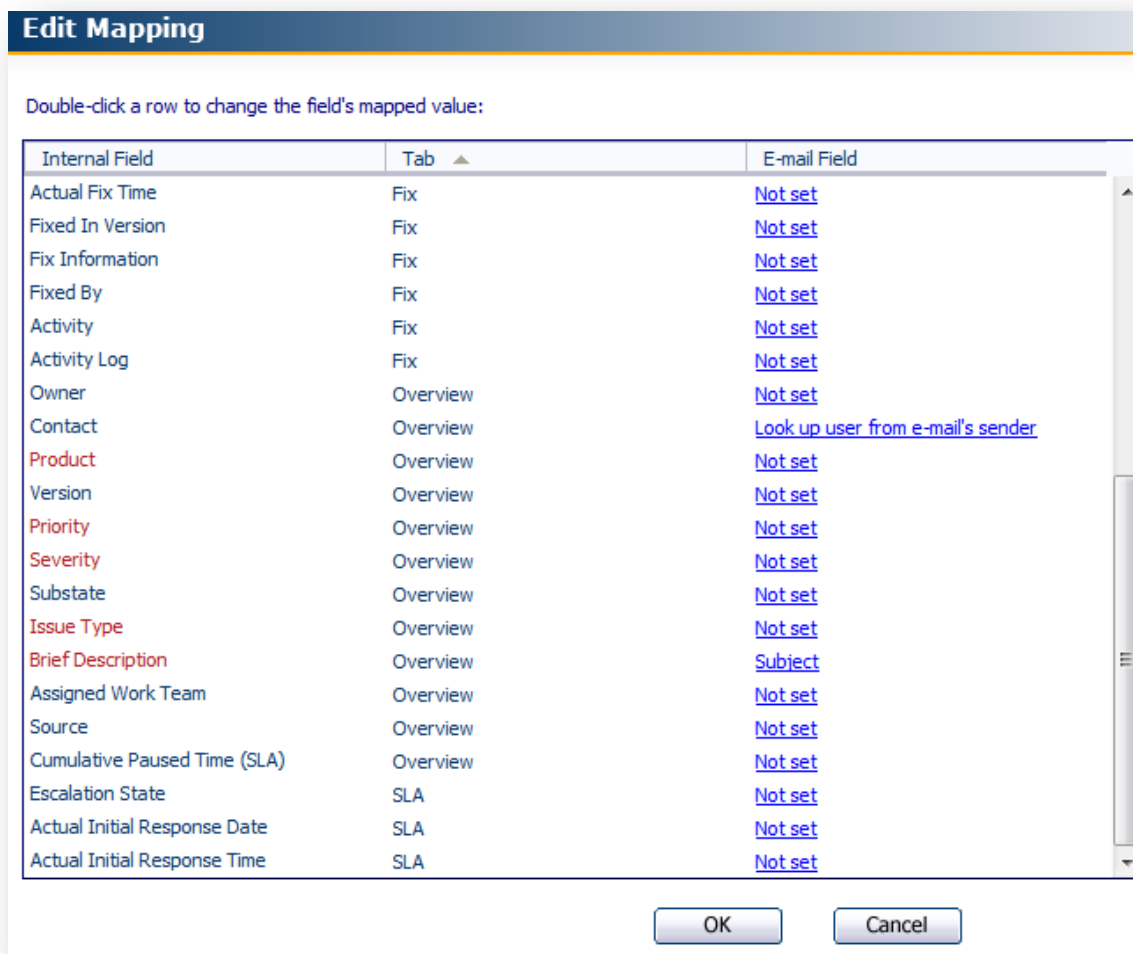
Option	Description
Check Server For New Mail Every	Check for new incoming e-mails at the specified interval (in minutes).

MAPPING MESSAGE FIELDS TO ISSUE FIELDS

When an e-mail is sent to a monitored address, Vizor generates an issue and copies information from the e-mail to specified issue fields. Vizor automatically creates a default mapping between the e-mail fields and specified issue fields when you integrate an account. For example, the *body* of the e-mail is copied to the issue's *Description* field. However, you can change the fields that the e-mail information is copied to using the **Edit Mapping** button.

To view the field mappings for the specified account:

Click the **Edit Mapping** button. The **Edit Mapping** window lists the fields in your selected project, with mandatory fields shown in red.



Edit Mapping

Double-click a row to change the field's mapped value:

Internal Field	Tab ▲	E-mail Field
Actual Fix Time	Fix	Not set
Fixed In Version	Fix	Not set
Fix Information	Fix	Not set
Fixed By	Fix	Not set
Activity	Fix	Not set
Activity Log	Fix	Not set
Owner	Overview	Not set
Contact	Overview	Look up user from e-mail's sender
Product	Overview	Not set
Version	Overview	Not set
Priority	Overview	Not set
Severity	Overview	Not set
Substate	Overview	Not set
Issue Type	Overview	Not set
Brief Description	Overview	Subject
Assigned Work Team	Overview	Not set
Source	Overview	Not set
Cumulative Paused Time (SLA)	Overview	Not set
Escalation State	SLA	Not set
Actual Initial Response Date	SLA	Not set
Actual Initial Response Time	SLA	Not set

OK Cancel

To map an e-mail field to a project field:

Double-click the row of the project field you wish to map. The **Edit Mapping Field** window is displayed.

In the **Edit Mapping Field** window, select the E-mail Field you want to map to the selected project field, and click the OK button.

To undo a field mapping:

In the **Edit Mapping** window, double-click the mapping you want to cancel.

In the **Edit Mapping** Field window, deselect the **Assign value to this field** check box and click the OK button.

Click the **OK** button to save your changes, and close the **Edit Mapping** window.

In addition to the usual E-mail Fields, a special field called **Lookup user from e-mail's sender** is included. If a project field is mapped to this field, the system will use the incoming e-mail's **From** address in an attempt to find an existing Vizor user or contact who matches the credentials. When looking for a match, the system will consider all e-mail addresses associated with the user - not only the primary address. (See [Associating Multiple E-mail Addresses to a Single Contact](#))

If a user is found, that user will be associated with the e-mail. Otherwise, a contact may be created automatically if the option **Create a contact based on the "from" address** in the Default Values is set (See [Setting Default Values for Issues Submitted by E-mail](#)).

For example, a typical mapping could be like this:

E-mail Field	Vizor Field
Subject	Summary
Body	Description Log
From	Instead of mapping From to a Vizor field, you can use the <Look up User from sender's e-mail> macro to fill in the field with Vizor user ID.
To	Contact (this would allow you to search for issues submitted by e-

	mail)
--	-------

SETTING DEFAULT VALUES FOR ISSUES SUBMITTED BY E-MAIL

While the Edit Mapping window will allow associating e-mail fields to Vizor project fields, it is possible to define a separate set of values that will be applied along with the imported e-mail fields. When Vizor fields are not mapped to e-mail fields, or when those mapped values are empty, the default values are considered. There are two different types of Default Values:

Global Default Values are set in the **Issue Completion** section of the **E-mail Integration Account** window.

Rule Default Values can be set within any specific Rule that has **Create** as an **Action Type**, and if specified, they can override the **Global Default Values**.

For example, you could have a **Global Default Value** that sets the Vizor field *Priority* to *Medium*. This may be acceptable in the majority of cases, but for the sake of this example, let's say a Rule is created to treat e-mails sent from the CEO to the Vizor system. In this case, you may want the issue *Priority* to be set to *High*, instead of *Medium*, but still use all other existing **Global Default Values**. By using the **Rule Default Values**, special cases like the one just described are possible.

Default Values for the E-mail Rule

When an issue is created, the e-mail values will be used to determine some of the Issue Tracker field values. When Issue Tracker fields are not mapped to e-mail fields, or when those mapped values are empty, the default values are considered. If there are default values specified *within this specific rule*, they will override any global default values. The Global Default Values are accessible from the Issue Completion section of the E-mail Integration Account window.

Double-click a row to change the field's default value:

Internal Field ▲	Mapped To	Default Value	Using Global?
Brief Description	Subject	(no subject in e-mail)	✓
Contact	Look up user from e-mail's sender	Not set	
Cumulative Paused Time (SLA)		Not set	
Description Log	Body	Not set	
Detailed Description		Not set	
Duplicate Issue		Not set	
Escalation State		On	✓
Estimated Fix Time		Not set	
Fix Information		Not set	
Fixed By		Not set	
Fixed In Version		Not set	
Fixed On Platform		Not set	
Functional Area		Not set	
How Found		Not set	
Issue Type		Documentation Enhancement	✓
Owner		Not set	
Platform		Not set	
Priority		High	✓
Product		Census	✓

☒ Create a contact based on the "from" address if one doesn't already exist.

OK Cancel

If you mapped **Lookup user from e-mail's sender** to any Vizor field in the **Edit Mapping** window, selecting the **Create a contact based on the "from" address if one doesn't already exist** checkbox will enable the auto-creation of contacts when no match is found in the system. (See [Mapping Message Fields to Issue Fields](#))

When a contact is created, Vizor will analyze the "from" address, and attempt to automatically associate the contact with a company. To achieve this, Vizor looks at the e-mail domain (ex: bob@mycompany.com), and looks for a company that has a similar e-mail address. It knows to ignore common e-mail domains like Gmail, Hotmail, Yahoo, etc. To make modifications to the invalid e-mail domain list, see [Ignoring Specific E-mail Domains When Creating Companies](#).

SENDING AUTOMATIC E-MAILS

From both the E-mail Integration Rules and the SLA Escalation Rules, you can send an e-mail message automatically as an action.

TO SEND AN AUTOMATIC E-MAIL AS AN SLA ESCALATION RULE ACTION:

Log on to Web Admin.

Click the **SLA** tab and the **Service Levels** section. Select a Project and a Service Level and click the **Escalation Rules** tab. Load an Escalation Rule, and click **Send an e-mail** as an Action.

Enter the e-mail information and click OK. This e-mail will be sent when the conditions of the SLA Escalation Rule are met.

Tip: Automatic SLA Escalation Rule E-mail Actions are useful for informing the Owner of when they need to resolve an issue by.

TO SEND AN AUTOMATIC E-MAIL AS AN E-MAIL INTEGRATION RULE ACTION:

Log on to Web Admin.

Click the **E-mail** tab and the **Incoming Mail** section. Select a Project load an E-mail Integration. Load a Mail Handling Rule, and click **Send an e-mail** as an Action.

Enter the e-mail information and click OK. This e-mail will be sent when the conditions of the Mail Handling Rule are met.

Tip: Automatic E-mail Integration Rule E-mail Actions are useful for replying to the person who sent the e-mail to tell them the number of the issue that the e-mail created so they can follow up on it at a later time.

SENDING E-MAILS FROM THE WEB VIEWS

TO SEND AN E-MAIL CONTAINING INFORMATION ABOUT AN ISSUE:

Log on to a Web view.

Load an issue and click on the  button.

The E-mail Editor will open, with the default Current Issue **Template**. You can add the value of additional fields using the Field Value **Macro**.

Click the Send button.

This e-mail will be added to the issue's Conversation if the Save to Conversation checkbox is selected.

SERVICE LEVEL AGREEMENTS (SLA) AND E-MAILS:

If the issue has an SLA agreement associated with it, the first time you send an e-mail from Vizor to the **Contact** of the issue, the **Actual Initial Response Date** and **Actual Initial Response Time** fields in the issue will automatically be set to the date and time when the e-mail is sent (if they aren't set already).

USING THE E-MAIL EDITOR

The E-mail Editor in Vizor allows you and your users to create and send e-mail messages from the Web view. You can also configure e-mail to be sent automatically from SLA or E-mail Integration (for example, when Vizor creates a new issue from an incoming e-mail, or when an SLA rule changes the Owner of an issue).

E-mail Content Editor - Microsoft Internet Explorer

Customize the e-mail that should be sent for this action.

To:

Cc:

From:

Subject:

Attachments:

A new issue has been assigned to you. Please respond to issue #<Issue Number> in the <Project Name> project by <MqField:dTargetIRT>.

Summary: <MqField:tBriefDescription>

Contact Name: <MqField:nOriginatorID>

Target Closure Time: <MqField:dTargetClosureTime>

☐ Send e-mail to each recipient individually ☐ HTML in Body

Macros

Available Macros

Click on a macro below to add it to the e-mail.

- Field Value
- Link To Issue
- Issue Number
- Project Name

OK Cancel

SENDING E-MAILS

EDITING THE TO, CC, AND FROM FIELDS

You can specify users by either looking up users and contacts in Vizor, using **macros** or typing in the e-mail address directly.

To lookup users and contacts in Vizor, beside the To, CC, or From box, click .

In the Select Users and Contacts dialog box, double-click the users and contacts you want to add.

The list of selected users and contacts is displayed under **Selected Users and Contacts**. You can edit this string of text, for example, to remove a user, or to add the e-mail address of a person who is not a user or contact.

NOTES

The text box beside the Find button allows you to display all users and contacts whose names start with the same letters. In the text box, type the first few letters of the name and then click Find.

To display the complete list of users and contacts again, delete all text from the box and then click Find.

Click Advanced to search for all users and contacts whose names contain the same string, or to search based on e-mail addresses.

If you would like to send separate e-mails to several recipients, select the Send e-mail to each recipient individually checkbox. If you have recipients entered in the CC field, you will not be able to send individual e-mails.

EDITING E-MAIL CONTENT

You can type in any text you want, and you can use **macros** to insert the issue number, the name of the project, and the values of different fields (such as the Summary and Description Log).

If you prefer your e-mail body to be in HTML format, click the **HTML in Body** checkbox.

ATTACHING FILES

Beside the Attachments box, click .

In the E-mail Content Attachments dialog, click Browse and locate the file you want to attach. Click Upload or Link.

This adds the file to the list of files attached to this e-mail message. Every time the e-mail message is sent, the files are attached to the message.

E-MAIL CONVERSATION AND QUEUE

E-mail integration brings the ability to view support-related and other e-mails directly from the Web views. The E-mail **Conversation** and **Queue** are where you can **view and manipulate** these integrated e-mails.

THE E-MAIL CONVERSATION

The E-mail Conversation is where you can access all the e-mails related to a specific issue.

To access an issue's E-mail Conversation:

Log on to the Web view where the issue can be found.

Find the issue using the **search** feature.

Select the issue from the summary list.



Click **Conversation** in the toolbar.

THE E-MAIL QUEUE

The E-mail Queue is where you can access and process all the e-mails pending assessment from a help desk analyst. These e-mails have not been associated to any existing issue.

To access the E-mail Queue:

Log on to a Web view of the desired Project. (Each Project has its own E-mail Queue.)



Click **Queue** in the toolbar.

PROCESSING E-MAILS IN THE WEBVIEW

The E-mail Conversation and Queue are where integrated e-mails can be processed, in the Web view. There, you can:

Send a new e-mail.

Reply to an e-mail.

Forward an e-mail.

Link an e-mail to an existing issue.

Create a new issue from an e-mail.

SENDING NEW E-MAILS

E-mails are created and sent using the **e-mail editor** (see [Using the E-mail Editor](#)).

You might be interested, for example, in sending an e-mail and keep a copy in an issue's E-mail Conversation and **Activity Log**.

New e-mails' fields will be completely blank with the exception of the From address which is determined by your [E-mail Options](#), and the Subject which, for the Conversation only, is pre-filled with the respective issue number.

To send a new e-mail from an E-mail Conversation or the E-mail Queue, click .

REPLYING TO E-MAILS

E-mail replies are created and sent using the **e-mail editor** (see [Using the E-mail Editor](#)).

You might be interested, for example, in requesting more information from a customer after receiving a support e-mail.

E-mail replies are pre-filled with the original e-mail's content, and the From address is determined either by the original e-mail's **Integration Account's** (see [Setting up E-mail Integration](#)) e-mail address, or by your [E-mail Options](#) if the original e-mail didn't come from E-mail Integration.

To reply to an e-mail from an E-mail Conversation or the E-mail Queue, click  or , depending whether you want to reply only to the sender or to all the original recipients as well.

Note: see [Sending E-mails From Web views](#) to see how an E-mail Conversation reply can automatically update the respective issue's Actual Initial Response Date and Time.

FORWARDING E-MAILS

Forwarded e-mails are created and sent using the [E-mail Editor](#).

You might be interested, for example, in sending a copy of a customer e-mail to a colleague for second-line support.

Forwarded e-mails are pre-filled with the original e-mail's content with the exception of the To recipient which is left blank. The From address is determined by your [E-mail Options](#).

To forward an e-mail from an E-mail Conversation or the E-mail Queue, click .

LINKING AN E-MAIL TO AN EXISTING ISSUE

Whether an e-mail is already in an issue's E-mail Conversation or still in the E-mail Queue, you might find out it is related to some (other) existing issue.

To link an e-mail to an issue, from an E-mail Conversation or the E-mail Queue, click .

After the e-mail is linked to this new issue, it can be found in this issue's E-mail Conversation.

CREATING A NEW ISSUE FROM AN E-MAIL

Whether an e-mail is already in an issue's E-mail Conversation or still in the E-mail Queue, you might decide a new issue needs to be created with regards to the information found in it.

To create a new issue from an e-mail in an E-mail Conversation or the E-mail Queue, click .

After the issue is created, you will receive a message stating the new issue's number, and the e-mail will now be in this issue's E-mail Conversation.

E-MAIL MACROS

Macros are available when you use the E-mail Editor in the Web views and Web Admin.

Macros resolve values for information related to the issue so you don't have to look it up manually.

Macro	Available In Fields	Description
<Contact>	To, CC, From	The person who is entered in the Contact field in the issue. This is usually the person who reported the issue.
<Owner>	To, CC, From	The person who is entered in the Owner field in the issue.
<Previous Owner>	To, CC, From	The person who was entered in the Owner field in the issue before it was reassigned to a new Vizor user.
<Submitter>	To, CC, From	The person who is entered in the Submitter field in the issue. This is the person who entered the issue in Vizor.
<Work Team Members>	To, CC	The people who belong to the work group that is entered in the Assigned Work Team field in the issue.
<Reply To Sender>	To	The person who sent the e-mail. This only applies when replying to an e-mail through the Conversation or Queue or from

		the E-mail Integration Actions.
<Reply To CC>	CC	The people who were CCed on the e-mail that was sent. This only applies when replying to an e-mail through the Conversation or Queue or from the E-mail Integration Actions.
<Integration E-mail>	From	The e-mail address specified for the E-mail Integration Account.
<Server Mail Sender From Admin Options>	From	The e-mail address specified in the Mail Settings for who to Send Automatic E-mails Notifications From.
Field Value	Subject, Body	The value of a field in the issue.
Issue Number	Subject, Body	The number of the issue.
Project Name	Subject, Body	The name of the project the issue is in.
Link To Issue	Body	Generates a URL that you can include in the e-mail to allow people who have access to the Web views to log in to a specific view and automatically load the issue you specified.

Position your cursor in the e-mail where you would like the resolved macro to be inserted.

Click on the **Macros** tab.

Click on the macro you would like to insert. In the Web views, the macro will be resolved to the current value. In templates and in Web Admin, the macro will remain unresolved so that it can be resolved when the automatic e-mail is sent or when the template is applied to an e-mail.

E-MAIL TEMPLATES

Templates are available when you use the E-mail Editor in the Web views.

Use templates to store reusable e-mail content so that your users can quickly create e-mails when standard responses are required. A template can contain Subject, Body and Attachment information.

TO CREATE AN EMAIL TEMPLATE

All users who have access to the Web View – Manage E-mail Templates can create and edit templates.

To create a template, open the E-mail Editor to create a new e-mail. Click the **Templates** tab and click **Add a new template**.

By default, the new template is filled in with any subject, attachment and body text that was entered in the e-mail you added the template from.

Enter a **Name** for the template, as well as any subject, attachment or body text that you want in the template.

Click OK, and your new template will be available in the Templates list for all users who have the feature Web View – Manage E-mail Templates.

LINKS WITHIN NOTIFICATION TEMPLATE

It is possible to create a link within an email template. This is most useful for email notifications that use an email template. For example, most notifications have a link at the top right that brings the user to the issue when clicked on. Additionally, notifications can link to a survey a user can fill out once an issue is closed (See **Issue Closed Notification** in the templates tab).

To create a link to an issue, first go to the templates tab and either create a new template or edit an existing one. Enter the text you want the link to have, for example **View the ticket online**, and click on the Link button in the top toolbar (Make sure **HTML in Body** is checked in the bottom right to view the toolbar).

Within the Link Info tab, URL should be selected as the Link Type by default. For the protocol, select http:// or https:// depending on whether or not you have SSL enabled. Next, for the URL, enter something similar to this:

[customer.vizor.cloud/connect?View=HelpDesk%2FEmployee&nId= <Issue Number>&Qry=22](customer.vizor.cloud/connect?View=HelpDesk%2FEmployee&nId=<Issue Number>&Qry=22)

Note that the **<Issue Number>** in the URL is a macro used to get the ID of the issue related to the notification being sent. If you prefer having this value encrypted, use **eId= <Encrypted Issue Number>** instead.

To link to a survey for the related issue, the steps will be the same as mentioned above, except the link will be like this:

<customer.vizor.cloud/connect?View=HelpDesk%2FSurvey&eId= <Encrypted Issue Number>>

You can copy the link from the **View the ticket online** link, but make sure View=HelpDesk%2FEmployee is replaced with View=HelpDesk%2FSurvey, and remove "&Qry=22" from the URL.

TO USE AN EMAIL TEMPLATE

All users who have access to the Web View – Manage E-mail Templates can use templates.

To use a template, open the E-mail Editor to create a new e-mail. Click the **Templates** tab and select the template you want to use. Click **Apply** the selected template.

PERSONAL E-MAIL OPTIONS

The Web view E-mail Options affect all e-mails created from the Web views, be it from the E-mail toolbar button, an issue's [E-mail Conversation](#) or the [E-mail Queue](#).

Two options are available to Web view users regarding e-mails:

The **From address** to be used when composing e-mails.

The requesting of **read receipts**.

By default, every user's E-mail Options are:

From Address: their own e-mail address.

Read Receipts: requested.

To change your E-mail Options:

Log on to a Web view.



Click **Options** in the toolbar.

Click the E-mail tab to open it.

Perform the desired changes and click OK.

From Address

E-mail address that will appear in the "From" field when creating new e-mails within HelpDesk.

☒ **Me** (support@vector-networks.com)

☐ **Specific Address:** support@vector-networks.com

Read Receipts

Read receipts will appear in the conversation, confirming that the recipient has read the message. Note, however, that some recipients may choose to never send read receipts.

Request Receipt? Yes

The changes will be applied the next time you send an e-mail.

OK Close

THE "FROM ADDRESS" E-MAIL OPTION

You have the option to use the e-mail address of your Vizor user account or any other e-mail address, when sending e-mails from Web views.

No matter what this option is set to, replies to e-mails coming from an [E-mail Integration Account](#) will always have the respective account's e-mail address as their From address.

THE "READ RECEIPTS" E-MAIL OPTION

You have the option to request a read receipt for all the e-mails you send from any Web view.

Depending on the From address of your e-mails and the [E-mail Integration](#) configuration, these read receipts can be processed automatically by the E-mail Integration feature.

Note that read receipts can be recognized by their subjects starting with "Read:".

CHAPTER 18 - DEFINING WORKFLOW RULES

ABOUT WORKFLOW RULES

Workflow rules allow you to define and enforce an issue handling process. Together with e-mail notifications, workflow rules help you automate the tracking and management of issues.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[What is a Workflow Rule?](#)

[Creating Rule Templates](#)

[Defining Conditions](#)

[Defining Rules](#)

[Deleting Rules](#)

[Renaming Rules](#)

[Using Macros in Rules](#)

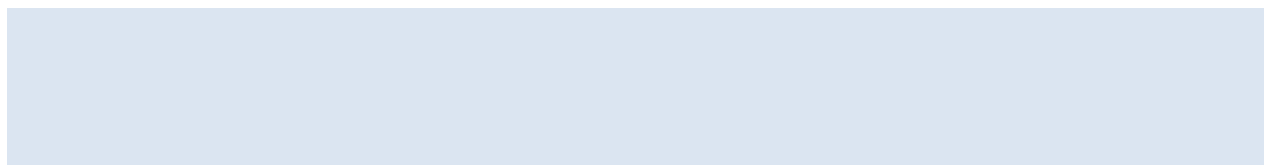
[Setting Possible Values](#)

[Changing When Rules Are Evaluated](#)

[Changing the Order of Evaluation](#)

[Applying Workflow Rules](#)

[Disabling Workflow Rules](#)



WHAT CAN YOU DO WITH WORKFLOW RULES?

When a user selects a choice from a choice list, you can:

- Select choices in other choice lists.
- Change the possible choices in other choice lists.
- Hide fields.
- Disable fields.
- Make fields required.
- Hide tabs.

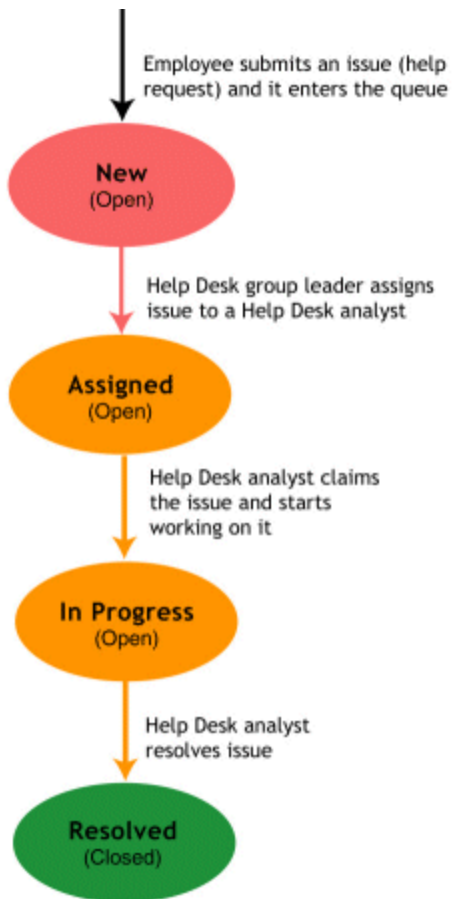
These are the basics of workflow rules. However, this explanation leaves out some important details:

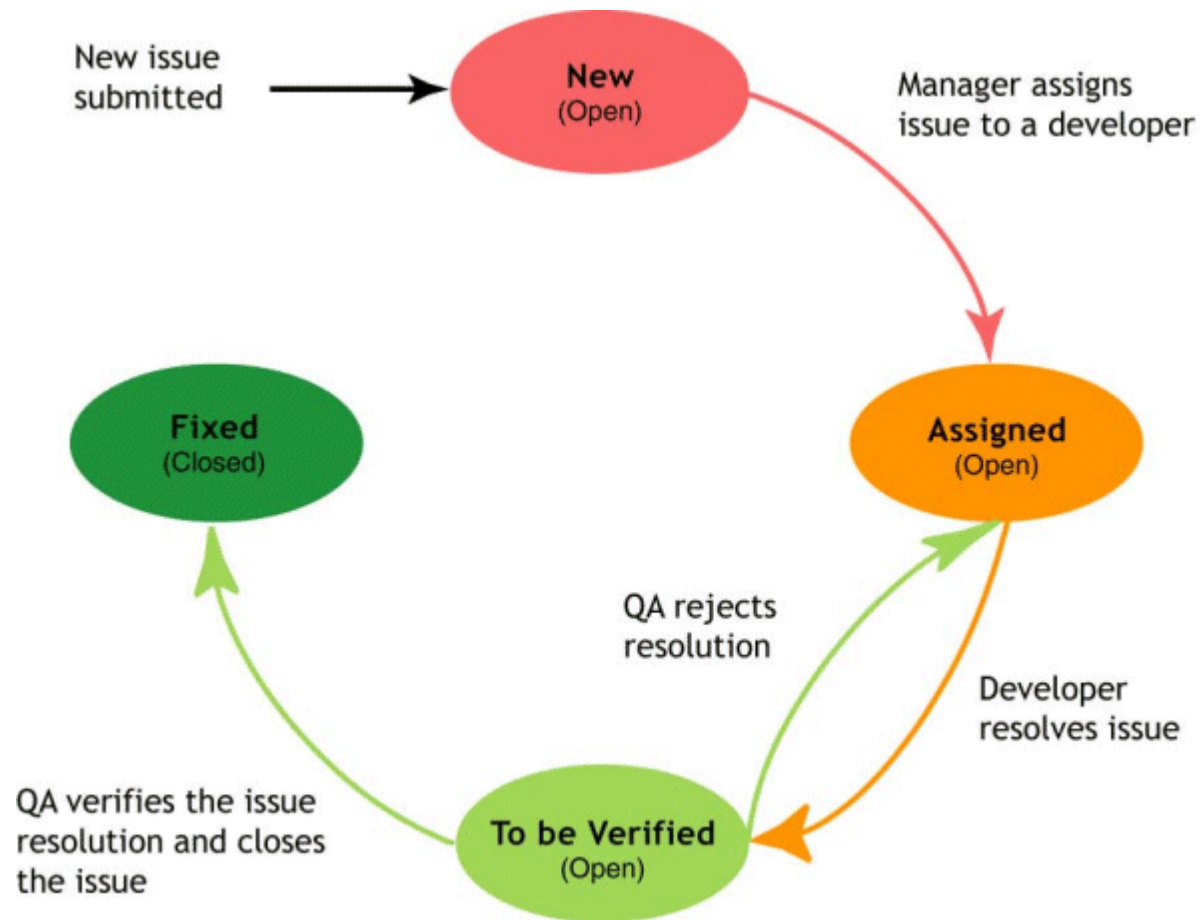
- By default, rules are applied when a user creates, saves, or loads an issue. You can also force rules to be evaluated when a user selects a choice.
- Workflow rules can also be based on more complicated conditions. For example, a rule may require that certain choices are selected from a number of different choice lists, or that the user is a member of a specific group, or both.

AUTOMATE WORKFLOW

By controlling the possible values of the **Progress** field, you can define and enforce a process.

Simple workflow controlled by Progress values





EXAMPLES

EXAMPLE 1:

After an analyst starts work on an issue and marks it **In Progress**, you can enforce a process where only the analyst can mark the issue as **Dropped**, **Resolved**, or **To be Verified**.

When:

<User in Group> = Help Desk Analyst AND

Progress = In Progress

The possible values are:

Progress = Dropped

Progress = Resolved

Progress = To be Verified

Note that **Progress = In Progress** is an implied possible value. A choice list can always be set back to its current value. For example, if a user is not a help desk analyst and an issue is marked **In Progress**, the **Progress** list contains only one choice: **In Progress**.

This rule is an example of a *possible values* rule.

EXAMPLE 2:

You can prevent anyone except a group leader from assigning new issues.

When:

<User in Group> = Help Desk Group Leader AND
Progress = New

The possible values are:

Progress = Assigned

This prevents new issues from being resolved or dropped without first being assigned to an analyst by the group leader. If a user is not a group leader, the only possible choice is **New**.

EXAMPLE 3:

You can enforce a process where developers must forward issues to QA:

When:

<User in Group> = Developers AND
Progress = Assigned

The possible values are:

Progress = To be Verified

And where only QA can close issues (or assign them back to developers):

When:

<User in Group> = QA AND
Progress = To be Verified

The possible values are:

Progress = Fixed
Progress = Assigned

Note that a choice list can always be set back to its current value. For example, if a user is not part of **QA** and an issue is marked **To be Verified**, the **Progress** list contains only **To be Verified**.

EXAMPLE 4:

You can prevent anyone except a manager from assigning new issues.

When:

<User in Group> = Managers AND
Substate = New

The possible values are:

Substate = Assigned

This prevents new issues from being resolved or dropped without first being assigned to a developer by the manager.

CREATE RELATED CHOICE LISTS

You can define rules to create a relationship between two choice lists.

EXAMPLES

If **Problem Type** can be **Hardware**, **Software**, or **System**, **Problem Area** can be a type-specific list that displays a list of hardware components, software applications, or system components.

When:

Problem Type = Hardware

The possible values are:

Problem Area = Disk

Problem Area = Monitor

Problem Area = Keyboard

Problem Area = Memory

You need to create similar rules for software and system problems.

By default, workflow rules are evaluated when an issue is saved. For related choice lists like the ones in this example, you probably want to evaluate the rule whenever the **Problem Type** changes. Otherwise, users would have to select a **Problem Type**, save the issue, and then select the **Problem Area**.

DEPENDING ON THE TYPE (SOFTWARE DEFECT, HARDWARE DEFECT, OR

DOCUMENTATION DEFECT) YOU CAN UPDATE THE FUNCTIONAL AREA LIST.

When:

Type = Documentation Defect

The possible values are:

Functional Area = Online Help

Functional Area = User Guide

Functional Area = Tutorials

Functional Area = Readme

You need similar rules for software and hardware defects.

By default, workflow rules are evaluated when an issue is saved. For related choice lists like the ones in this example, you probably want to evaluate the rule whenever the **Type** changes. Otherwise, users would have to set the **Type**, save the issue, and then select the Functional Area.

SET FIELD VALUES

Instead of setting the possible values of a choice list, you can define rules that select choices from other lists. These type of rules are called a dependent values rules, because they make the value of one field depend on the value of another field.

EXAMPLES

This rule that assigns a default owner based on the **Problem Area**.

When:

Owner = <None> AND

Problem Area = SW - Outlook

The values to set are:

Owner = Erick Yanez

This rule assigns a default owner for new issues, but allows the issue to be reassigned. Without <**New Issue**> = **yes**, users could not change the owner of the issue. This rule assigns a default owner based on the Functional Area.

When:

<New Issue> = yes AND

Functional Area = Security

The values to set are:

Owner = Erick Yanez

RELATED TOPICS

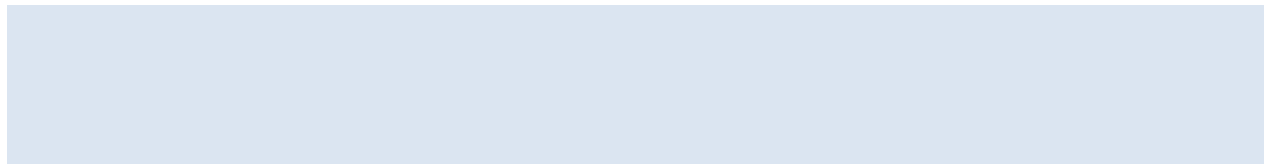
[What You Should Know about Workflow Rules](#)

[What is a Workflow Rule?](#)

[Creating Rule Templates](#)

[Defining Rules](#)

[Changing When Rules Are Evaluated](#)



WHAT YOU SHOULD KNOW ABOUT WORKFLOW RULES

- **Rules work with single-choice list fields**

You cannot define rules that apply to any other type of field, including multiple-choice list fields.

- **Rules are defined per project**

All Web views of a project share the same workflow rules. You cannot disable the workflow rules for specific Web views. The rules are either enabled for all views or disabled for all views.

- **Rules can impact performance**

Defining a large number of rules may impact the performance of Web views.

Using workflow rules to set default values is a good choice if you have a small number of rules and are not limited by resources (server, network, end-user computers).

- **Rules are evaluated when issues are loaded and created**

By default, workflow rules are evaluated when an issue is loaded or created (note that when you save an issue it is immediately re-loaded, triggering the workflow rules). You can force rules to be evaluated when a user changes the value of one of the fields specified in the condition. Evaluating rules on field changes has a potentially higher performance cost. However, if you want to implement dependent fields, you should evaluate the workflow rule on field changes.

- **Changes to choice lists can break workflow rules**

If you remove choices from a choice list, or change the meaning of a choice, rules that use the choice list may stop working or result in unexpected behavior.

- **By default, rules apply to all users**

To build group-specific rules, use the macro **<User in Group>** in your conditions.

- **The order of rules is important**

Rules are evaluated in the order they are listed in the Workflow Editor. For example, in the default workflow, the rule **Admins-<Any>** is the first rule in the list. This ensures that any member of the **Admin** group can make any change to the **Progress** field.

If the **Admins-<Any>** rule is last, then an analyst who is also a member of **Admins** does not have full administrator permissions for changing the **Progress** field. The stricter rules are evaluated first, limiting the changes the analyst can make. If the **Admins-<Any>** rule is last, a developer who is also a member of **Admins** does not have full administrator permissions for changing the **Progress** field. The stricter Developer rules are evaluated first, limiting the changes the developer can make.

RELATED TOPICS

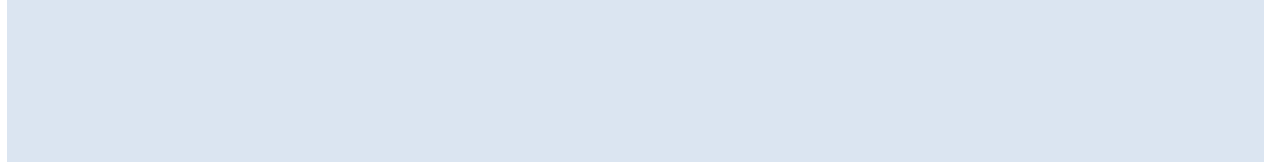
[What Can You Do with Workflow Rules?](#)

[What is a Workflow Rule?](#)

[Creating Rule Templates](#)

[Defining Conditions](#)

Defining Rules



WHAT IS A WORKFLOW RULE?

A workflow rule has this simple format:

```
if ( condition ) then  
    action
```

The **condition** can test:

- If a user belongs to a specific group
- If a specific choice was selected from a choice list.

Multiple conditions can be joined with a logical **AND**. The only test operator available is the equal to operator (=).

If the rule is a possible values rule, the **action** can specify the possible values for one or more choice lists.
If the rule is a dependent values rule, the **action** can select choices from one or more choice lists.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[Creating Rule Templates](#)

[Defining Conditions](#)

[Defining Rules](#)

CREATING RULE TEMPLATES

Before you can define any rules, you must create a template. Templates define the fields used in rules, and rules provide the values. For example, a template might look like this:

When:

```
<User in Group> = _ _ _ _ AND  
Progress = _ _ _ _
```

The possible values are:

```
Progress = _ _ _ _
```

A template defines what the rules look like, while a rule fills in the _ _ _ _ parts with specific values.

To create a rule template:

- 1 Log on to Vizor Web Admin. Click the **Workflow** tab. In the **Project** list, click a project.

Workflow rules based on this template apply to all Web views of this project.

- 2 Click **Template** and click **Add**.

- 3 In the Add New Template dialog:

- In the **Template Name** box, type a name for the template.
- In the **Template Description** box, type a short description of the template.
- Select the type of rules you want to define:

A **Dependent Values** rule makes the value of one field depend on the value of another field.

For example, if How Found = Reported by Customer then Priority = Highest.

A **Possible Values** rule makes the possible values of one field depend on the value of another field.

For example, if Type = Doc Defect then the possible values for Functional Area = Help, Readme, or Manual.

A **Hidden Fields** rule makes the visibility of one or more fields depend on the value of another field.

For example, if Substate = Cannot be Fixed then the Estimated Fix Time field will disappear.

A **Required Fields** rule makes the required state of one or more fields depend on the value of another field.

For example, if *Substate* = *Duplicate* then *Duplicate Issue* will become a required field.

A **Disabled Fields** rule makes the functional availability of one or more fields depend on the value of another field.

For example, if *Substate* = *Cannot be Fixed* then *Fixed by* will be disabled, preventing user input.

A **Hidden Tabs** rule makes the visibility of one or more tabs depend on the value of a field.

For example, if *Substate* = *Cannot be Fixed* then the *Fix* tab will disappear.

4 Click **OK**.

5 In the list of templates and rules, click the new template.

6 Under **Conditions**, click . Then click the **Field** box three times and click a field whose value you want to test. Repeat for each additional field you want to test.

Conditions determine when a rule is applied. If all the field tests evaluate to True, the rule is applied.

Note that the **Field** list contains single-choice fields only.

7 If you created a **Dependent Values** template, click  under **Dependent Values**, then click the **Field** box three times and click the field whose value you want to set. Repeat for each field whose value you want to set.

Note that the **Field** list contains single-choice fields only.

If you created a **Possible Values** template, click  under **Possible Values**, then click the **Field** box three times and click the field whose possible values you want to specify. Repeat for each field whose possible values you want to specify.

Note that the **Field** list contains single-choice fields only.

If you created a **Hidden Fields** template, click  under **Hidden Fields**. Once an empty field row has appeared, no further action is necessary. Multiple fields will be available for selection when [Defining Rules](#) defining rules.

If you created a **Required Fields** template, click  under **Required Fields**. Once an empty field row has appeared, no further action is necessary. Multiple fields will be available for selection when defining rules.

If you created a **Disabled Fields** template, click  under **Disabled Fields**. Once an empty field row has appeared, no further action is necessary. Multiple fields will be available for selection when defining rules.

If you created a **Hidden Tabs** template, click  under **Hidden Tabs**. Once an empty tab row has appeared, no further action is necessary. Multiple tabs will be available for selection when defining rules.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[What is a Workflow Rule?](#)

[Defining Conditions](#)

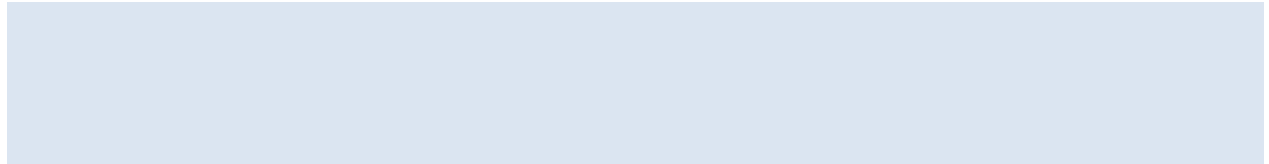
[Defining Rules](#)

[Changing When Rules Are Evaluated](#)

[Changing the Order of Evaluation](#)

[Applying Workflow Rules](#)

[Disabling Workflow Rules](#)



DEFINING CONDITIONS

DEFINING CONDITIONS BASED ON USER GROUPS

By default, a rule applies to all users. To build group-specific rules, use the macro **<User in Group>** in your conditions.

If a user belongs to a group for which there is no rule, then for possible values, the fields displays all possible values.

Rules that apply to the Users group apply to all users, because all users are members of the Users group. Rules for specific groups can override the base rule defined for the Users group.

DEFINING CONDITIONS FOR NEW ISSUES

To define rules that apply only to new issues, use the macro **<New Issue>** in the condition. **<New Issue>** is equal to **Yes** only for issues that have never been saved.

DEFINING RULES

Defining a rule involves filling in the template with specific values.

Note that while you edit rules, you can edit the values, but you cannot add or remove fields. To add or remove fields you must edit the rule template.

To define a rule:

- 1 Log on to Vizor Web Admin. Click the **Workflow** tab.
- 2 In the **Project** list, click a project, then click a rule template.
- 3 Click **Rule** and then click **Add**.
- 4 Under **Conditions**, specify the values to test. For each field, click the **Value** box three times and then click the field value you want to test for.
- 5 If you are defining a **Dependent Values** rule, then under **Dependent Values**, for each field, click the value you want to set. To use the value of another field, double-click the **<Values of Field???** macro.

If you are defining a **Possible Values** rule, then under **Possible Values**, for each field, select the check boxes for each possible value.

If you are defining a **Hidden Fields** rule, then under **Hidden Fields**, for each field, select the check boxes for each field you wish to hide.

If you are defining a **Required Fields** rule, then under **Required Fields**, for each field, select the check boxes for each field you wish to make required.

If you are defining a **Disabled Fields** rule, then under **Disabled Fields**, for each field, select the check boxes for each field you wish to disable.

If you are defining a **Hidden Tabs** rule, then under **Hidden Tabs**, for each field, select the check boxes for each tab you wish to hide.

- 6 Click **Apply** to apply the rule to the Web views of the project.

NOTES

You can also copy a rule and edit it. To copy a rule, click the rule, then click **Rule**, click **Copy**, and type a name for the new rule.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[What is a Workflow Rule?](#)

[Creating Rule Templates](#)

[Deleting Rules](#)

[Renaming Rules](#)

[Using Macros in Rules](#)

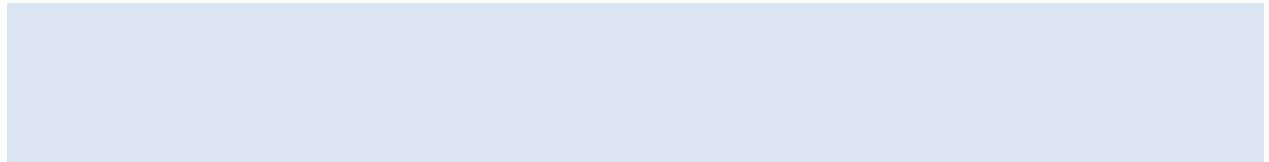
[Setting Possible Values](#)

[Changing When Rules Are Evaluated](#)

[Changing the Order of Evaluation](#)

[Applying Workflow Rules](#)

[Disabling Workflow Rules](#)



DELETING RULES

To delete a rule:

- 1 In the list, click a rule.
- 2 Click **Rule** and then click **Remove**.
- 3 Click **Apply** to apply the changes to the Web views of the project.

RELATED TOPICS

[Defining Rules](#)

[Renaming Rules](#)

[Disabling Workflow Rules](#)

RENAMING RULES

To rename a rule:

- 1 In the list, click a rule.
- 2 Click **Rule** and then click **Rename**.
- 3 Type the new name for the rule.

To change the default names for new rules:

- 1 In the list, click a rule template. Click **Template** and then click **Properties**.
- 2 In the **Name** rules according to box, type the new default name for rules.

By default, rule names are based on the condition values (the values entered in the **Value** column). The string **%Value0%** represents the value in the first row of the condition, **%Value1%** represents the value in the second row, and so on.

If you use a text string (such as *WorkflowRule*) as the default rule name, rule names are formed by appending a number to the string.

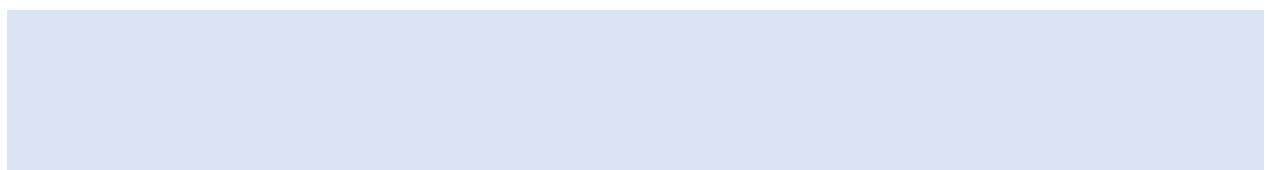
RELATED TOPICS

[Defining Rules](#)

[Deleting Rules](#)

[Applying Workflow Rules](#)

[Disabling Workflow Rules](#)



USING MACROS IN RULES

<Any>

Use the <Any> macro in conditions to define a rule that applies to any value of a field, including <Empty>. For example, in the default workflow, the Admins-<Any> rule allows members of the **Admins** group to set any value in the **Progress** field.

<Empty>

Use the <Empty> macro in conditions to define a rule that applies when a field is empty.

<User>

<User> represents the current user (the user currently logged on to the Web view). You can use <User> in conditions, as a possible value, and as a dependent value.

<New Issue>

<New Issue> is used to create rules that apply only when an issue is created.

For example, to ensure all issues submitted by managers have high priority by default, a rule like this can be created:

Condition	Dependent Values
<New Issue> = Yes	Priority=High
<User in Group> = Managers	

If the <New Issue> condition is not included, the manager would not be able to change the priority because the rule would reset the value to High when the issue is saved.

<Value of Field??>

This macro allows you to define a Dependent Values rule that assigns the value of a field to another field. For example, when an issue is fixed, a workflow rule could set the value of the Fixed By field to <Value of Owner>.

When you add a dependent values rule, you'll see the <Value of Field??> macro in the **Values** column of the **Dependent Values** table. Double-click <Value of Field??> and then select the field whose value you want to use.

By default, the Submitter field is not enabled. To use the field in the <Value of Field ???> macro, you must enable it using the Field Editor in Vizor Admin.

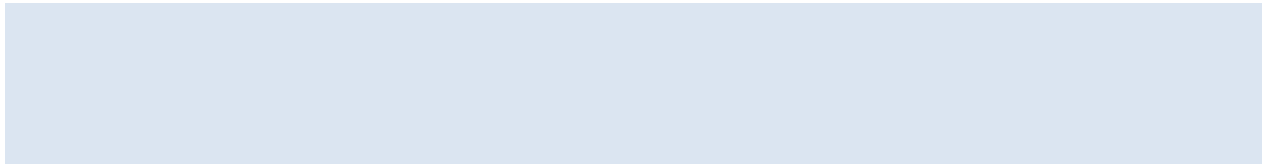
RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[Defining Conditions](#)

[Defining Rules](#)

[Setting Possible Values](#)



SETTING POSSIBLE VALUES

When you set the possible values of a field, you don't have to include the current value of the field in the list. For example, given this rule:

When:

<User in Group> = Employees AND
Progress = Waiting

The possible values are:

Progress = Dropped
Progress = In Progress
Progress = Resolved

When **Progress** is set to **Waiting**, the **Progress** list contains **Dropped**, **In Progress**, **Resolved**, and **Waiting**. When:

<User in Group> = Developers AND
Substate = Assigned

The possible values are:

Substate = To Be Verified

When **Substate** is set to **Assigned**, the **Substate** list contains **Assigned** and **To Be Verified**.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[Defining Conditions](#)

[Defining Rules](#)

[Using Macros in Rules](#)

[Setting Possible Values](#)

CHANGING WHEN RULES ARE EVALUATED

By default, workflow rules are evaluated when an issue is loaded and when issues are created.

However, you can force rules to be evaluated when a user changes the value of one of the fields specified in the condition. Keep in mind that evaluating rules on field changes can have a potentially higher performance cost.

To evaluate rules on field changes:

- 1 In the list, click a template.
- 2 Click **Template** and then click **Properties**.
- 3 Select the **Evaluate rules when users change field values** check box.
- 4 In the Template Properties dialog, click **OK**.
- 5 Click **Apply** to apply the change to the Web views of the project.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[Defining Rules](#)

[Changing the Order of Evaluation](#)

[Applying Workflow Rules](#)

CHANGING THE ORDER OF EVALUATION

Rules are evaluated in the order they are listed in the Workflow Editor. You cannot change the order of rules in a template (except by creating them in the desired order), but you can change the order of the rule templates.

To change the order of evaluation of rule templates:

- 1 In the **Templates** list, click a template.
- 2 Click  or  to move the template up or down in the list.
- 3 Click **Apply** to apply the change to the Web views of the project.

RELATED TOPICS

[What Can You Do with Workflow Rules?](#)

[What You Should Know about Workflow Rules](#)

[Creating Rule Templates](#)

[Defining Rules](#)

[Changing When Rules Are Evaluated](#)

[Applying Workflow Rules](#)

APPLYING WORKFLOW RULES

When you edit a rule, define a new rule, define a new group of rules, or change the group options, you must apply the changes. **Apply** automatically enables the group of rules. If you enable the rules, but don't apply them, nothing happens.

To apply workflow rules:

- 1 In the list, click a template.
- 2 Click **Apply**.

NOTES

Web view users must log out and then log back on before the changes take effect.

RELATED TOPICS

[Defining Rules](#)

[Deleting Rules](#)

[Renaming Rules](#)

[Changing when Rules Are Evaluated](#)

[Changing the Order of Evaluation](#)

[Disabling Workflow Rules](#)

DISABLING WORKFLOW RULES

You can disable all the rules based on a given template.

To disable workflow rules:

- 1 In the list, click a template.
- 2 Click **Template** and then click **Properties**.
- 3 Clear the **Enable in all Web views of the project** check box, and click **OK** in the message box that appears.
- 4 In the Template Properties dialog, click **OK**.
- 5 Click **Apply** to apply the change to the Web views of the project.

RELATED TOPICS

[Defining Rules](#)

[Deleting Rules](#)

[Applying Workflow Rules](#)

CHAPTER 19 - DEFINING SERVICE LEVEL AGREEMENTS

ABOUT SERVICE LEVEL AGREEMENTS

A [service agreement](#) is an agreement between the help desk and users. It defines a required level of service for the users.

A service agreements can be with individual users (contacts), with all users in a department, or all users in a company.

A service level is a guarantee of a certain level of service. For example, a basic service level may provide 9-to-5 service during the regular working week, while a more comprehensive service level might provide 24 hour service, 7 days a week.

Each service level has its own hours of service, target response and closure times for issues, and escalation rules.

A Web view used by help desk analysts has an **SLA** tab. This tab contains information about the service level agreement that applies to the issue.

For example, when employee submits new issue, the **Escalation State** is automatically set to On, and the **Problem Area** is set based on the Service Type selected by the employee. Based on the service type, priority, and the employee's contact information (name, department, and company), Vizor sets the **Target Initial Response Time**, **Target Closing Time** and **Associated SLA** fields.

SETTING UP SERVICE AGREEMENTS

- 1** Define [companies](#) and [departments](#), so you can create service level agreements with groups of users. When you create an agreement, it can be either with a specific user or with all users in a department or in a company.
- 2** Define the [escalation levels](#).
- 3** Define the [service types](#).
- 4** Define [service levels](#). Each service level is defined by the following:
 - [Target times](#)
 - [Escalation Rules](#)
 - [Operational Hours](#)
- 5** Define [service agreements](#).
- 6** Optionally, define [workflow rules](#). Vizor includes default workflow rules for the Service Type and Escalation State fields in Web views.

TARGET TIMES

A service level defines target response and closure (resolution) times for each possible priority value.

Target Initial Response Time The time it takes to initially respond to a new issue. When a help desk analyst responds to an issue, the analyst has to enter the response date and time on the SLA tab of the Web view.

Target Closure Time The time it takes to resolve the issue, either by fixing the problem or providing an acceptable workaround. When an issue is resolved, Vizor automatically enters the date and time into the issue.

When you create a service level, the level gets the default target times.

To edit the target times:

- 1 In the **Service Level** list, click a service level.
- 2 On the **Target Times** tab, click a row (for example, to edit the target times for the Highest priority, click that row).
- 3 Click **Edit**.
- 4 In the Edit Target Times dialog, type the target times in the **Initial Response Time** and **Closure Time** boxes.

The target times must be integer values, such as 2, 7, or 10. You cannot enter decimal values, such as 2.5.

You can change the time units between minutes, hours, and days. For example, to change from hours to minutes, click **hour(s)** and then click **minute(s)** in the pop-up list.

ESCALATION LEVELS

Escalation levels allow the help desk to better identify, query, report, and monitor priorities and urgent issues. By default, Vizor defines five escalation levels. **Level 1** is the lowest level; **Level 5** is the highest.

Time-based escalation rules are used to define a sequence of escalation levels. For example, when a new issue is first submitted to the help desk, the issue is at **Level 1**. If the help desk does not respond to the issue within the target initial response time, the issue is escalated to **Level 2**. As more time goes by without either a response or a resolution, the escalation level of the issue is raised, until it reaches the highest escalation level.

Escalation rules and SLAs are enforced only for issues where the **Escalation State** is set to **On**. **Escalation State** is a field on the **SLA** tab of a Web view. By default, escalation is on, but help desk analysts can pause or disable escalation.

To edit the escalation levels:

- 1 In Vizor Web Admin, click the **Choices** tab.
- 2 In the **Project** list, click a project.
- 3 In the **Choice List** list, click **Escalation Level**.
- 4 Edit the escalation levels:
 - To edit an escalation level, click the escalation level and then click .
 - Click  to create a new escalation level.
 - To delete an escalation level, click the escalation level and then click .
 - To reorder the list, click an escalation level and then use  and  to move it.

ABOUT ESCALATION RULES

Escalation rules change the escalation level of an issue. Each service level has its own escalation rules.

An escalation rule consists of conditions and actions. If all the conditions are true, then all the actions are executed.

To define an escalation rule:

- 1 In Vizor Web Admin, click the **SLA** tab.
- 2 In the **Project** list, click a project.
- 3 Click **Service Levels**.
- 4 Click the **Escalation Rules** tab.
- 5 Click **Add**, or click an existing rule and then click **Copy**.
- 6 In the **Escalation Rule Name** box, type a name for the rule.
- 7 Define the [condition](#) that triggers the rule.
- 8 Define an [action](#) that changes the escalation level when the condition is met.
- 9 Click the check box to enable the condition.

ORDER OF EVALUATION

Escalation rules are evaluated in the order they are listed on the **Evaluation Rules** tab. The first (top) rule is evaluated first, then the next rule below it, and so on.

You can change the order of evaluation by using the Up and Down buttons.

CONDITIONS

You can define multiple conditions. When you have multiple conditions, all conditions must be satisfied before the corresponding actions are executed.

Note that you must select the check box adjacent to a condition; otherwise the condition is disabled.

TARGET TIME CONDITIONS

You can define conditions based on the target response and closure times. For example, an escalation rule can escalate an issue if the issue has not been assigned to a help desk analyst within the guaranteed response time for that service level.

Alternatively, an escalation rule can send an e-mail reminder when an issue has not been resolved within the guaranteed response time.

<FIELD> = <VALUE>

For convenience, the Add Escalation Rule editor allows you to easily add conditions that test the values of up to five choice list fields.

MEETS THE QUERY <VALUE>

For more complex conditions, you can use queries built with the [Query Editor](#) in Vizor Admin. Queries can use any number and type of field to determine whether to escalate an issue.

ACTIONS

Escalation rules can perform the following actions:

- Change the issue priority, owner, assigned work team, or escalation level.
- [Send](#) an e-mail message.
- [Run](#) a program.

SENDING E-MAIL MESSAGES

To send an e-mail message

- 1 Under **Actions**, click **Send an e-mail**.
- 2 In the E-mail Content editor, complete the **To**, **Subject**, and **Body** of the message. You can also attach files.

EDITING THE TO, CC, AND FROM FIELDS

You can use the macros or you can specify users.

To insert a macro:

- 1 Click in the **To**, **CC**, or **From** box.
- 2 Under **Available Macros**, click the macro you want to insert.

The **Contact** is the person who reported the issue, while **Submitter** is the person who entered the issue into Vizor. For example, if a help desk analyst takes a support call, the contact is the person who called, and the analyst is the submitter.

Previous Owner is useful when you send an e-mail message because an issue is reassigned.

To add users to the To, CC, and From lists:

- 1 Beside the **To**, **CC**, or **From** box, click .
- 2 In the Select Users and Contacts dialog box, double-click the users and contacts you want to add.

The list of selected users and contacts is displayed under **Selected Users and Contacts**. You can edit this string of text, for example, to remove a user, or to add the e-mail address of a person who is not a user or contact.

NOTES

- The text box beside the **Find** button allows you to display all users and contacts whose names start with the same letters. In the text box, type the first few letters of the name and then click **Find**.
To display the complete list of users and contacts again, delete all text from the box and then click **Find**.
- Click **Advanced** to search for all users and contacts whose names contain the same string, or to search

based on e-mail addresses.

EDITING E-MAIL CONTENT

You can use macros to insert the issue number, the name of the project, and the values of different fields (such as the Summary and Description Log).

To insert the value of a field:

- 1 In either the **Subject** box or the **Body** box, click where you want to insert the field value.
- 2 Under **Available Macros**, click **Field Value**.
- 3 In the **Field Value** dialog, click the field whose value you want to insert.

If you want to insert the issue description, insert the **Description Log** field, not the **Description** field.

To switch to HTML format:

- Select the **HTML in Body** check box.

ATTACHING FILES

To attach files to an e-mail message:

- 1 Beside the **Attachments** box, click .
- 2 In the **E-mail Content Attachments** dialog, click **Browse** and locate the file you want to attach. Click **Link**.

This adds the file to the list of files linked to this e-mail message. Every time the e-mail message is sent, the linked files are attached to the message.

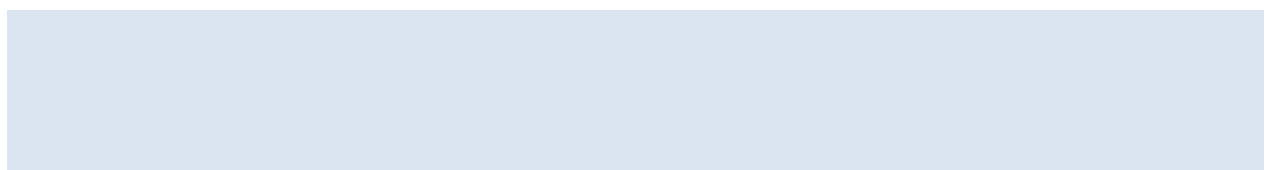
RUNNING PROGRAMS

To run a program, type the location and name of the program in the **Run Program** box. You can include command-line arguments for the program.

The default working directory for programs is the Windows system folder (for example, C:\Windows\system32).

NOTES

Programs must not have a user interface or require any user input. The programs are run silently by the **Mq.Evaluator** service. By default, the programs are run under the credentials of the same user who runs the **Mq.Evaluator** service (CensusUser by default). You can check this in the Services dialog (**Administrative Tools** > **Services**, open the **Properties** for the service, and then go to the **Log On** tab).



SPECIFYING OPERATIONAL HOURS

The operational hours define when the service level agreement is in effect. The service level clock stops running outside of those hours.

For example, if the hours of operation are from 9 to 5, Monday to Friday, then a new issue submitted at 4:59 p.m. on Friday is not going to trigger any escalation rules during the weekend. If the service level specifies an initial response time of 30 minutes, then the help desk has until about 9:29 a.m. on Monday before any escalation rules are triggered.

Note also that help desk analysts can stop the clock for specific issues (by setting the Escalation State to Paused). For example, the analyst may “hit pause” while waiting for more information from the end user.

To set the operational hours

- 1 In Vizor Web Admin, click the **SLA** tab.
- 2 In the **Project** list, click a project.
- 3 Click **Service Levels**.
- 4 Click the **Operational Hours** tab.
- 5 In the **Start Time** and **End Time** boxes, enter the working hours during which the service level is in effect. Type the times in the format 9:00 AM or 5pm.
- 6 Enter the number of working hours in a day.
- 7 Add holidays and other exceptions to the standard working hours.
 - aClick **Add**.
 - bClick the day you want to add.
 - cClick **OK**.

DEFINING SERVICE LEVELS

To define a service level:

- 1** In Vizor Web Admin, click the **SLA** tab.
- 2** Click **Service Levels**.
- 3** In the **Project** list, click a project.
- 4** Beside the **Service Level** box, click .
- 5** In the Add Service Level dialog, type a name and a description for the service level.

CREATING SERVICE AGREEMENTS

A service agreement is an agreement between the help desk and users. The service agreement defines a required level of service for the users.

A service agreement can be with individual users (contacts), with all users in a department, with all users in a company or apply to everyone (Global).

GLOBAL SERVICE LEVEL AGREEMENTS

If you would like to define an agreement that will be applied for all issues that don't fall under other agreements for users, departments or companies, select **Global** for who the agreement is with. To override the global agreement for specific scenarios, you can still define agreements for certain users, departments or companies.

To create a service agreement:

- 1 In Vizor Web Admin, click the **SLA** tab.
- 2 In the project list, click a project. (Each project has its own service agreements.)
- 3 Define a service type. (Each agreement covers a specific type of service, such as Network, Software, or Hardware service.)
- 4 In the **Service Agreements** section, click **Add**.
- 5 In the Add Service Agreement dialog, enter the required information:

Agreement Name The name used to identify the agreement.

Agreement Description Allows you add some notes or comments related to the agreement. Note that you can also attach documents to an agreement.

Agreement is with a A service agreement can be with individual users (contacts), with all users in a department, all users in a company or everyone (Global).

Agreement Customer Depending on whether the agreement is with a contact, a department, or a company, this list includes all the possible users, departments, or companies. If the agreement is global, this list is disabled.

Reporting Period Choose whether reports **for this agreement** are weekly, monthly, quarterly, semi-annual, or annual. The reports are available from the **Service Agreements** section of the **SLA** tab (click the **Reports** button).

APPLYING SERVICE AGREEMENTS TO ISSUES

You can apply a service agreement to the issues that are already in the project database. To do this, use a query to specify the issues to which the service agreement is applied.

To apply a service agreement:

- 1 In Vizor Web Admin, click the **SLA** tab.
- 2 In the project list, click a project.
- 3 In the **Service Agreements** section, click a service agreement.
- 4 Click **Apply**.
- 5 In the Apply Service Agreement dialog, click a query. The agreement is applied to the issues returned by that query.

DISABLING SERVICE AGREEMENTS

To disable a service agreement:

- 1** In Vizor Web Admin, click the **SLA** tab.
- 2** In the project list, click a project.
- 3** In the **Service Agreements** section, click a service agreement.
- 4** Click **Edit**.
- 5** In the Update Service Agreement dialog, clear the **Service Agreement is enabled** check box and then click **OK**.

SERVICE LEVEL REPORTING

Vizor includes a set of service level reports to help you monitor and measure help desk performance against service level objectives,

To view a service level report:

- 1 In Vizor Web Admin, click the **SLA** tab.
- 2 In the project list, click a project.
- 3 In the **Service Agreements** section, click **Reports**. (If the **Reports** button is disabled, click a service agreement.)
- 4 In the Service Agreement Reports dialog, click the report you want to view, and then click **OK**, or double-click the report.
- 5 In the toolbar of the Crystal Reports viewer, click  to print the issue. Click  to save the report as PDF, Word, or Excel.

SERVICE TYPES

A service type identifies a type of service delivered by the help desk to help desk customers. The Service Type (for example, software or hardware) is set when an issue is submitted.

Vizor includes default workflow rules that base the **Problem Area** on the **Service Type**.

To edit the list of service types:

1 In Vizor Web Admin, click the **SLA** tab.

2 In the **Project** list, click a project.

Each project has its own list of service types.

3 In the **Choice List** list, click **SLA Service Types**.

4 Edit the service types:

- Click  to create a new service type.
- To delete a service type, click the service type and then click .
- To rename a service type, click the service type and then click .
- To reorder the list, click a service type and then use  and  to move the service type.

CHAPTER 20 - E-MAIL NOTIFICATIONS

ABOUT NOTIFICATIONS

VIZOR can generate notifications as email messages in order to inform team members, customers and any stakeholder about important events. Notification messages are sent through the native e-mail system.

Notifications keep team members up to date with changes to reported issues. They don't have to run queries, inspect issues, and review revision histories to find out about changes. Instead, they get e-mail messages whenever new issues are assigned to them, or whenever someone edits a issue already assigned to them.

You should try to keep the number of notifications to a minimum. Defining too many notifications for field updates can overwhelm users with e-mail, resulting in users missing important notifications. Use notifications for changes to fields like **Owner**, **Progress**, and **Priority**.

RELATED TOPICS

[What Can You Do With Notifications?](#)

[Editing Notifications](#)

[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

[Setting Up Notifications](#)

[Handling Notification Failures](#)

WHAT CAN YOU DO WITH NOTIFICATIONS?

You can send notifications to specific users or contacts, or to the current owner, the previous owner, or the contact (the person who reported the issue).

To trigger notifications, use simple *when conditions* that test for changes to an issue:

- Sending a notification when an issue is reassigned

When an issue is reassigned, you can use the **<Owner>** and **<Previous Owner>** to send notifications to both the new and the previous owner.

```
Send Notification To:
  <Owner>;<Previous Owner>
```

```
When:
  Owner Update
```

The when condition in this example tests for updates to the **Owner** field.

- Sending a notification when a field is updated

You can use the **Update** test to send a notification when a field is updated.

```
When:
  Priority Update AND
  Revision Number > 1
```

In this when condition, the **Update** test checks for changes to the **Priority** field.

Testing the **Revision Number** allows you to distinguish between new issues and existing issues that have had their priority changed. Without the **Revision Number > 1** test, a Priority Updated notification would be generated each time someone sets the **Priority** when they submit a new issue.

- Sending a notification when a field is set to a specific value

You can send a notification when the priority of an issue is changed to ASAPHighest. To do this, use a when condition that looks like this:

```
When:
  Priority Update AND
  Priority = ASAPHighest
```

This when condition sends a notification when **Priority** is changed. Without the test, a notification would be sent every time an issue with the specified **Priority** is saved.

- Sending a notification when a new issue is submitted

Each issue has a revision number that increases by one each time it is saved. When an issue is first submitted, this revision number is set to 1, so you can test the revision number to find new issues. For example:

```
When:
  Revision Number = 1 AND
  Submitted Date = <Today> AND
  Priority = ASAP OR Priority = TodayPriority = Highest OR Priority = High
```

This when condition looks for new issues submitted today with the highest priorities.

- Sending a notification when certain users update an issue

If employees mark the issues they reported as **Dropped** or **Resolved**, you can notify the owner of the issue using this test:

```
Send Notification To:  
  <Owner>
```

```
When:  
  <User in Group> = Employees AND  
  Progress Update
```

The **<User in Group>** macro allows you to test for user group membership. When a manager changes the priority of an issue, you can notify the owner of the issue.

```
Send Notification To:  
  <Owner>
```

```
When:  
  <User in Group> = Employees AND  
  Progress Update
```

- **Sending a notification when an issue is resolved**

You can use the **<Contact>** macro to notify the submitter when an issues is resolved, or dropped. For example:

```
Send Notification To:  
  <Contact>
```

```
When:  
  Progress Update AND  
  ( Progress = Resolved OR  
    Progress = Dropped )
```

To send an e-mail with a different subject for each possible case (for example, **Issue resolved**, and **Issue dropped**), define a separate notification for each **Progress** value.

RELATED TOPICS

[About Notifications](#)

[Editing Notifications](#)

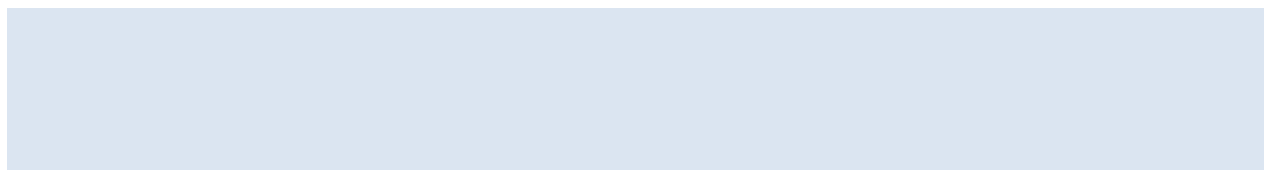
[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

[Setting Up Notifications](#)



EDITING NOTIFICATIONS

Vizor Admin provides a Notification editor for editing notifications. To open the editor, click **Notification Editor** on the **Project** menu. With the Notification editor, you can:

- Specify who receives the notification. Select the names of one or more users from the **Send Notification to** list.
- Define when you want the system to send the notification. Click the **When** list  button to define the condition.
- Specify what you want to include in the notification message. The body of a notification message can include information from the **issue** or the **revision history**.
- Specify the reason for the notification. Type a one-line description in the **Notification Description** box. You can include this description in the body of the e-mail message.

RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

[Setting Up Notifications](#)

SPECIFYING NOTIFICATION RECIPIENTS

While you can send notifications to specific Vizor users and contacts, it is more useful to send notifications to the users listed in the **Owner** and **Contact** fields of an issue. You can do this using macros. For example, to send a notification to the owner of an issue, click the **<Owner>** macro in the **Send Notification To** list.

<Owner> The user whose name is currently entered in the Owner field.

<Previous Owner> The user who owned the issue before it was reassigned.

<Dev Owner> In projects based on XSupportBugTrk, **<Dev Owner>** is the developer assigned to an issue, and **<Owner>** and **<Previous Owner>** are the current and previous support owners.

<Contact> The person whose name is currently entered in the **Contact** field.

RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

[Editing Notification](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

[Setting Up Notifications](#)

[Handling Notification Failures](#)

DEFINING WHEN CONDITIONS

When conditions determine when Vizor sends notifications. A when condition is a test. If an issue satisfies the test, Vizor sends a notification for that issue. You typically use when conditions to:

- Test for updates of a given field. For example, a when condition can generate a notification when issues are reassigned (by testing if the **Owner** field has been updated).
- Test for updates by a specific user or by users in a specific group. For example, a when condition can generate a notification when anybody except a manager updates the **Priority**.

To edit the when conditions for a notification, use the When editor. Like the Query editor, the When editor allows you to combine multiple conditions with parentheses and the **And** and **Or** operators.

To open the When Editor:

Click  beside the **When** field in the Notification editor.

SENDING NOTIFICATIONS OF FIELD UPDATES

To send a notification when someone edits a field, use the **Update** test.

To send Update notifications for a field, select the [Maintain Revision History](#) check box for that field (in the Field Editor).

BASING NOTIFICATIONS ON USERS

The **Field Name** list includes the macros **<User>** and **<User in Group>**. Use these macros together with **=** and **<>** to test who updated an issue. For example, the following condition sends a notification when any employee updates the **Progress** of an issue.

When Editor

When
Name: Progress Updated by Employee

When Condition:

	Field Name	Test	Value	Operator
▶	Progress	Update		And
*	<User in Group>	=	Employees	

View When Condition:
Progress Update And
<User in Group> = Employees

RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

[Editing Notification](#)

[Specifying Notification Recipients](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

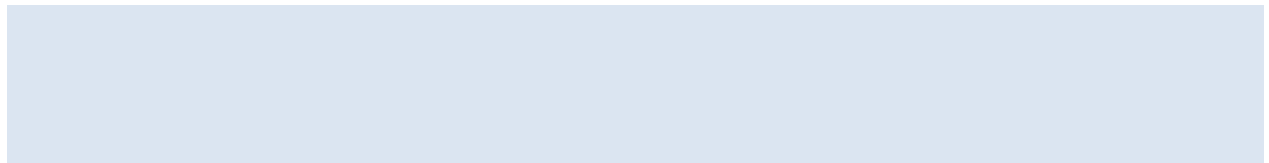
[Setting Up Notifications](#)

[Customizing the Format of Attachments](#)

[Attaching Custom Information](#)

[Adding Custom Mail Contents](#)

[Adding New Notification Reports](#)



SPECIFYING NOTIFICATION CONTENTS

A notification message can include information about the event and issues that generated the notification. For example, if Vizor generates a notification when a new high priority issue is submitted, the message can include a brief description, such as **New high priority issue**.

Notifications generated because of changes to a field can include entries from the Revision History that show the changes. The notifications can also include either a summary or a complete copy of the issue.

To specify the contents, you select one or more of the following from the Include list:

- **Notification Description** is the text entered in the **Notification Description** box of the Notification editor.
- **Detailed** is a copy of the issue.
- **Revision Record** is the entry from the **Revision History** for the change that generated the notification.
- **Summary** is a brief summary of the issue.

The **Notification Description**, **Revision Record**, and **Summary** are included as text in the message body. **Detailed** is included as a separate attachment.

To change the information included in the **Summary** section, edit the **Notification - Summary** report. For the **Detailed** attachment, edit the **Notification - Detailed Record** report.

By default, the **Detailed** attachment is an HTML file. If you use Crystal Reports for listing reports, the **Detailed** attachment is an RTF file. See [Viewing Reports](#) for more information on using Crystal Reports for listing reports.

The subject of an e-mail notification message is automatically generated, and has this format:

```
Notification [<project>]: Record <issue number>  
<notification name>
```

For example:

```
Notification [Vizor]: Record 11 Progress Updated
```

RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

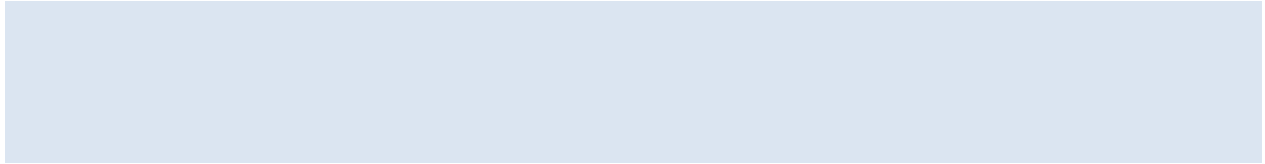
[Editing Notification](#)

[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Managing Workflow with Notifications](#)

[Setting Up Notifications](#)



MANAGING WORKFLOW WITH NOTIFICATIONS

The **Progress** (or status) of an issue indicates where the issue is in your issue tracking system. As the status of an issue changes from **New** to **Assigned** to **In Progress** to **To be Verified** to **Resolved**, different team members become responsible for the issue.

For example, when the issue is assigned to a help desk analyst, you can use update notifications to help manage your help desk workflow. To do this, define a notification that is sent to <Owner> when the issue is assigned and the **Progress** field set to **Assigned**.

When the help desk analyst marks the issue an **In Progress**, you can send a notification to the employee who submitted the issue. Similarly, when an Employee updates the issue (for example, by adding more details to the description), you can have Vizor notify the help desk analyst of the change. Each time the current owner changes the **Substate** and reassigns the issue, Vizor sends a notification to the new owner. For example, suppose a developer fixes an issue:

14. The developer sets the **Substate** to **Fix To Be Reviewed** and reassigns the issue to the person responsible for reviewing the fix (the code reviewer).
15. The code reviewer receives a Substate update notification telling them that they need to review the fix.
16. After the code review, the reviewer then sets **Substate** to **Reviewed** and reassigns the issue back to developer responsible for the fix. The reviewer can also add notes and comments in the **Fix Information** field of the issue record.
17. The developer receives a Substate update notification telling them that the fix is reviewed and accepted.
18. The developer then checks in the fix, sets **Substate** to **To Be Verified**, and reassigns the issue to a tester.

THE TESTER RECEIVES THE SUBSTATE UPDATE NOTIFICATION AND REVIEWS THE

FIX. AFTER THE FIX IS VERIFIED, THE TESTER SETS THE **SUBSTATE TO FIXED** AND

THE ISSUE IS CLOSED. RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

[Editing Notification](#)

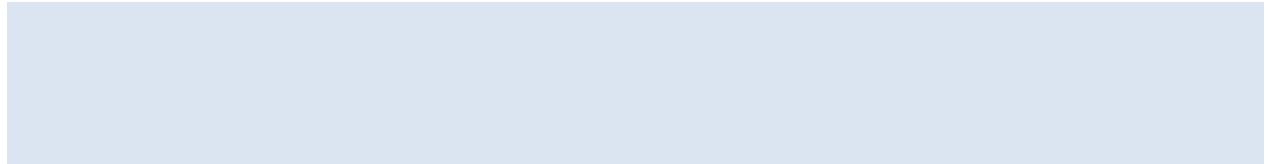
[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Setting Up Notifications](#)

[Handling Notification Failures](#)



SETTING UP NOTIFICATIONS

To set up notifications for Vizor Web, use:

- Vizor Web Admin, a Web-based administration utility.
- Vizor Admin, a Windows-based administration tool.

To set up notifications:

- 1 Set up the notification service.
- 2 In Vizor Admin or in Vizor Web Admin, set the Mail Server options. These options configure the mail server used to send notifications, and specify how often the notification services checks for new notifications to send. For more information, refer to [Mail Settings](#).
- 3 In Vizor Admin, define the notifications you want to send.

SETTING UP THE NOTIFICATION SERVICE

Vizor installs a Microsoft Windows service named **Mq Issue Agent**. This service runs on the server where you installed Vizor Web, and takes care of generating and sending e-mail notifications.

In the Windows Services, you can specify whether you want the service to start automatically whenever the server boots or you want to start the service manually.

TO SET UP THE NOTIFICATION SERVICE ON WINDOWS 2003 AND WINDOWS XP:

- 1 Open the Windows **Control Panel**. Double-click **Administrative Tools**, then double-click **Services**.
- 2 In the right pane, right-click the **Mq Issue Agent** service, and then click **Properties**.
- 3 On the **Log On** tab, select **This account** to choose a user account. Type the password for the account in the **Password** and **Confirm password** fields.

The account must be a valid domain user account and have permissions through the network.

- 4 Click **Apply**.
- 5 On the **General** tab, stop and start the service.

TO SET UP THE NOTIFICATION SERVICE ON WINDOWS 2000:

- 1 Open the Windows **Control Panel**, click **Performance and Maintenance**, then click **Administrative Tools** to open Services.
- 2 In the right pane, right-click the **Mq Issue Agent** service, and then click **Properties**.
- 3 On the **Log On** tab, select **This account** to choose a user account. Type the password for the account in the **Password** and **Confirm password** fields.

The user account must be a valid domain user account and have permissions through the network.

- 4 Click **Apply**.
- 5 On the **General** tab, stop and start the service.

TO SET UP THE NOTIFICATION SERVICE ON WINDOWS:

- 1 Click **Start, Settings, Services** to open Services.
- 2 Stop the **Mq Issue Agent** service.
- 3 Click **Startup**.
- 4 In the Service dialog, select **Startup Automatic** if you want to start the service automatically every time the computer is restarted.
- 5 Select **This Account** and enter a valid domain user account and password that has permissions through the network.
- 6 Click **OK**.
- 7 Click **Start** to start the service.

SETTING THE MAIL SERVER OPTIONS

You can set the outgoing e-mail server options in Vizor Admin or in Vizor Web Admin.

TO SET THE MAIL SERVER OPTIONS IN VIZOR ADMIN:

- 1 Click **Options** on the **Tools** menu.
- 2 Set the server mail options. These options are used when sending notifications.
- 3 If you use an SMTP mail system, click **Test** to check your settings.

Option	Description
Server Mail Sender	Name of sender for SMTP e-mail. Some SMTP mail servers require the sender name to be a valid e-mail address. Note that this name does not appear in the From field.
SMTP Server	Name or IP address of your SMTP mail server. You can test your settings by clicking Test .
Server SMTP Port	The port for the SMTP server.
Server Mail Interval	Specifies how often (in minutes) Vizor generates notifications.
Send Invalid Notifications To	If the Maximum Notification Retries is surpassed, the notification will be sent to this address. If there is no address specified, the notification service will continue trying to send to the original recipient.
Maximum Notification Retries	The maximum number of retries for e-mail notifications. Once surpassed, the notification will be sent to the address specified in the Send Invalid Notifications To option.

To set the mail server options in Web Admin, refer to [Mail Settings](#).

RELATED TOPICS

[About Notifications](#)

[What Can You Do With Notifications?](#)

[Editing Notification](#)

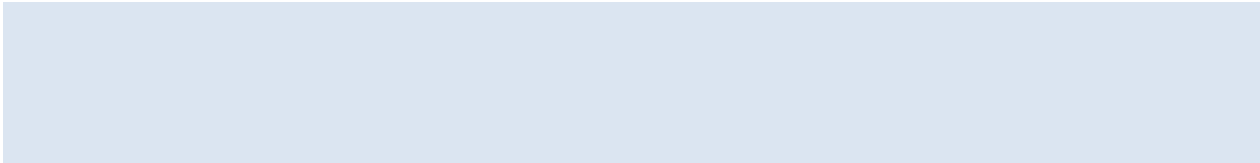
[Specifying Notification Recipients](#)

[Defining When Conditions](#)

[Specifying Notification Contents](#)

[Managing Workflow with Notifications](#)

[Handling Notification Failures](#)



HANDLING NOTIFICATION FAILURES

If Vizor is unable to send a notification, it tries to resend the notification. By default, the maximum number of retries is 100. When the maximum number of retries is reached, the notification is sent to the e-mail address specified by the **Send Invalid Notifications To** option.

To change the **Maximum Notification Retries** option or the **Send Invalid Notifications To** option, click **Tools>Options** in Vizor Admin.

FROM TRAINING

Notifications

You can configure HelpDesk to automatically generate notification messages that inform users of changes to the HelpDesk database. Notifications work by monitoring one or more fields in the database and then sending e-mail messages to the contact, the owner, the previous owner, or a specific user when a change occurs.

Notifications are sent using the native e-mail system. They can be used to keep users up-to-date with changes to issues avoiding the need to run queries, inspect issues, and review revision histories to find out about changes.

*Use notifications for changes to fields like **Owner**, **Progress**, and **Priority**.*

Defining too many notifications can overwhelm users with e-mail, and result in them missing important notifications

Editing Notifications

Use the Notification Editor to define and edit the notifications for your system.

With it, you can:

- ☐ Specify who you want to receive the notification. Select the names of one or more users from the **Send Notification to** list.
- ☐ Define when you want HelpDesk to send the notification. Click the button beside the **When** list to define a when condition, which is like a query that retrieves updated issues.
- ☐ Specify what you want to include in the notification message. The body of a notification message can include information from the **Issue Record** or the **Revision History**.

☐ Describe why the notification was sent. Enter a one-line description in the **Notification Description** field. You can include this description in the body of the e-mail message.

Exercise: Sending a notification when a new issue is submitted.

- 1 Click the Windows Start button, and select Programs, Vector, HelpDesk, HelpDesk Admin.
- 2 Log on as admin (password = admin).
- 3 In the Projects list, select HelpDesk.
- 4 Click Project > Notification Editor.
- 5 Click the New Notification button.
- 6 As a notification name, enter New Issue Submitted.

Issue Tracker Page 52

Your notes:

Issue Tracker

- 7 Click OK.
 - 8 Click the Send Notification To dropdown, select <Owner>.
 - 9 Click the [...] button next to the When field.
 - 10 Click the New When button.
 - 11 As the when name, write Issue Submitted.
 - 12 Click OK.
 - 13 Click the Field Name cell in the first When Condition row, and select the value Revision Number.
 - 14 In the Test cell, select =.
 - 15 In the Value cell, type 1.
 - 16 In the Operator cell, select And.
 - 17 Click the Field Name cell in the second When Condition row, and select the value Submitted Date.
 - 18 In the Test cell, select =.
 - 19 In the Value cell, select <Today>.
 - 20 Click Apply.
 - 21 Click OK.
 - 22 Click the When dropdown, and select the newly created Issue Submitted entry.
 - 23 Set the Include field to include Notification Description, Summary and Detailed.
 - 24 In the Description field, write A new issue has been submitted.
 - 25 Click Apply.
- Log in, and create a new issue in the HelpDesk system, and make yourself the owner. An e-mail should be sent to the e-mail address associated with your user account shortly.

Exercise: Sending a notification when a non-owner updates the issue.

- 1 Open the Notification Editor
- 2 Click the New Notification button.
- 3 As a notification name, enter Owner Issue Edited.
- 4 Click OK.
- 5 Click the Send Notification To dropdown, select <Owner>.

Issue Tracker Page 53

Your notes:

Issue Tracker

- 6 Click the [...] button next to the When field.
 - 7 Click the New When button.
 - 8 As the when name, write Non-owner updated the issue.
 - 9 Click OK.
 - 10 Click the Field Name cell in the first When Condition row, and select the value Revision Number.
 - 11 In the Test cell, select >.
 - 12 In the Value cell, type 1.
 - 13 In the Operator cell, select And.
 - 14 Click the Field Name cell in the second When Condition row, and select the value Owner.
 - 15 In the Test cell, select <>.
 - 16 In the Value cell, select <User>.
 - 17 Click Apply.
 - 18 Click OK.
 - 19 Click the When dropdown, and select the newly created Nonowner updated the issue entry.
 - 20 Set the Include field to <All>.
 - 21 In the Description field, write Someone edited your issue – see the included revision history changes.
 - 22 Click Apply.
- Log in, and create a new issue in the HelpDesk system, and make yourself the

owner. Log in with another user, and then modify your issue. An e-mail should be sent to the e-mail address associated with your user account shortly.

Exercise: Sending a notification when VIP issues are updated

For this exercise, we're going to be treating the users "Jean Manager" and "Jean Developer" as VIP users.

- 1 Open the Notification Editor.
- 2 Click the New Notification button.
- 3 As a notification name, enter VIP Update Notification.
- 4 Click OK.
- 5 Click the Send Notification To dropdown, select <Owner>.
- 6 Click the [...] button next to the When field.

Issue Tracker Page 54

Your notes:

Issue Tracker

- 7 Click the New When button.
- 8 As the when name, write VIP user updated highest priority issue.
- 9 Click OK.
- 10 Click the Field Name cell in the first When Condition row, and select the value Revision Number.
- 11 In the Test cell, select >.
- 12 In the Value cell, type 1.
- 13 In the Operator cell, select And.
- 14 Click the Field Name cell in the second When Condition row, and select the value Priority.
- 15 In the Test cell, select =.
- 16 In the Value cell, select Highest.
- 17 In the Operator cell, select And.
- 18 Click the Field Name cell in the third When Condition row, and select the value Submitter.
- 19 In the Test cell, select =.
- 20 In the Value cell, select Jean Manager.
- 21 In the Operator cell, select Or.
- 22 Click the Field Name cell in the fourth When Condition row, and select the value Submitter.
- 23 In the Test cell, select =.
- 24 In the Value cell, select Jean Developer.
- 25 Select the row you created in steps 24-26 and click the Add () button.
- 26 Click Apply.
- 27 Click OK.
- 28 Click the When dropdown, and select the newly created VIP user updated highest priority issue entry.
- 29 Set the Include field is set to <All>.
- 30 In the Description field, write A VIP issue has been updated!
- 31 Click Apply.

Log in as "Jean Manager" or "Jean Developer" (no password), and create a new issue in the HelpDesk system. Log in as "admin", and modify this issue to have Priority of *Highest*, and

change the Owner to *admin*. After saving the issue, further edits should trigger the notification.

CHAPTER 21 - PROVISIONING

ACTIONS

Types of Actions available by default

The framework can be extended to have more native adapters.

APPROVAL

This action triggers a multi-level approval process where multiple approvers can take part. Required approvers are notified of the pending approval within the VIZOR system and via email, and they can approve or reject from the Web UI or directly from the email.

ACTIVE DIRECTORY

- The following actions are built-in, via our native AD adapter.
- Add user to Security Group
- Assign user to OU
- Change user password
- Create user account
- Delete user account
- Disable user account
- Enable user account
- Remove user from Security Group

- Reset user password

HELPDESK TICKET

This action creates a helpdesk ticket, assigns it to a person or team, and monitors its status until completion. At the moment, out-of-the box, only the VIZOR HelpDesk is supported.

EXECUTE COMMAND

This action runs an executable and passes dynamic information from Connect via macros. For example, you can execute a Powershell or a 3rd party tool.

CHANGE REQUEST (CHANGE MANAGEMENT TICKET)

This action creates a Change Request in the VIZOR Change Management system. The change request is started and its progress monitored until completion.

SEND EMAIL

This action simply sends an email to one or multiple recipients based on a configurable template.

VIZOR ACTION

This action creates and monitors a Connect Action, which is essentially a generic task/issue inside Connect and assigned to a Connect user. It is only accessible inside the Connect system and it is useful for following up activities that are not tracked by other systems.

CHAPTER 22 - GENERATING WEB VIEWS

WHAT IS A WEB VIEW?

Web views are different views of a Vizor project. Each Web view of a project can display different issues and different fields.

For example, employees don't need to see all the issues in the database, just the ones they reported. Also, they don't need to see internal notes and other information entered by the support staff. Support staff, on the other hand, need access to all the information in the issue database.

For example, customers on a beta list don't need to see all the issues in a project, just the ones submitted by the beta list. And customers don't need to see any fix-related information added to the issues by the development staff. Internal development staff, on the other hand, needs access to all the issue information in the project.

RELATED TOPICS

[Generating Web Views - Issues to Consider](#)

[Starting the Web View Editor](#)

[Types of Web View](#)

[Setting Web View Attributes](#)

[Setting Group Access Permissions](#)

[Adding the Child Issues Tab](#)

[Exporting Fields](#)

[Exporting Queries](#)

[Editing Field Attributes](#)

[Pointing Users to Web Views](#)

[Generating Web Views](#)

[Copying Web Views](#)

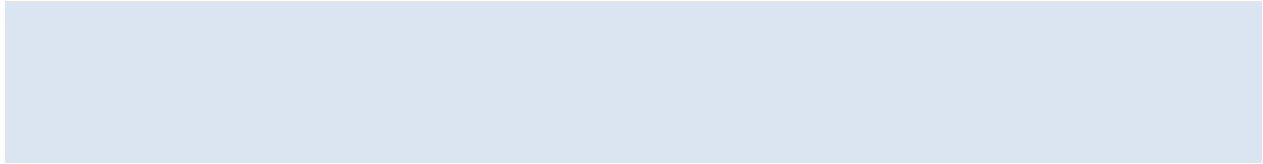
[Deleting Web Views](#)

[Renaming Web Views](#)

[Changing the Date and Time Formats for Web Views](#)

[Customizing Web Views](#)

[Setting Default Values for Fields](#)



ABOUT THE WEB VIEW EDITOR

Use the Web View Editor to create, edit, and generate Web views. To create Web views, you must run the Web View Editor on the Web server.

To start the Web View Editor from Vizor Admin:

- On the **Tools** menu, click **Web**, then click **Web View Editor**.

The **Tools > Web > Web View Editor** command is also available when the Web View editor is installed on the same computer where you are running Vizor Admin.

To start the Web View Editor from the Start menu:

- Click **Start > Programs > Vector > Vizor > Web View Editor**. On the **Start** menu, point to **Programs**, click **Vector** Vizor, and then click **Web View Editor**.

RELATED TOPICS

[Creating Web Views](#)

[Copying Web Views](#)

[Deleting Web Views](#)

[Renaming Web Views](#)

CREATING WEB VIEWS

To create a new Web view, you must run the Web View Editor on the Web server. Web View Editors installed on other computers can only edit and regenerate existing Web views.

You can either create a new, blank Web view or you can copy an existing web view. Copying allows you to reuse all the settings in the existing Web view. For example, when you [copy a Web view](#), all the field attribute settings are copied.

To create a new, blank Web view:

- 1 Run the Web View Editor on the Web server.
- 2 Click **New View** , type the name of the Web view, and click OK.
- 3 In the **Project Name** list, click a project.
- 4 In the **View Type** list, click the type of view you want to generate.
- 5 Set the [group access permissions](#) for the view.
- 6 Set the [Web view attributes](#).
- 7 [Export fields](#) to the Web view.
- 8 [Export queries](#) to the Web view.
- 9 Click  to save your settings and generate the Web view.

NOTES

- To log on to the Web view, click .
- Click  to save your settings without generating the Web view.

RELATED TOPICS

[Generating Web Views - Issues to Consider](#)

[Starting the Web View Editor](#)

[Types of Web View](#)

[Setting Web View Attributes](#)

[Setting Group Access Permissions](#)

[Adding the Child Issues Tab](#)

[Exporting Fields](#)

[Exporting Queries](#)

[Editing Field Attributes](#)

[Pointing Users to Web Views](#)

[Generating Web Views](#)

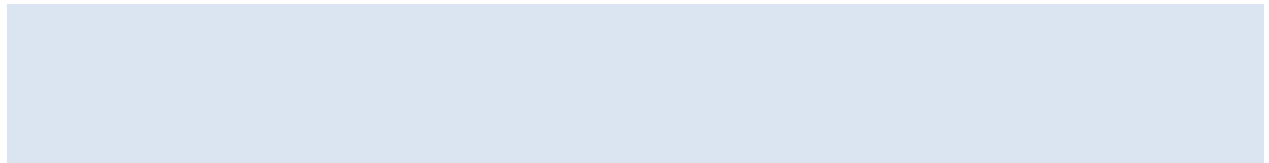
[Deleting Web Views](#)

[Renaming Web Views](#)

[Changing the Date and Time Formats for Web Views](#)

[Customizing Web Views](#)

[Improving Web View Performance](#)



GENERATING WEB VIEWS

To generate a Web view:

- 1 In the **Project / View** list, click a Web view.
- 2 Click  to generate the Web view.
- 3 To log off all users before generating the Web view, click **Yes**. If you click **No**, users logged on to the Web view do not see the changes until they next log on.
- 4 Click **Yes** to overwrite the existing files.
- 5 After the Web view is generated, click  to log on to the Web view and test it.

NOTES

- You can generate Web views without using the Web View Editor. On the Web server computer, click **Start > All Programs > Vector > Vizor > Vizor Tools > Generate Web Views**.
- Vizor has an optional

RELATED TOPICS

[Starting the Web View Editor](#)

[Creating Web Views](#)

[Releasing Databases](#)

COPYING WEB VIEWS

You can create a Web view based on an existing Web view. Note that if you copy a view and then change the project, all view settings are cleared.

To copy a Web view:

- 1 In the **Project / View** list, click a Web view.
- 2 Click **Copy** .
- 3 Type a name for the new Web view.

The Web View Editor creates a new Web view that is an exact copy of the selected Web view.

RELATED TOPICS

[Starting the Web View Editor](#)

[Creating Web Views](#)

[Generating Web Views](#)

[Deleting Web Views](#)

[Renaming Web Views](#)

DELETING WEB VIEWS

To delete a Web view:

- 1 In the **Project / View** list, click a Web view.
- 2 Click **Delete** .

Deleting a view deletes all the generated files and folders (used by that Web view) on your server.

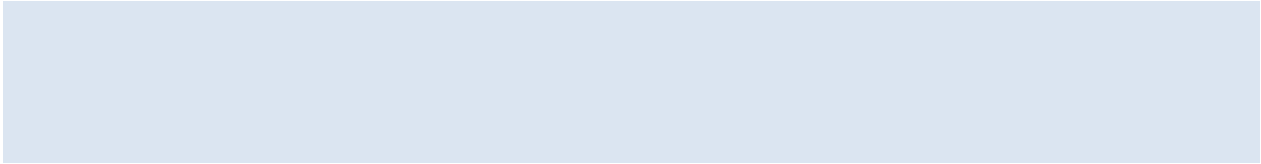
RELATED TOPICS

[Starting the Web View Editor](#)

[Creating Web Views](#)

[Copying Web Views](#)

[Renaming Web Views](#)



RENAMING WEB VIEWS

To rename a Web view:

- 1 In the **Project / View** list, click a Web view.
- 2 Click **Rename** .
- 3 Type a new name for the Web view.

RELATED TOPICS

[Starting the Web View Editor](#)

[Generating Web Views](#)

[Creating Web Views](#)

[Copying Web Views](#)

[Deleting Web Views](#)

TYPES OF WEB VIEWS

Normal Views

Normal views allow users to submit and update issues, to view the history and e-mail conversation of the issue, to view reports and summary lists, and more. This type of Web view has more features available than the other types and every user session (from logon to logoff) takes up one license.

Read-only Views

Read-only views allow users to review existing issues, but not to edit them. All fields are read-only, and users cannot create or submit new issues.

Knowledge Base Views

Knowledge Base views allow users to look through Knowledge Base Articles that have already been published by Vizor analysts. These views are equipped with a quick search feature, as well as the ability to load a specific article by ID.

The administrator can choose whether to create a Knowledge Base view that requires a logon, or not. If the view will not require the user to login, one privileged user account must be associated with the view.

For more information, refer to [Knowledge Base Views](#).

Frequently Asked Questions Views

Frequently Asked Questions views provide a list of articles from the Knowledge Base, specifically, all of those that have Article Type set to *FAQ*. These views feature a quick search feature, and the ability to open multiple FAQs right in the search results.

The administrator can choose whether to create a FAQ view that requires a logon, or not. If the view will not require the user to login, one privileged user account must be associated with the view.

For more information, refer to [FAQ Views](#).

Submit-only Views

Submit-only views are used to submit new issues. A submit-only view is basically just a form and a **Submit** button. Users cannot run queries or edit (or even view) existing issues.

A submit-only view allows an unlimited number of users to submit issues. All you need is a single Vizor user account. Users automatically log on to the view with this account (in fact, users never see the logon window, they go straight to the view).

This user account must belong to a group that has permission to open the Web view, and to add and update contacts. The account is used to set the **Submitter** field.

If you have more than one submit-only view for the same project, you may want to create separate accounts for each view. This allows you to distinguish between issues submitted from different views.

The user submitting the issue is considered the contact. Users must enter their contact information (name, e-mail, and so on) the first time they submit an issue.

SELF-SERVICE VIEWS

Self-Service views are a type of Web view to be used by either internal or external Vizor users to enter issues and be able to view (and possibly edit) the issues they have submitted. You can choose to allow them to edit certain fields in issues that they have submitted (for example, the Description field); or allow them to edit fields that were left empty when they submitted the issue; or simply view their submitted issues in a read-only format.

For more information, refer to [Self-Service Views](#).

HOW LICENSES ARE USED DEPENDING ON THE TYPE OF WEB VIEW

Each user session in a Normal Web view takes up one concurrent or floating license.

Submit-Only views do not take up any license.

Unlike normal Web views, the Knowledge Base, FAQ, Self-Service and Read-Only Web views follow a slightly different licensing model: for each concurrent license that is available, 5 logons will be allowed.

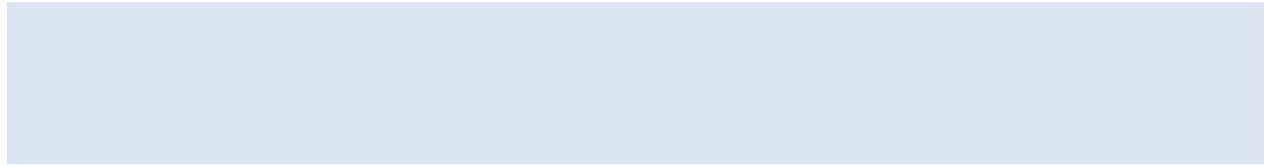
For example, if you have 20 concurrent licenses, and 15 licenses are already being used in Normal Web views, that means there are 5 free concurrent licenses (20 total minus 15 used) available to the Knowledge Base, FAQ, Self-Service or Read-Only views. These 5 free concurrent licenses are shared across all four Web views - which means you could have a scenario where all logons are used up with the Knowledge Base view using 6 logons, the Self-Service view using 8, the Read-Only view 4, and the Submit-Only view 7.

If you have 20 concurrent licenses and none of them are in use, there are 100 logons available to the Knowledge Base, FAQ, Self-Service or Read-Only views.

Reserving Licenses

It is possible to reserve licenses for specific users.

For more information, refer to [Reserving Licenses](#).



SETTING WEB VIEW ATTRIBUTES

To set Web view attributes:

- 1 In the shortcut bar, click  **Web View**.
- 2 Click  (beside **Attributes**) to open the Web View Attributes dialog:

Auto Refresh Summary List

Frequency (in seconds) to automatically refresh the Summary List. Enter 0 to turn off auto refreshing.

By default, the summary list will refresh every 300 seconds (5 minutes).

If you change this attribute, you must regenerate the Web view.

Contact Tab in Record Details

Whether the fields exported on the Contact tab should appear in the record details section (tabs section) or just in the user details pop-up.

If you change this attribute, you must regenerate the Web view.

Description

Description of the Web view. Displayed in the list of Web views that appears when a user logs on. The default description is the Project / View name. To include the project and view names in the description, use the macros <ProjectName> and <ViewName>.

If you change this attribute, you must regenerate the Web view.

Display All Read-Only Fields As Text

Read-only fields can be displayed as text (**Yes**) or as disabled controls (**No**). In a read-only Web view, all fields are read-only. In normal and submit-only Web views, read-only fields have been disabled through the Field editor.

If you change this attribute, you must regenerate the Web view.

Include VSS

Adds the **Version Control** tab to the Web view. This tab allows users to link issues to source files in a Microsoft Visual SourceSafe database, and to perform version control operations such as check out and check in directly from a Web view.

Issues per Page

The number of issues to show per page in the Summary List.

Timeout

The number of minutes of inactivity before the browser session times out and the user is logged off.

Load Issues Not In Query

If **Yes**, users can load any issue by typing the issue number in the **Issue** box of a Web view. If **No**, users can load only the issues shown in the Summary List.

Web View Text

Text that will appear above the fields on the first tab of the web view.

If you change this attribute, you must regenerate the Web view.

Web View Text CSS Class

CSS Class to use for the **Web View Text**. For more information, see [Applying CSS Styles](#).

If you change this attribute, you must regenerate the Web view.

Web View Theme

Which theme to use for the web view. Themes alter visual elements like icons, text, etc.

This is an advanced attribute, and usually does not need to be changed.

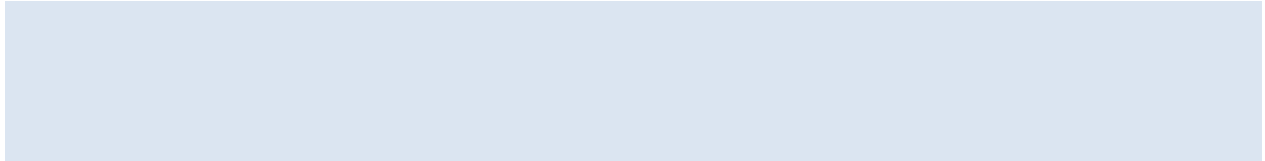
If you change this attribute, you must regenerate the Web view.

RELATED TOPICS

[Creating Web Views](#)

[Types of Web View](#)

[Setting Group Access Permissions](#)



SETTING GROUP ACCESS PERMISSIONS

You can control access to Web views on a group-by-group basis. When users try to log on, they see only the Web views that they have permission to access.

To set group access permissions:

- 1 In the shortcut bar, click **Web View** .
- 2 Click  (beside **Groups Allowed to Open**) to open the View Permissions dialog.
- 3 In the **Groups** box, click the groups you want to be able to access the Web view.

If you allow the **Users** group to open the Web view, then all users can open the Web view.

The **Groups** box lists the groups that are allowed to open the project (as specified in Vizor Admin).

RELATED TOPICS

[Creating Web Views](#)

[Types of Web View](#)

[Setting Web View Attributes](#)

KNOWLEDGE BASE VIEWS

Knowledge Base views, or KB views, are a type of Web views used to openly publish specific information about certain issues. KB views act like Read-Only views only they are accessible by everybody. They can be designed to give customers and/or users such information as:

- Solutions to known problems
- Project-related news

The screenshot shows the Vector Issue Tracker Knowledge Base interface. At the top, the logo "Vector issue tracker" is on the left, and "Search the Knowledge Base (KB)" is in the center. On the right, there are "Switch" and "Log Off" buttons. Below the header is a search bar with the text "Search for" and a "Go" button. A table lists four articles with columns for Article ID, Title, and Status. Article 2 is selected. Below the table, the "Knowledgebase" tab is active, showing the details for Article 2: "How to scroll through the entire detailed section of a web view". The summary states: "This Technical Brief will walk you through the process of scrolling through the entire Detailed Section of the view and still use the tabs to jump into different parts of the Detailed Section of the view." A "Please note:" section follows, with two bullet points: "The folder CensusWeb is assumed to exist in the Census Server install directory." and "The Census Server installation folder is assumed to exist." The default location is given as "C:/Program Files/CensusServer/CensusWeb". At the bottom, there is a status bar with "Article ID: 2", a green checkmark, "Ready.", a printer icon, and "Powered by Vector Issue Tracker".

Vector
issue tracker

Search the Knowledge Base (KB)

Switch Log Off

Search for Go

Article ID	Title	Status
1	How to implement timestamping with just one memo field	--
2	How to scroll through the entire detailed section of a web view	--
3	Error message when running the report editor	--
4	Windows 2003 error message when opening an attachment: "Http Error 404 - File or directory not found"	--

Page 1 of 1 (4 matches)

Knowledgebase

Title:

Summary: This Technical Brief will walk you through the process of scrolling through the entire Detailed Section of the view and still use the tabs to jump into different parts of the Detailed Section of the view.

Please note:

- The folder CensusWeb is assumed to exist in the Census Server install directory.
Default Location: C:/Program Files/CensusServer/CensusWeb
- The Census Server installation folder is assumed to exist.

Article ID: Ready. Print **Powered by Vector Issue Tracker**

USING KNOWLEDGE BASE VIEWS

To use Knowledge Base views:

In your browser, type or paste the KB view URL given to you by your Vizor administrator.

Type in a search word or phrase and click Go.

Find the article (issue) in the summary list.

Click an article in the summary list to open it and view its details.

To integrate a Knowledge Base view to your website:

Open the HTML file corresponding to the webpage in which you want the KB view link to appear.

At the proper location, type `<A href="URL">LINK TEXT</A>`, where URL is the address specified in the Web View Editor, and LINK TEXT is the text you want to appear as the link.

Browse to this webpage in your website and click on the new link to verify that it is functional.

CREATING AND UPDATING A KNOWLEDGE BASE VIEW

To create or update a Knowledge Base view:

Create a user account to be used as the KB view logon account (see [Creating User Accounts](#)).

Note:

The user specified in the Web View Editor must belong to a group who has access to the Ad-hoc Query feature.

Use the Web View Editor to create or update KB views (see [Creating Web Views](#)).

Set the proper username and password for the KB view account by clicking  beside the View Type list.

Export to the view the desired fields to be displayed, including the Add To Knowledge Base field.

To change the list of articles (issues) published to a KB view:

Update the KB Query to filter the issues as desired (see [About Queries](#)).

To change the information (fields) displayed in a KB view's summary list:

Update the KB Layout to include the desired fields (see [About Layouts](#)).

Update the KB Sort to have the summary list ordered the desired way (see [About Sorts](#)).

THE ADD TO KNOWLEDGE BASE FIELD

Knowledge Base views use their Project's KB Query by default. This Query controls which articles (issues) are accessible through the KB view. The Add to Knowledge Base field helps define this Query and allows Vizor analysts to easily specify if an issue is to be accessible as a KB article. By default, the KB Query returns all the issues where this field is set to either Internal or Public.

The Add to Knowledge Base field is found in the Resolution tab of the HelpDesk view by default. You will have to use the Web View Editor to export this field to other Web views if needed. Please note that this field must be exported to KB views when it is used in the KB Query; the Query will fail otherwise.

Typically, you will set this field to "No" to prevent the opened issue from being published in the KB view, and set it to anything else to have it published.

Related Topics

[Types of Web Views](#)

[FAQ Views](#)

SELF-SERVICE VIEWS

Self-Service views are a type of Web view to be used by either internal or external VizOR users to enter issues and be able to view (and possibly edit) the issues they have submitted. You can choose to allow them to edit certain fields in issues that they have submitted (for example, the Description field); or allow them to edit fields that were left empty when they submitted the issue; or simply view their submitted issues in a read-only format.

USING SELF-SERVICE VIEWS

To use Self-Service views:

Log on to the Vizor Web views and select the HelpDesk / Employee view.

You will be able to create new issues and view any issues that you have submitted.

By default, you will be able to edit the Description and Attachments fields of the issues that you have submitted. You will be able to edit the Computer Name field of the issues that you have submitted if there is no value already entered.

CREATING AND UPDATING A SELF-SERVICE VIEW

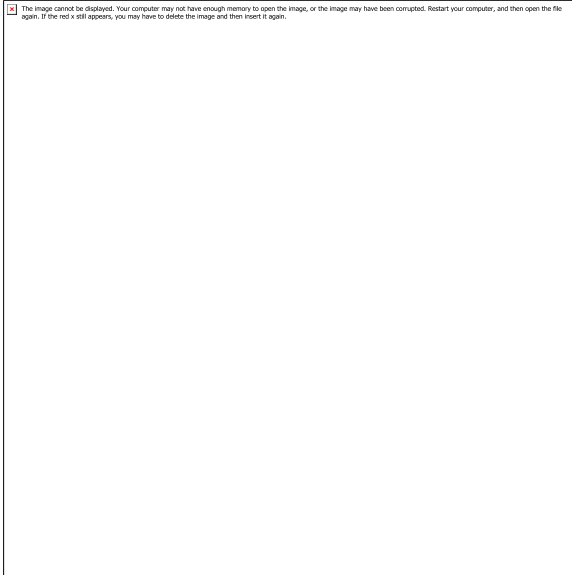
To create or update a Self-Service view:

Use the Web View Editor to create or update Self-Service views. Please refer to the [Generating Web Views](#) section for further details on this procedure.

To ensure the logged in user will only see the issues they submitted:

Only export the My Reported Issues query to the Self-Service view.

To specify which fields the user should be able to edit after the issue has been saved:



Click beside the View Type box to open the Editable Fields dialog.

The fields that are exported to the Web view will be available in the The user will always be able to edit list.

Select the fields that the user should be able to edit after their issue is saved (up to a maximum of 2 fields).

Click OK and generate your Web view.

To specify which fields the user should be able to edit if the field is empty:

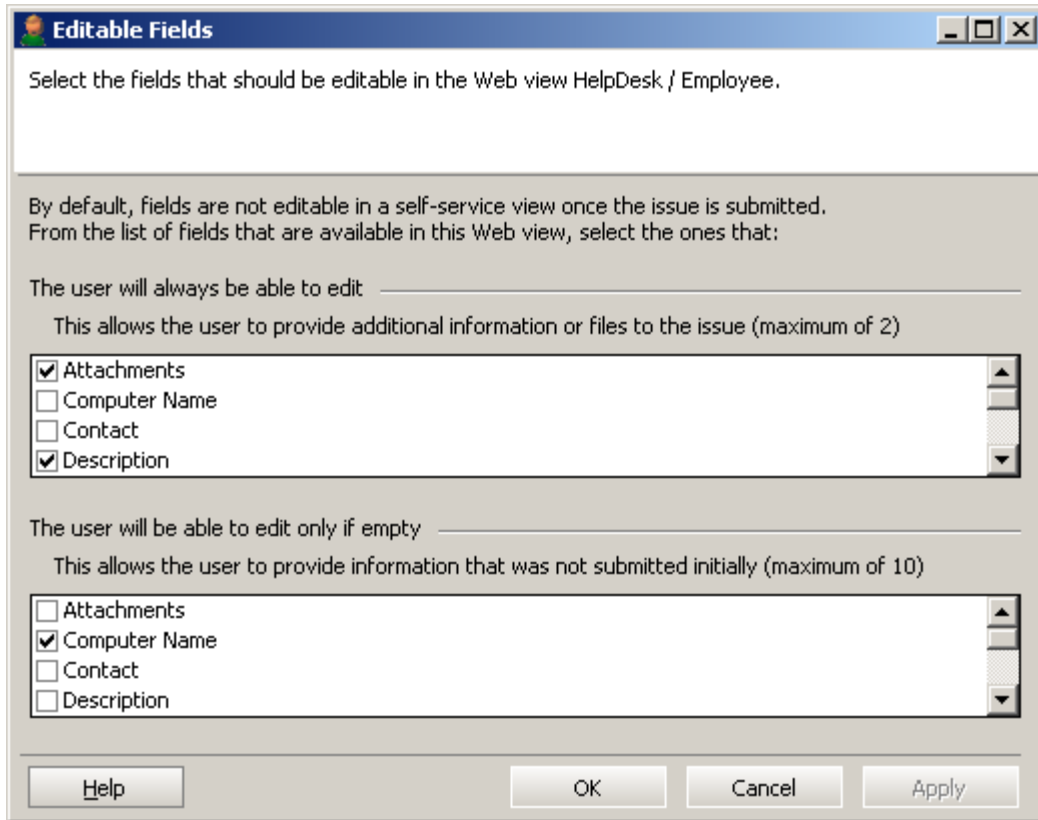


Click beside the View Type box to open the Editable Fields dialog.

The fields that are exported to the Web view will be available in the The user will be able to edit only if empty list.

Select the fields that the user should be able to edit after their issue is saved, if the field is still empty (up to a maximum of 10 fields).

Click OK and generate your Web view.



Editable Fields

Select the fields that should be editable in the Web view HelpDesk / Employee.

By default, fields are not editable in a self-service view once the issue is submitted. From the list of fields that are available in this Web view, select the ones that:

The user will always be able to edit _____

This allows the user to provide additional information or files to the issue (maximum of 2)

- ☒ Attachments
- ☐ Computer Name
- ☐ Contact
- ☒ Description

The user will be able to edit only if empty _____

This allows the user to provide information that was not submitted initially (maximum of 10)

- ☐ Attachments
- ☒ Computer Name
- ☐ Contact
- ☐ Description

Help OK Cancel Apply

EXPORTING FIELDS

To export fields:

- 1 In the shortcut bar, click **Fields** .
- 2 In the **Tab** list, click a tab (a tab is a group of related fields).
- 3 In the **Available** list, click a field and then click . This moves the field to the **Export To View** list, which lists the fields included in the Web view when it is generated.
- 4 Use  and  to change the order of the fields in the **Export to View** list. The order of the fields in the list determines the layout of the fields in the Web view.

NOTES

- Click  to move all fields in the **Available** list to the **Export To View** list.
- Required fields are exported by default. You can remove the fields—just ignore the warning.
- The **Field** list in Ad-hoc Query editor lists exported fields only, plus the **Issue** field
- Layouts show only exported fields.

RELATED TOPICS

[Creating Web Views](#)

[Exporting Queries](#)

[Editing Field Attributes](#)

[About Adding Fields](#)

[Setting Default Values for Fields](#)

EXPORTING QUERIES

Queries control which issues a user can see in a Web view. For example, if **My Submitted Issues** is the only query available in a Web view, users can see only the issues they submitted.

To ensure that users cannot access any issues other than the ones found by the exported queries, you should hide the ad hoc query editor. The Ad-hoc Query editor allows users to see any issue (for example, by running a query like **Issue Number > 0**, which returns all issues).

In the **Queries** section, the **Available** box lists all the queries that you can export to the Web view. The **Export to View** box lists all the queries selected for export to the Web view.

To export queries:

- 1 In the shortcut bar, click **Queries** .
- 2 In the **Available** list, click a query.
- 3 Click  to move the query from the **Available** box to **Export To View**.

NOTES

- To move all queries from **Available** to **Export To View**, click .
- To make a query the default for the web view, click **Make Default**. This query will be automatically selected when a new user first logs in; however, their preferences will take precedence from that point on.

RELATED TOPICS

[Creating Web Views](#)

[Exporting Fields](#)

[About Queries](#)

ADDING THE CHILD ISSUES TAB

Web views can have a **Child Issues** tab that allows users to link one or more child issues to a parent issue.

To add the Child Issues tab:

- 1 In the shortcut bar, click **Fields** .
- 2 In the **Tab** list, click **Child Issues**.
- 3 In the **Available** list, click the **Child Issues** field and then click .

The **Child Issues** field is a special grid control that allows users to build a list of child issues.

- 4 If you want to allow users to control whether child issues can have a different Progress value than their parent, export the **Progress controlled by parent** field (**Details** tab).

By default, child issues have the same substate as their parent. Users can set **Progress controlled by parent** to **No** to override this default behavior.

- 5 Regenerate the Web view.

EDITING FIELD ATTRIBUTES

The fields on a Web view are arranged in a two-column layout inside an HTML table. The order of the fields is determined by the order in which they are listed in the **Export to View** list.

The appearance and layout of the fields can be customized through field attributes, which control options such as alignment, width, and height.

To edit field attributes:

- 1 In the shortcut bar, click **Fields** .
- 2 In the **Tab** list, click a tab. In the **Export to View** list, click a field.
- 3 Click **Attributes** to open the Attribute editor.

The Attribute editor displays the attributes for the selected field.

- 4 After you open the Attribute editor, you can edit the attributes for another field by clicking the field in the **Field Name** list.

RELATED TOPICS

[Using Field Variables](#)

[Setting Default Values for Fields](#)

[Adding Pop-up Editors](#)

[Aligning Fields](#)

[Automatically Updating Choice Lists](#)

[Changing Field Captions](#)

[Spanning Columns](#)

[Applying CSS Styles](#)

[Displaying Read-Only Fields](#)

[Inserting Custom HTML Code](#)

[Text Boxes](#)

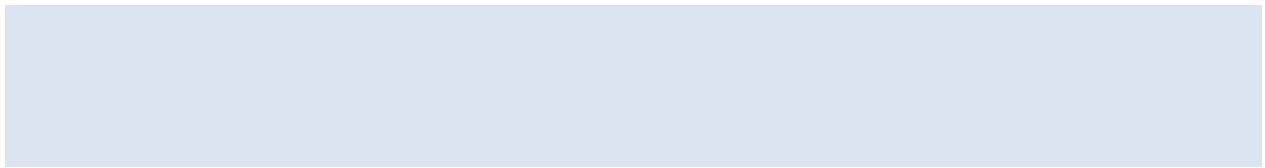
[Multi-Choice Lists](#)

[Configuring Choice Lists](#)

[Linking User Details](#)

[Adding URL Buttons](#)

[Hiding Fields](#)



USING FIELD VARIABLES

Field variables allow you to insert the name, value, or id of a field into an attribute. The field variables are particularly useful with the **Html code after control and URL** attributes.

Variable	Inserts
%fieldvalue%	The current value of the field. For choice lists, %fieldvalue% is the ID of the choice.
%fieldname%	The name assigned to the input control (for example, cbo_18_cboPriority).
%fieldid%	The numeric ID of the field (this is the nID of the field in tblDtsFields).

RELATED TOPICS

[Inserting Custom HTML Code](#)

[Adding URL Buttons](#)

ADDING POP-UP EDITORS

A Memo field such as the **Detailed Description** field has an Editor attribute that controls how the content of the field is edited.

Value	Description
<None>	The Memo field is an editable text area in the Web view.
HTML	The contents of the Memo field are edited in a pop-up HTML editor.
Text	The contents of the Memo field are edited in a pop-up Text editor.

ALIGNING FIELDS

Alignment

Aligns a field (both the caption and the input control). Possible values are **Left**, **Right**, or **Center**. By default, all fields are left-aligned.

Contact: RIGHT	<None>
Substate: CENTER	
Severity: LEFT	

Vertical Alignment

Sets the vertical positioning of a field (both the caption and the input control). Possible values are: **Top**, **Middle**, and **Bottom**. The default is **Top**.

Contact: TOP	<None>
Substate: MIDDLE	
Priority: BOTTOM	

RELATED TOPICS

[Editing Field Attributes](#)

AUTOMATICALLY UPDATING CHOICE LISTS

Automatically Update List

If **Yes**, the choice list is updated each time a user logs on to the Web view. To turn on the automatic update of choice lists, you must regenerate the view.

If **No**, the choice list is updated only when the Web view is regenerated.

NOTES

Automatically updating a large number of choice lists may affect performance. You should automatically update only the choice lists that change frequently.

RELATED TOPICS

[Editing Field Attributes](#)

[Should you automatically update choice lists?](#)

[Generating Web Views](#)

CHANGING FIELD CAPTIONS

The **Caption** attribute specifies a text label for a field. Note that reports and the **Field** list in the Ad-hoc Query editor still use the original field name, not the value of the **Caption** attribute.

SPANNING COLUMNS

FIELDS

Set **Column Span** to **1** to have the field span a single column, or **2** to have the field span the entire width of the page.

To make a field span 2 columns, you must also set the **CSS Class** to **MemoFieldWidth**. Use the **SpanTwoWidthOne** class if you want a field to span two columns but still be one column wide (that is, be alone on a line).

For example, in the default Vizor Web view, the **Priority** field uses the **SpanTwoWidthOne** class. For example, in the default BugTrk / Customer view, the **How Found** field on the **Detail** tab uses the **SpanTwoWidthOne** class.

CAPTIONS

Set **Caption Span** to **1** to have the caption span a single column, or **2** to have the caption span the entire width of the page.

RELATED TOPICS

[Editing Field Attributes](#)

APPLYING CSS STYLES

CSS Class

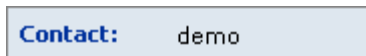
Name of a style class defined in the cascading style sheet (**CensusMain.css**) used by the Web view. This class is applied to the controls (for example, the <input> form element for a text box). By default, fields use the classes **ComboBoxWidth**, **MemoFieldWidth**, and **TextFieldWidth**. These classes set the width and border of the control. **ComboBoxWidth**, and **TextFieldWidth** set the width to one column, and **MemoFieldWidth** sets the width to two columns.

The CSS Class attribute does not apply to read-only/disabled fields. These fields use the **readOnlyStyle** class.

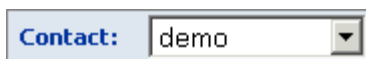
DISPLAYING READ-ONLY FIELDS

You can display read-only fields as either text or as disabled controls.

Read-only field displayed as text

A screenshot of a web form showing a label 'Contact:' in blue text followed by the text 'demo' in a standard font, all contained within a light blue rectangular border.

Read-only field displayed as a disabled control

A screenshot of a web form showing a label 'Contact:' in blue text followed by a disabled text input field containing the text 'demo'. The input field has a light gray border and a small downward arrow on the right side.

In a read-only Web view, all fields are read-only. In normal and submit-only Web views, read-only fields are fields that have been disabled through the Field editor.

To set the default for all read-only fields in the Web view:

- 1 In the shortcut bar, click **Web View** .
- 2 Click  (opposite **Attributes**) to open the Web View Attributes dialog.
- 3 Set **Display All Read-Only Fields As Text** to **Yes** or **No**.

By default, this setting applies to all read-only fields in the Web view.

To override the default for a specific read-only field:

- 1 In the shortcut bar, click **Fields** .
- 2 In the **Tab** list, click the tab that contains the field.
- 3 In the **Export to View** list, click the field and then click **Attributes**.
- 4 Change **Display Read-Only Field As Text** to **No** or **Yes**.

When **Display Read-Only Field As Text** is set to **Default**, the field uses the value of the Web view attribute **Display All Read-Only Fields As Text**.

NOTES

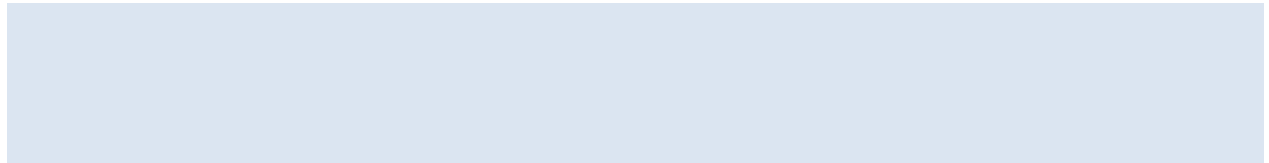
- If you display a read-only memo field as a disabled control, the scroll bar is disabled, so the user may not be able to see everything in the field.
- You must regenerate the Web view after you change the Web view attribute or the field attribute.

RELATED TOPICS

[Editing Field Attributes](#)

[Generating Web Views](#)

[Making Fields Read-Only Per View](#)



WIDTH

Width in characters of a single-line text box, or width in pixels of a list box.

The CSS classes assigned to text boxes and list boxes override the **Width** attribute. These CSS classes (**TextBoxWidth** and **ComboBoxWidth**) specify the widths of the input controls.

RELATED TOPICS

[Applying CSS Styles](#)

INSERTING CUSTOM HTML CODE

Html code before/after control

HTML inserted before and after the control in the generated HTML page. Must be a single line of text. For example, suppose you want to add a help icon that displays a tooltip.



A simple way to implement field-level tooltips is to put this HTML in the **Html code after control** attribute:

```

```

This inserts a help icon that displays a tooltip when a user points to the icon.

A more generic approach would be to write a javascript function in CustomCode.js that returns a string of HTML.

```
function getHelpIcon ( tooltip ) {  
    var str;  
    str = '';  
    return str;  
}
```

The **Html code after control** would then call this function to get the HTML:

```
<script type="text/javascript" language="javascript">document.write(  
    parent.objCustomCode.getHelpIcon( 'Tooltip text')  
);  
</script>
```

You must declare your function in CustomCode.js. Look for the public declarations at the bottom of the file, and add this line:

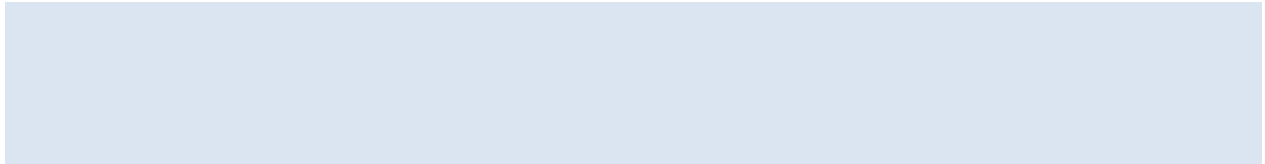
```
this.getHelpIcon = getHelpIcon;
```

NOTES

In the Html code before/after control attributes, you can use the string **%fieldname%** to refer to the name of the control. (The name of a control is the value of the name attribute on the form element.) For example, you could pass %fieldname% into the getHelpIcon() function, which would then return the help for that field.

RELATED TOPICS

[Using Field Variables](#)



ADDING URL BUTTONS

URL buttons open a new browser window and load the document specified by the **URL** attribute. A URL button is associated with a field, and can use the field value as part of the URL.

For example, suppose you wanted to record the ID number of a Microsoft KB article that explains how to resolve a problem.



To add a URL button that opens the specified KB article in a browser window, set the **URL** and **URL Button CSS** attributes as follows:

```
URL = http://support.microsoft.com  
      /default.aspx?scid=kb;en-us;%fieldvalue%
```

```
URL Button CSS = URLButton
```

In the **URL** attribute, %fieldvalue% is a placeholder for the KB article ID number entered in the field.

If you omit the http:// prefix from the URL, Vizor automatically adds it when you generate the Web view.

The **URL Button CSS** attribute identifies the CSS classes used for the button. Vizor includes several different URL button classes you can use:



URLButton



MailToButton



RemoteControl

URL buttons always open a new browser window, even when you specify a URL like mailto:xsupport@company.com.

RELATED TOPICS

[Editing Field Attributes](#)

[Using Field Variables](#)

[Customizing URL Buttons](#)

TEXT BOXES

Number of Characters

Maximum number of characters per line in the text box.

RELATED TOPICS

[Editing Field Attributes](#)

MULTI-CHOICE LISTS

Number of Visible Items

Number of items visible in a multi-choice list. Users can always see items that are outside the visible window by scrolling the list box.

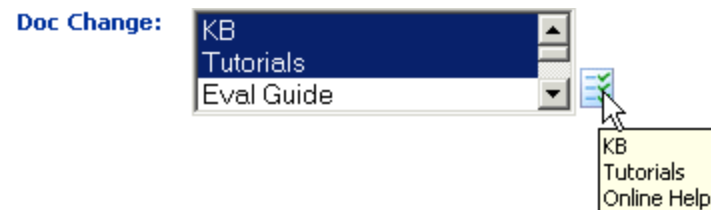
Show Selected Items In Text



Shows a comma-separated list of the selected items in a text box above the multi-choice list. This helps users see what items are selected.

Show Selected Items In Tooltip

Shows a comma-separated list of the selected items in a tooltip. This helps users see what items are selected in a multi-choice list.



CONFIGURING CHOICE LISTS

The tblUser table can contain a large range of entries. For example, a big organization can have several thousand users and contacts, whereas a smaller organization may have only a few tens of entries. Use the Select from dialog attribute to specify the method which support staff can use to make a selection from single- or multi-choice lists populated from the tblUser table.

To specify the type of choice list control you want to use:

- 1 In the Web View Editor, select the Web View that contains the fields you want to work with.
- 2 Select **Fields** in the shortcut bar.
- 3 Select a field you want to customize, and click the **Attributes** button. For example, to select the control type for the **Contact** field in the Console's **Overview** tab, select **Contact**.
- 4 Choose a control option:

To display the data in a search dialog, click the **Select from dialog** attribute and click **Yes**. A browse  button is displayed alongside the field to enable users to open the Search dialog. (Select from dialog = Yes is used by default.)

To display the data in a drop-down list, click the **Select from drop-down list** attribute and click **No**.

We recommend you select the **No** option only when the field can contain fewer than 100 entries as it can take several minutes to populate a large drop-down list.

- 5 Click **Close** to save your changes.

RELATED TOPICS

[Editing Field Attributes](#)

LINKING USER DETAILS

When dealing with issues, support staff often need to request more information from users or ask them to perform checks on their computers. To help support staff contact users, you can configure Vizor to make user details available through a dialog linked to the **User**, **Contact** or **Company** fields.

To link to user details:

- 1 In the Web View Editor, select the Web View that contains the fields you want to work with.
- 2 Select **Fields** in the shortcut bar.
- 3 Select **User** and click the **Attributes** button.
- 4 Click the **Link to user details** attribute and click **Yes**.
- 5 Click **Close** to save your changes.

The contents of the field are underlined when users mouse over it in the Web View. This indicates a link to the contact details of the selected person.

RELATED TOPICS

[Editing Field Attributes](#)

HIDING FIELDS

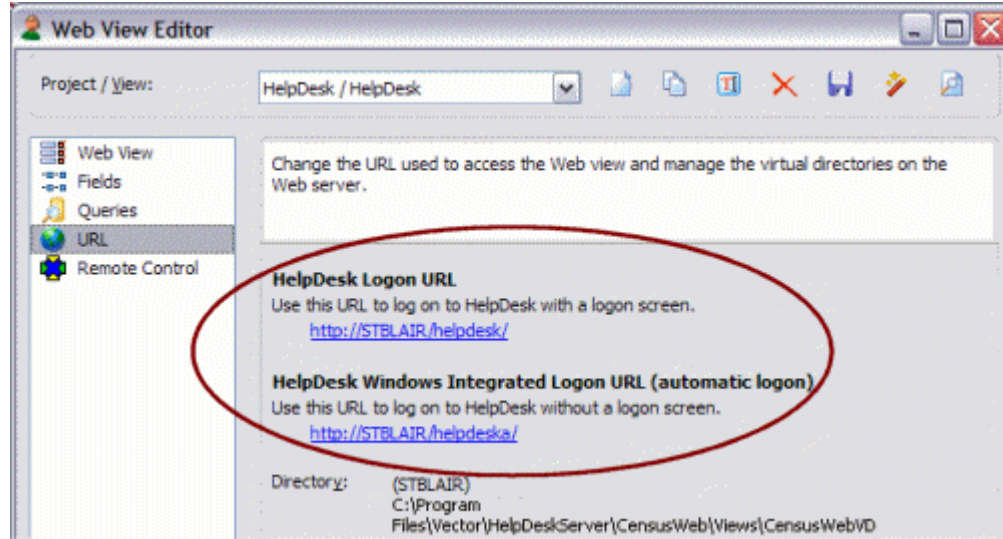
Setting **Visible** to **No** hides the field in the Web view. The field is still included in reports and listed in the Field list of the Ad-hoc Query editor.

RELATED TOPICS

[Editing Field Attributes](#)

POINTING USERS TO WEB VIEWS

The URLs for a normal or read-only Web view are displayed in the Web View Editor. The Vizor Logon URL is for users with Vizor users accounts. The Windows Integrated Logon URL is for users whose [Windows user accounts](#) have been imported into Vizor.



The default page at the URL is **logon.asp**.

For Submit-only and Knowledge Base views, you have to add the **logon.asp** file name and a query string to the URL displayed in the Web View Editor. For example:

```
http://server/vit/logon.asp?View=Vizor%2FReport%20Issue
http://server/vit/logon.asp?View=Vizor%2FKB
http://server/census/logon.asp?View=BugTrk%2FSubmit%20Issue
http://server/census/logon.asp?View=BugTrk%2FKB
```

The query string (the part after the question mark) specifies the project and view:

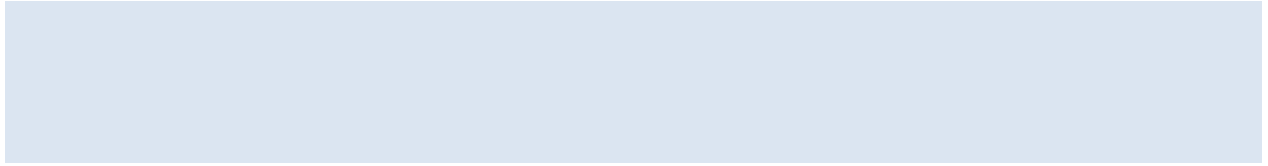
```
View=<project>%2F<view>
```

%2F is the escape sequence for a forward slash (/). Spaces in the view name are replaced with the escape sequence %20.

RELATED TOPICS

[Changing URLs and Directories](#)

[Changing the Windows Account](#)



CHANGING URLS AND DIRECTORIES

A Web view is accessed through a URL like **//server/vit**, where **server** is the name of the Web server computer, and **vit** is the name of a virtual directory.

On a Web site hosted by Microsoft Internet Information Services (IIS), a virtual directory is an alias (a short, descriptive name) that is mapped to a directory on the Web server.

The Web View Editor includes a Web View wizard to help you:

- Change the URL (for example, change **//nan/vit** to **//nan/CallTracking**) by changing the virtual directory. Change the URL (for example, change **//nan/BugTrk** to **//nan/bugs**) by changing the virtual directory.
- Change the directory where the Web view is published.

There are two ways to change the URL. You can create a new virtual directory, or you can use a existing virtual directory.

To change the URL by creating a new virtual directory:

- 1 In the shortcut bar, click **URL** .
- 2 Click  to open the Web View Wizard and then click **Next**.
- 3 Click **Advanced**, click **Create a new URL and a new Directory**, and then click **Next**.
- 4 Type the name of your Web server and click **Next**.
- 5 In the **Web Site** list, click the Web site where you want to create the virtual directory.
- 6 In the **Virtual Directory** box, type the name of the new virtual directory.
- 7 In the **Path** box, select the directory where you want to put the generated Web view files.
- 8 If you need to change the user account used for anonymous access to the virtual directory, select the **Show advanced options** check box.
- 9 Click **Next** and then click **Finish**.

To change the URL by using a different virtual directory:

- 1 In the shortcut bar, click **URL** .
- 2 Click  to open the Web View Wizard and then click **Next**.
- 3 Click **Use existing URL** and click **Next**.

4 In the **Web Site** list, click the Web site that contains the virtual directory you want to use.

5 In the **Virtual Directory** list, click the virtual directory you want to use.

By default, the **Virtual Directory** list contains only the virtual directories used by Vizor Web views. To show all the available virtual directories, select the **Show all virtual directories from IIS** check box.

6 Click **Next** and then click **Finish**.

To change the directory:

1 In the shortcut bar, click **URL** .

2 Click  to open the **Web View Wizard** and then click **Next**.

3 Click **Advanced**, click **Update existing virtual directory**, and then click **Next**.

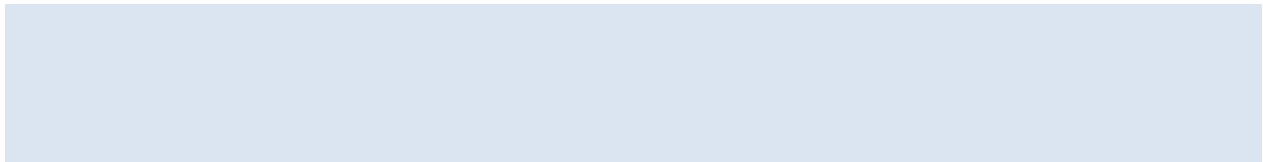
4 Click **Next** and then click **Next** again.

This assumes you want to change just the directory, not the Web server, Web site, or virtual directory.

5 In the **Path** box, select the new directory where you want to put the generated Web view files.

6 If you need to change the user account used for anonymous access to the virtual directory, select the **Show advanced options** check box.

7 Click **Next** and then click **Finish**.



CHANGING THE WINDOWS ACCOUNT

Vizor needs a Windows domain account to use for:

- Anonymous access to the virtual directories on the Web server.
Anonymous access is required to run any Web view or Vizor Web Admin, which are all stored in virtual directories on the Web server.
- Access to Vizor databases in a distributed configuration, where the databases are not on the Web server.
- Access to attachments, which may be stored on a different computer than the Web server.
- Access to Vector Asset Management site databases

By default, Vizor uses the CensusUser account, which is created when you install Vizor. The default password for the CensusUser account is nd5kvmWJ.

The CensusUser account is a Windows domain account.

NOTES

If you change the password for the CensusUser account, or delete the user account and use another one, you must update the virtual directories in the Internet Service Manager (Windows) or Internet Services Manager (Windows 2003, 2000, XP).

To change the Windows account used for anonymous access to a virtual directory:

- 1 Open the Web View editor. In the shortcut bar, click **URL** .
- 2 Click  to open the Web View Wizard and then click **Next**.
- 3 Click **Advanced**, click **Update existing virtual directory**, and click **Next**.
- 4 Click **Next** until you get to the Virtual Directory Properties dialog.
- 5 Select the **Show Advanced Options** check box.
- 6 In the **User Name** box, enter the name of the user account.

Click **Browse** to locate a domain user account.

- 7 In the **Password** box, type the user account password.

You can also use Vizor Admin to change the Windows account.

To change the Windows account used by Vizor:

- 1** Open Vizor Admin.
- 2** On the **Tools** menu, click **Windows Account**.
- 3** Enter the name and password of the domain user account you want to use.

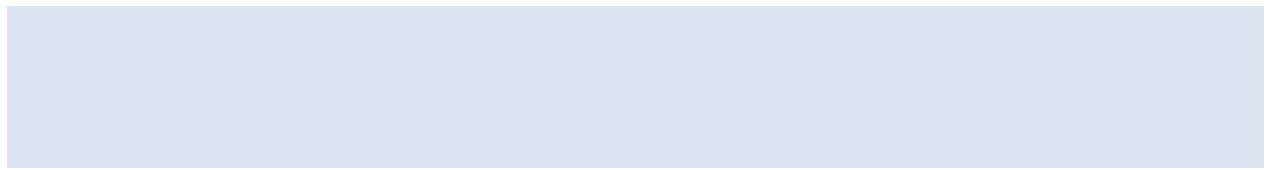
This user account is used for anonymous access by all subsequent Web views you generate.

NOTES

When you change the user account used for anonymous access with the Web View Editor or Vizor Web Admin (Notifications tab), all required permissions and rights are automatically set.

RELATED TOPICS

[Windows Permissions](#)



SETTING DEFAULT VALUES

To set default values for a Web view:

- 1 Log on to the Web view.
- 2 In the Web view toolbar, click **Defaults** .
- 3 In the Default Values dialog, set the field values you want to use as defaults for new issues.
- 4 Click **Apply**.

This creates a file named MqDefaultValues.js in the folder

```
CensusWebVD\<<project>_<view>\Users\<<logon-name>
```

where <logon-name> is the name you used to log on to the Web view.

- 5 Copy the file

```
\CensusWebVD\<<project>_<view>\Users\<<logon-name>\MqDefaultValues.js
```

to

```
\CensusWebVD\<<project>_<view>\Users\Users\MqDefaultValues.js
```

The folder Users\Users applies to all users that log on to the Web view.

- 6 Copy Users\Users\MqDefaultValues.js to the CustomizedFiles folder.

For example, if you want the default values to apply to all Web views of the HelpDesk project, copy MqDefaultValues.js to:

```
CUSTOMIZEDFILES\#Project#HelpDesk\#AllWebViews#\Users\Users
```

If you want the default values to apply only to the HelpDesk_HelpDesk view, copy MqDefaultValues.js to:

```
CUSTOMIZEDFILES\#Project#HelpDesk\#WebView#HelpDesk_HelpDesk\Users\Users
```

NOTES

- Users can override these default values by setting their own defaults.
- You can remove the **Defaults** button from Web views by disabling the Default Value Editor feature.

TEMPLATES AND THE CUSTOMIZEDFILES FOLDER

The files that make up a Web view (such as .JS, .CSS, .HTML, and .ASP files) are generated from templates. If you edit the original templates, your customizations apply to all Web views. In addition, if you edit the generated files for a Web view, you must reapply your changes every time you generate the view.

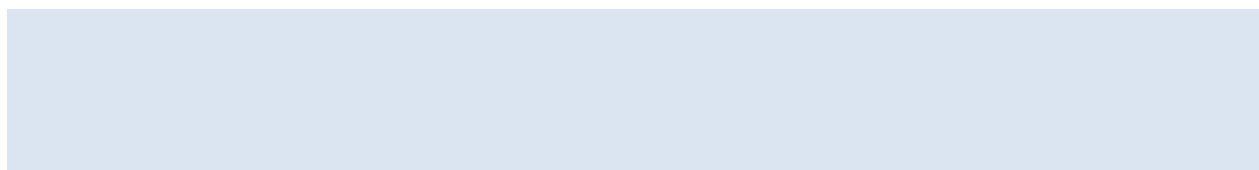
To customize specific Web views without editing the generated files, you put your customized files in a special **CustomizedFiles** folder. Each time you generate the view, the Web View Editor copies your customized files from the **CustomizedFiles** folder into the view.

RELATED TOPICS

[Customizing Web Views](#)

[Customizing Shared Web View Files](#)

[Overriding Customizations](#)



CUSTOMIZING WEB VIEW FILES

To customize Web view files:

- 1 In the **<ConnectServer>\CensusWeb** folder, create a **CustomizedFiles** folder.

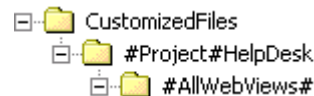
<ConnectServer> represents the folder where your Vizor Server is installed.

- 2 In the **CustomizedFiles** folder, create a folder structure to hold the customized files.

To customize files for all Web views of all projects, create this folder structure:

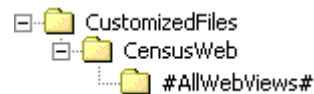


To customize files for all Web views of a specific project, create a folder structure like this:



and replace **vit** with the name of the project.

To customize files for all Web views accessed through a given URL, create a folder structure like this:

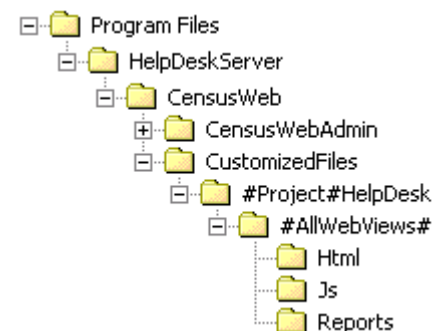


and replace **CensusWeb** with the last part of the URL. (For example, if the actual URL is **http://SERVER/vit**, replace **CensusWeb** with **Vizor**).

Use this approach if you generate Web views outside the ConnectServer folder.

- 3 Copy the files you want to customize from the view into the **CustomizedFiles** folder.

You must recreate the same folder structure. For example, if you want to customize files in the **HTML**, **JS**, and **Reports** folders of Web views of the Vizor project, you must create this folder structure:



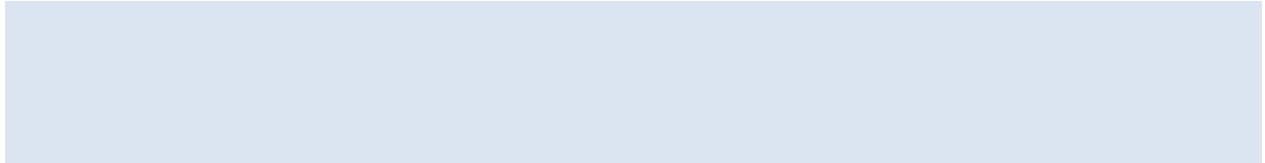
- 4 Customize the files and generate the view.

RELATED TOPICS

[Templates and the CustomizedFiles Folder](#)

[Customizing Shared Web View Files](#)

[Overriding Customizations](#)



CUSTOMIZING SHARED WEB VIEW FILES

You can also customize the files that are shared by Web views accessed through the same URL. These are the files found in the **CensusWebVD** folder, such as **logon.asp**.

To customize the files in CensusWebVD for all Web views:

Put the files in the folder

CustomizedFiles\#AllProjects#

-or-

CustomizedFiles\<virtual directory>

To customize files for all Web views accessed through a given URL:

Put the files in the folder

CustomizedFiles\<virtual directory>

For example, if the URL is //server/vit, then put the customized files in a folder named CustomizedFiles\vit.

Use this approach if you generate Web views outside of the ConnectServer folder.

RELATED TOPICS

[Templates and the CustomizedFiles Folder](#)

[Customizing Web Views](#)

[Overriding Customizations](#)

OVERRIDING CUSTOMIZATIONS

To override the customizations in **#AllWebViews#** for a given Web view, you can create a **#WebView#<project>_<view>** folder. For example:



The files in **#WebView#Vizor_Vizor** override all **#AllWebViews#** customizations.

You can create a **#WebView#<project>_<web view>** folder in any of the three customization folders (**#AllProjects#**, **#Project#<project>**, and **CensusWeb**).

RELATED TOPICS

[Templates and the CustomizedFiles Folder](#)

[Customizing Web Views](#)

[Customizing Shared Web View Files](#)

CUSTOMIZING THE WEB VIEW INTERFACE

You can customize most of the HTML and ASP pages that make up a Web view. If you edit the templates, make sure to backup the original files before you begin your customization.

CensusMain.css

Style sheet for a Web view. The **normalfield**, **requiredfield**, and **disabledfield** classes define the appearance of the fields.

TmplCensusMain.htm

Defines the frame structure for the frames that make up the Web view page. You can change the size and position of each frame, add and remove scrollbars, and set the frame border size.

TmplToolbar.htm

Defines the Web view toolbar.

TmplSummaryList.htm

Defines the Summary List. **You cannot customize this page.**

TmplSQE.htm

Defines the Ad-hoc Query Editor.

Tabs.asp

Defines the layout of the tab navigation buttons.

TmplCmdBar.htm

Contains the navigation controls.

When editing these files, do not change references to file names, the <WC@ ... > tags, or any of the Javascript code.

CHANGING THE DATE AND TIME FORMATS FOR WEB VIEWS

The format of the dates and times entered and displayed in a Web view are controlled by the **Regional Settings** of the Web server, not the **Regional Settings** of the local computer.

Web views use the short date format.

ADMINISTERING VIZOR ON THE WEB

Vizor Web Admin is a Web-based administration tool that allows you to:

- Add and edit users and contacts.
- Define workflow rules.
- Create new tabs.
- Set up choice fields.
- Enable e-mail notifications.
- Migrate projects to SQL Server and move projects to different SQL Server computers.
- Move attachments to a different computer.
- Integrate InventoryLink VSS databases to Vizor projects.

LOGGING ON

You can log on to Vizor Web Admin directly from the Web View Editor, using the same logon account you used for the Web View Editor.

To log on to Vizor Web Admin

Do one of the following:

- In your Web browser, go to **//server/vitadmin**, where server is the name of your Web server.
- In Vizor Admin, click **Tools > Web > Web Admin**.
- At the Web server, click **Start > Programs > Vector > Vizor > Vizor Web Admin**
- In the Web View editor, click **Admin** and then click **Log On**.

GENERATING VIZOR WEB ADMIN

When you install the Vizor Web Server, the setup program generates a copy of Vizor Web Admin. If necessary, you can generate another copy of Vizor Web Admin. For example, you may want to change the URL used to access Vizor Web Admin.

To generate Vizor Web Admin:

- 1 In the Web View editor, click **Admin**.
- 2 If required, [change](#) the URL and directory for Vizor Web Admin.
- 3 Click **OK**.

WINDOWS PERMISSIONS

When you log on to Vizor Web Admin, it uses your Windows credentials to run tasks if you are a member of the CensusAdminsGroup group or the local Administrators group on the Vizor Server computer. Otherwise, Vizor Web Admin runs tasks as CensusUser.

Members of the CensusAdminsGroup group (or of the local Administrators group on the Vizor Server computer) have the required permissions to perform any task with Vizor Web Admin. The CensusUser account, on the other hand, cannot perform IIS-related tasks such as unloading virtual directories (when logging off users), and may not be able to relocate databases to other computers.

During installation, the Setup program creates the CensusAdminsGroup group and adds all members of the domain Administrators group to the CensusAdminsGroup group. Setup also adds the CensusAdminsGroup to the local Administrators group on the Vizor Server computer. The purpose of the CensusAdminsGroup is to control access to Web Admin.

Related Topics

[Improving Web View Performance](#)

CHAPTER 23 - MAINTAINING YOUR VIZOR SYSTEM

RELOCATING DATABASES TO SQL SERVER

Vizor uses Microsoft SQL Server as the primary database backend. In addition, you can also run all or some databases in Jet/MSAccess.

You can relocate (migrate) databases from Jet/MSAccess to SQL Server, or from one SQL Server to another.

There are two types of databases you can relocate:

- Global databases are the databases shared by all projects: the users database (users.mdb), the Web views database (censusweb.mdb), and the licenses databases (licenses.mdb).
- Project databases contain issues and project definitions (such as the definitions of fields, queries, and reports).

To relocate databases to SQL Server:

1 Log on to Vizor Web Admin:

- In Vizor Admin, click **Tools > Relocate > Databases** to log on to Vizor Web Admin, where you can use the **Relocation** tab to relocate databases to SQL Server.
- In your Web browser, go to `//server/vitadmin`, where **server** is the name of your Web server.

2 On the **Relocation** tab, under **Databases to relocate**, select the databases you want to relocate.

3 In the **Relocate databases to this SQL Server** list, click the computer to which you want to relocate the databases. In the **Relocate databases to this SQL Server** list, click the Web server computer. MSDE 2000 databases must be on the same computer as Vizor.

To relocate to a SQL Server instance, click **Add a SQL Server** to this list and then type the instance name using the syntax `<computer name>\<instance name>`.

4 In the **New location for database files** box, enter the name of a folder on the Server where the database files (*.mdf, *.ldf) can be stored.

The new location must be the path as seen from the Web server. If the SQL Server is in a different computer, enter a UNC., otherwise, enter a local path.

5 Click **Apply** to relocate the databases.

6 During the relocation, you may be prompted for the password for the **sa** SQL Server login account. Vizor needs the password so it can initialize the SQL Server.

7 After the databases are relocated, regenerate your Web views.

If you relocated the global databases, regenerate all Web views. If you relocated project databases, regenerate the Web views of the relocated projects.

NOTES

- We recommend that all databases be stored on the same SQL Server computer. This improves performance by reducing the amount of data sent over the network.
- When relocating databases from one SQL Server to another, the locations of the physical data files (.mdf,.ldf) on both the source and target computers must be accessible from the Web server. Also, the user running Vizor Web Admin must have full permission to move and create files on the locations .
- Relocating databases logs off users and prevent users from logging on, so choose a time when there is little or no activity on Vizor.
- When you relocate a database to SQL Server, Vizor creates a SQL Server database with one data file and one log file. After the database is created and is accessible in SQL Server, database administrators can add more files or configure the database as required.
- Vizor creates a SQL Server login account named **censusapplication**, which is used to connect to the SQL Server.

RELATED TOPICS

[Troubleshooting Relocation](#)

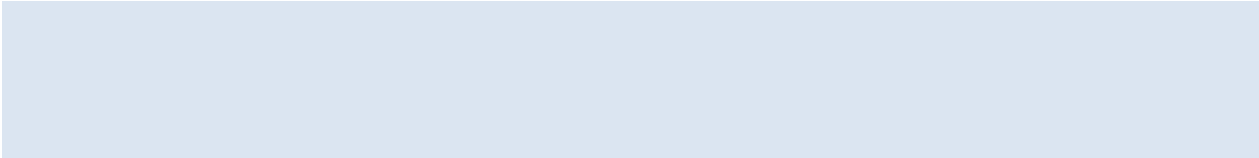
[Generating Web Views](#)

[Relocating Attachments](#)

[Editing Project Properties](#)

[Troubleshooting Vizor Web Admin](#)

[Releasing Databases](#)



TROUBLESHOOTING RELOCATIONS

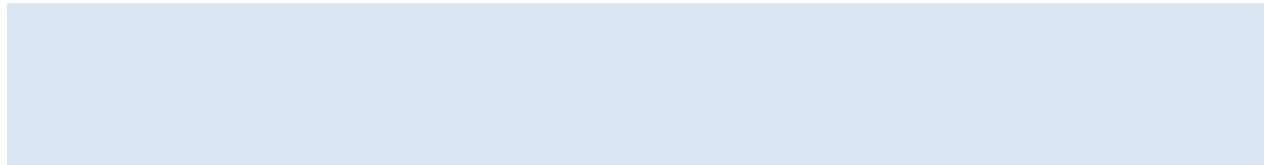
If Vizor Web Admin cannot contact or initialize the SQL Server during relocation:

- Check the SQL Server is running.
- Check the SQL Server is configured to use SQL Server and Windows authentication (Mixed Mode authentication).
- Verify that the SQL Server login account **sa** exists, is not disabled, and has permissions to log into the master database in the SQL Server. **sa** must be system administrator in SQL Server.
- Stop any antivirus software that is running on the SQL Server computer or the Web server computer.
- Verify that the Vizor Web Server computer and the SQL Server computer can communicate through the network (for example, use Ping to test connectivity).
- Try using the IP address instead of the computer name to identify the SQL Server computer.
- To relocate to a SQL Server instance, click **Add a SQL Server to this list** and then type the instance name using the syntax <computer name>\<instance name>.
- Verify that both the SQL Server and the Web server are using the same protocol (by default, TCP/IP). To do this, use the SQL Server Server Network Utility on the SQL Server computer, and the SQL Server Client Network Utility on the Web server.

RELATED TOPICS

[Relocating Databases to SQL Server](#)

[Releasing Databases](#)



RELOCATING ATTACHMENTS

When users upload attachments, the attachments are stored on the Web server. For example, the files attached to issue #3 in the Vizor project are stored in the folder

```
ConnectServer\vit\Data\Attachments
```

For security purposes, you may want to move the attachments to a separate file server.

To relocate attachments:

- 1 In Vizor Admin, click **Tools > Relocate > Attachments**.

This logs you on to Vizor Web Admin, where you can use the **Relocation** tab to move (relocate) attachments.

- 2 In Vizor Web Admin, click the **Relocation** tab and then click **Relocate Attachments**.

- 3 Under **Relocate the attachments for these projects**, select the projects whose attachments you want to move.

- 4 In the **New location for attachments** box, type or paste the name of an existing folder on another computer.

- The Windows account (by default, CensusUser) must have (full) permissions on the new location.
- The new location must be a UNC path (for example, \\sharename\Attachments) that the Web server can access.
- The new location is the root folder where Vizor creates folders to store the attachments. For example, if the new location is

```
//fileserver/vitWeb
```

then attachments to the Vizor project are stored in

```
//fileserver/VizorWeb/vit/Data/Attachments
```

If you created your own attachment field, then attachments are stored in

```
//fileserver/vitWeb/vit/Data/<fieldname>
```

- 5 Click **Apply** to relocate the attachments.

- 6 After the attachments are relocated, regenerate the Web views of the projects whose attachments were relocated.

NOTES

Relocating attachments logs off users and prevent them from logging on, so choose a time when there is little or no activity on Vizor.

RELATED TOPICS

[Relocating Databases to SQL Server](#)

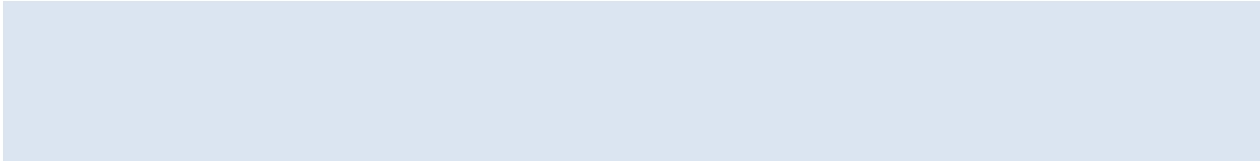
[Generating Web Views](#)

[Editing Project Properties](#)

[Troubleshooting Vizor Web Admin](#)

[Changing the Windows Account](#)

[Releasing Databases](#)



DELETING ISSUES

Vizor allows users to delete issues from two key locations - the web views, and Vizor Admin. Each issue loaded in Vizor can be deleted right from the web, provided the users are part of a group that has been given access to the feature. For more information about granting access to features, see [Defining User Groups](#). To delete a range of issues in one operation, users must have access to Vizor Admin. Both approaches come with the ability to recycle issue numbers, allowing them to be repurposed for future issues.

To delete issues using Vizor Admin:

- 1 In Vizor Admin, click **Delete Issues** on the **Project** menu.
- 2 In the **Issues to delete** box, type the issue numbers you want to delete.

You can type a specific issue number, a comma-separated list of issue numbers (for example: 1, 5, 7), or a range (for example, 1-25).
- 3 If you want to reuse the issue numbers of deleted issues, select the **Recycle Issue Numbers** check box.

Recycled issue numbers are assigned to new issues.

To delete issues from within the Vizor Views:

19. In the Vizor view select the Issue you wish to delete.
20. In the Issue Details pane, select the **Delete**  icon.
21. If you want to reuse the issue number, select the **Recycle Issue Numbers** check box.

Recycled issue numbers are assigned to new issues.

ABOUT THE LOGONS EDITOR

Vizor Admin includes a Logons editor (on the **Tools** menu) that lists all users logged on to Web views, Vizor Admin, Vizor Web Admin, and the Web View Editor.

The Logons editor displays the following information for each logon session:

Name

Logon name of the user.

Date/Time

Date and time that the user logged on.

Workstation

The user's IP address or computer name.

Project

If the user is logged on to a Web view, the name of the project is displayed.

Virtual Directory

If the user is logged on to a Web view, the virtual directory is displayed. The virtual directory is the final part of the URL used to access the Web view. For example, if the virtual directory is named **vit**, then the URL is **http://server/vit**. You can use the Web View Editor to determine which views are at the specified URL.

Virtual directories such as connect00, connect01, and connect02 are all accessed through the //server/vit URL.

Application

The component you want to work with. For example: Web View, Web View Editor, Admin, or Web Admin.

RELATED TOPICS

[Logging Off Users](#)

[Releasing Databases](#)

LOGGING OFF USERS

To log off a specific user:

- 1 On the **Tools** menu, click **Logons Editor**.
- 2 In the list of users, click a user.
- 3 Click **Log Off**.

To log off all users:

- 1 On the **Tools** menu, click **Logons Editor**.
- 2 Click **Log Off All**.

Before logging off a user, a Web view displays a message telling the user that Vizor is shutting down. You can give users a bit more time by sending an immediate message. To do this, set the **Send immediate shutdown message** project option to **Yes**.

RELATED TOPICS

[About the Logons Editor](#)

[Logging Off Users](#)

SETTING THE MONITOR INTERVAL

The Monitor Interval option (**Tools>Options**) controls how long (in seconds) Vizor waits before:

- Checking for messages from a Web view
- Checking for user activity in a Web view
- Terminating a user session when logging off users.

The default interval is 60 seconds. If your Web server is on a slow Internet connection and you want to reduce network traffic, you can increase the monitor interval. Too large an interval, however, affects the time administrators have to wait when performing an action that involves logging off users.

ABOUT THE INTEGRITY EDITOR

The Integrity editor (available from the **Tools** menu) allows you to remove database locks and validate the integrity of your databases. The integrity tasks apply to both SQL Server and Microsoft Access databases.

RELATED TOPICS

[Remove Locks](#)

[Validate Integrity of Fields](#)

[Validate Relationships](#)

[Validate Integrity of Issues](#)

REMOVE LOCKS

LOCKS ON PROJECT DEFINITIONS

Removes locks on project definitions, such as reports, queries, fields, and sorts.

If the locked by another user message is displayed when you try to use Vizor Admin, run the **Remove Locks** task to remove the locks.

LOCKS ON USERS AND LOGON SESSIONS

Removes locks on users and logon sessions. Users are locked while someone is editing a record in the users database (the users.mdb file or, in SQL Server, the **USERS_MDB** database).

Sessions shown in the Logons editor can be locked. For example, you may not be able to remove a logon that is several weeks old because it is locked. Run **Remove Locks** to unlock the logon.

RELATED TOPICS

[About the Logons Editor](#)

[Logging Off Users](#)

[Releasing Databases](#)

[About the Integrity Editor](#)

[Validate Integrity of Fields](#)

[Validate Relationships](#)

[Validate Integrity of Issues](#)

VALIDATE INTEGRITY OF FIELDS

This task verifies that the issue database and the project definition database are synchronized and have the same set of fields. The task also checks primary keys and indexes.

By running the **Validate Integrity of Fields** task, you can synchronize the backup issue database with the current project definitions database.

VALIDATE RELATIONSHIPS

This task verifies that built-in related fields are set correctly in the issue database. In the Vizor project, the only [built-in related fields](#) are **State** and **Progress**.

Validate Relationships does not apply to field relationships created with the Workflow editor.

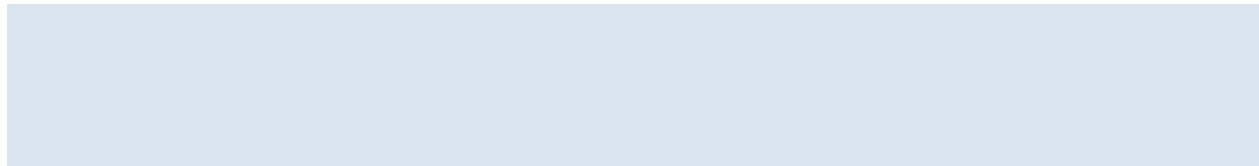
RELATED TOPICS

[About the Integrity Editor](#)

[Remove Locks](#)

[Validate Integrity of Fields](#)

[Validate Integrity of Issues](#)



VALIDATE INTEGRITY OF ISSUES

This task verifies that the issue database and the revision history are in-sync, by checking the revision numbers stored in the different tables of the database.

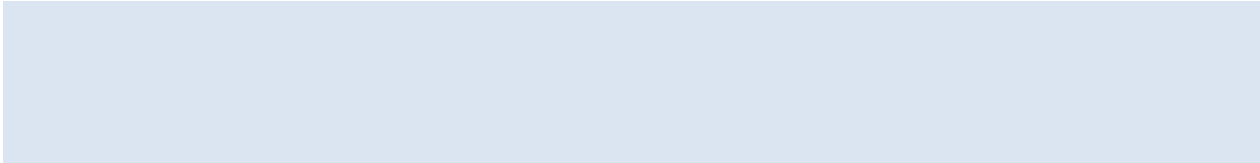
RELATED TOPICS

[About the Integrity Editor](#)

[Remove Locks](#)

[Validate Integrity of Fields](#)

[Validate Relationships](#)



ENABLING AUTO REPAIR

If you have problems running Vizor Admin, use the **Enable Auto Repair** option on the **Tools** menu.

Enable Auto Repair tries to repair Vizor Admin by fixing broken references, linking tables, and recreating objects.

To turn on auto-repair, click **Enable Auto Repair** in the **Tools** menu, then exit Vizor Admin and log back on. While auto-repair is turned on, Vizor Admin runs in repair mode, which may be slower than normal mode.

To turn off auto-repair, click **Enable Auto Repair** again.

RELEASING DATABASES

Some Vizor administrative tasks start a shutdown process. Examples of such tasks include relocating databases, creating projects, applying changes in the Field Editor, generating Web views, and running integrity tasks.

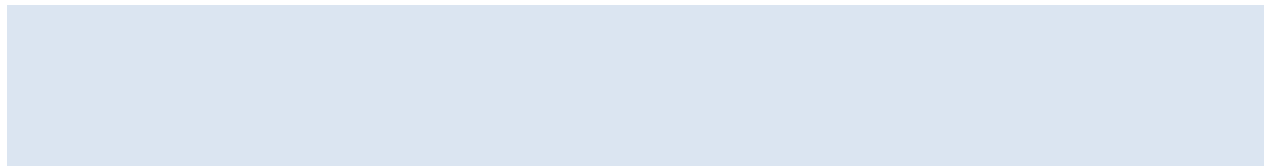
The shutdown process logs off users and releases databases. If you see a Still releasing databases message during the shutdown process, you can:

- Click **Cancel** to stop trying to release the databases, and start logging off users. Do this only if you are sure that no one is using the databases.
- Try to manually release the databases and then click **OK**. To release the databases:
 - Unload the virtual directories used by Vizor.
 - Stop the Mq Issue Agent service.
 - Exit all Vizor programs.
 - Exit any programs (such as Microsoft Access or SQL Server Enterprise Manager) that may be accessing the databases.

Note that if you set the option **Tools > Options > Always verify locked databases** to **No**, Vizor does not try to release databases during the shutdown process. In general, setting **Always verify locked databases** to **No** is not recommended, unless you know there is a process that regularly accesses the databases but does not affect Vizor.

RELATED TOPICS

[About the Logons Editor](#)



UPLOADING LARGE ATTACHMENTS

On Windows Server 2003 systems, the default size limit for uploaded files is 4096 KB. If a user tries to upload an attachment that exceeds this limit, the user may get "The page cannot be displayed" or "Cannot find server or DNS Error" error messages.

To increase the upload limit for attachments:

- 1 Using a text editor, open:

```
\Program Files\Common Files\MetaQuest\Scripts\IIS\MqWVDCOnfig.vbs
```

- 2 Set the constant `UPLOAD_FILE_MAX_SIZE` to the maximum size (in bytes) for uploaded attachments. For example:

```
UPLOAD_FILE_MAX_SIZE = 1048576          ' 1 MB
UPLOAD_FILE_MAX_SIZE = 5242880          ' 5 MB
UPLOAD_FILE_MAX_SIZE = 10485760         '10 MB
```

- 3 Re-create the virtual directories where you want the new size limit to apply. Note that re-creating virtual directories logs off users.

- a Start the Web View Editor.
- b In the shortcut bar, click **URL** .
- c Click  to open the Web View Wizard.
- d In the Web View Wizard, click **Next** until you get to the last page of the wizard, then click **Finish**. This recreates the virtual directories associated with the existing URL.
- e Click **Yes** each time the Wizard asks if you want to re-create a virtual directory and reset its properties.

A URL (such as `//Server/vit`) has several associated virtual directories (for example, `vit`, `connect00`, `connect01`, and `connect02`).

RELATED TOPICS

[Fixing 404 Errors](#)

[Changing URLs and Directories](#)

FIXING 404 ERRORS

When Vizor is installed on a Windows Server 2003 system, users may get 404 errors (File not found) when trying to open attachments. This is because IIS 6.0 on Windows Server 2003 only accepts requests for static files with extensions that it recognizes. You can prevent this problem by adding the extensions to MIME Types settings of the IIS server.

To define a MIME type for a specific extension:

- 1 Open **Internet Information Services**.
- 2 In the console tree, right-click the virtual directory used for attachments, then click **Properties**.
- 3 On the **HTTP Headers** tab, click **MIME Types**.
- 4 Click **New**.
- 5 In the **Extension** box, type a file name extension (for example, **.ini**).
- 6 In the **MIME Type** box, type the MIME type of the file (for example, **text/ini**).
- 7 Apply your changes.

NOTES

- You may need to restart the World Wide Web Publishing Service before your changes take effect.
- For more information on MIME types, see MIME Types in the IIS 6.0 Manager Help.
- The URLScan tool can also be configured to block processing of certain file name extensions. Verify these settings.

RELATED TOPICS

[Uploading Large Attachments](#)

CHAPTER 24 - IMPORTING DATA

ABOUT IMPORTING DATA

Vizor stores most of its issue data in the tblDts table of the issue database (<project>01.dat for Microsoft Access, <project>01_DAT for SQL Server). To import issue data into this table, use the **Import Issues** command in the **File** menu of Vizor Admin. Vizor stores most of its issue data in the tblDts and tblFixInformation tables of the issue database. tblFixInformation stores the information entered in the Fix tab (note that not all projects use the Fix tab). The Import Issues dialog shows the following information:

Source table

You import issues from a source table, which could be in Microsoft Access, SQL Server, a delimited text file, or a number of other formats.

Source fields

Source fields are the fields imported from the source table

Destination fields

Destination fields are the fields in the Vizor database table tblDts.

Mapped fields

You map source fields to destination fields. When you click Start, the mapping determines how the source fields are copied to the destination fields.

To import issues:

- 1 [Prepare your issue data and the Vizor database.](#)
- 2 [Import issue data into a source table.](#)

A source table is an intermediate table that holds the issue data you want to import into tblDts and tblFixInformation.

- 3 [Map the fields in the source table](#) to the fields in tblDts and tblFixInformation.
- 4 Click **Start** to create new records in tblDts and tblFixInformation and to copy the imported issues from the source table into these new records.

RELATED TOPICS

[What Can You Import?](#)

[How New Records are Created](#)

[Before You Import](#)

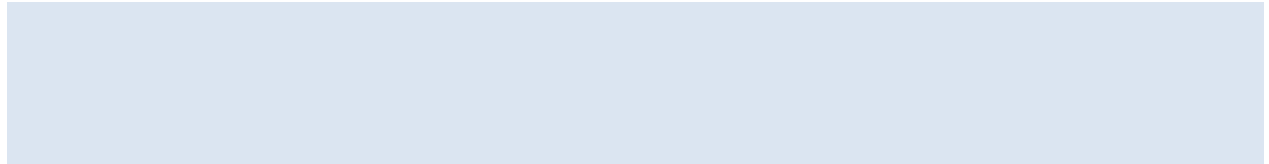
[Importing Issues into a Source Table](#)

[Combining Multiple Source Tables](#)

[Importing Issues from Projects](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)



WHAT CAN YOU IMPORT?

Vizor uses Microsoft Access functionality to import data, so you can import from any of the formats and applications supported by Microsoft Access:

Delimited text (values separated by commas, tabs, or other characters)

- Fixed-width text (values arranged so that each field has a defined width)
- Microsoft Excel (versions 2.x, 3.0, 4.0, and 5.0)
- SQL databases, using ODBC drivers
- Microsoft Access (databases other than the open database)
- HTML

With Vizor, you can import data into any of the fields in the tblDts table. This table contains most of the issue data.

You cannot use Vizor to import data into the following tables:

- tblUser, which contains user information such as the user's logon name, full name, e-mail address, and telephone number.
- tblAttachment, which contains the names of the files attached to different issues.
- Any of the tables that contain the possible values for choice lists.

HOW RECORDS ARE CREATED

When you click the Start button, Vizor creates new records in tblDts and tblFixInformation, and copies the data from the source table into the new records.

In general, if there is no mapping for a field or the imported data is invalid, Vizor leaves the field blank. However, there are exceptions. The following summarizes how Vizor handles fields when it creates new issues.

Issue Number

You cannot import issue numbers. Vizor automatically assigns issue numbers to issues.

Owner (nUserID)

You can import the full name of a user or the ID of the user in tblUsr.

If there is no mapping or the user name is not found in tblUsr, the field is left blank.

Submitter (nSubmitterID)

You can import the full name of a user or the ID of the user in tblUsr.

If there is no mapping or the user name is not found in tblUsr, the name of the current user is stored in the field.

Contact (nOriginatorID)

You can import the full name of a user or the ID of the user in tblUsr.

If there is no mapping or the user name is not found in tblUsr, the name of the current user is stored in the field.

Progress

You can import the ID of a choice value or the choice text. If there is no mapping for a choice list field, Vizor sets the **Progress** field to the choice with ID = 0 (by default, **New**).

State

You cannot import **State** values. sets the **State** field based on the **Progress** field.

Time and Date fields

You can import time and date values. If you do not map a date or time field, Vizor sets the field to the current date or time. See the tblSubstate choice table for the mapping of Progress values to **State** values.

The **Closed Date** field value is imported only if the **State** is **Closed**.

Imported date values should use the format yyyy/mm/dd.

Imported time values should use the time format specified in the Regional Options of Windows Control Panel.

Choice lists

You can import either the choice ID (the nID of the choice in the Vizor choice table) or the choice text.

RELATED TOPICS

[About Importing Issues](#)

[What Can You Import?](#)

[Before You Import](#)

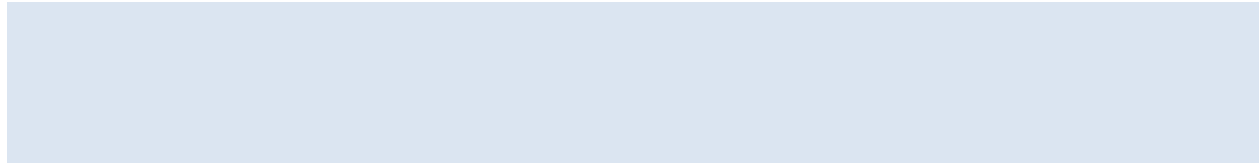
[Importing Issues into a Source Table](#)

[Combining Multiple Source Tables](#)

[Importing Issues from Projects](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)



BEFORE YOU IMPORT

ADD FIELDS FOR IMPORTED DATA

Compare the issue data you want to import against the fields in tblIDs and tblFixInformation. If you want to import additional issue data for which there are no corresponding fields in these two tables, use the Field editor to add the required fields.

CREATE A NUMBER FIELD TO HOLD IMPORTED IDS OR RECORD NUMBERS

If you want to import the IDs or record numbers of your data records, create a numeric field in Vizor. You can then map your record IDs to this field when you import your data.

CREATE USERS AND CONTACTS

Add users and contacts in Vizor before importing data that contains user names.

When you import user names into the **Owner**, **Submitter**, and **Contact** fields, the names must match the full names assigned to the users in Vizor. You must use the users' full names, not their logon names.

For example, if you import a set of issues owned by a user for whom no entry is defined, the **Owner** field is left blank.

Note that you can import the numeric ID for a user in the tblUser table (users database).

CUSTOMIZE VIZOR CHOICE LISTS TO MATCH THE IMPORTED DATA

In Vizor, customize choice lists to match the values you want to import. You could also change the values in the imported data to match the Vizor choice lists. You could do this in the original issue data, or you could use Microsoft Access to modify the source table created when you imported the original issue data.

IF YOU ARE IMPORTING TEXT FILES

If your issue data is in a text file, the first data record should contain the field names. Otherwise, Vizor assigns the fields the names 1, 2, 3, and so on. This makes it harder to map the fields.

Also, the first data record should have no blank fields. Microsoft Access uses the first data record to determine the data types of the fields. For example, if a date field is blank, Microsoft Access imports it as a text value, and you will be unable to convert the field into date format.

BACK UP PROJECTS

Make a [backup](#) copy of the issue database before you import any data. If you make a mistake mapping the fields, having a backup copy allows you to change the mapping and try again.

LOG OFF USERS

Make sure no one is using the project database while you import issues. Ask all users to exit Vizor before you start to import issues.

RELATED TOPICS

[Backing Up Projects](#)

[Logging Off Users](#)

[About Importing Issues](#)

[What Can You Import?](#)

[How New Records are Created](#)

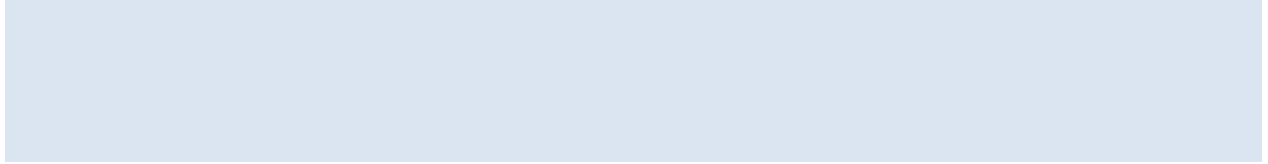
[Importing Issues into a Source Table](#)

[Combining Multiple Source Tables](#)

[Importing Issues from Projects](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)



IMPORTING ISSUES INTO A SOURCE TABLE

You don't import issues directly into the Vizor tables. Instead, you import the issues into a source table. Vizor copies the imported data into its tables after you define a mapping between the fields in the source table and the Vizor fields.

To import issues from a file:

- 1 On the **File** menu, click **Import Issues**.
- 2 Click **New Source Table** to import data from a text file, spreadsheet, or database table into a Microsoft Access table (the source table).
- 3 In the **Files of Type** list, click the file format you want to import.

To import issues from another Vizor database (or from any other Microsoft Access database), click **Microsoft Access**.

- 4 Click the file you want to import, and then click **Import**.

To import issues from a Vizor Access database, select the **project01.dat** file.

- 5 In the Import Objects dialog, select the import options you want.

If you are importing from a text file or spreadsheet, select the **First Row Contains Field Names** check box.

If you are importing from a Microsoft Access database, select the tables you want to import.

- 6 Click **OK** to import the data.

Vizor creates a source table named after the file you selected.

- 7 To import another file, repeat steps 4 and 5. When you've finished importing, click **Close**.

To import issues from a SQL database:

- 1 On the **File** menu, click **Import Issues**.
- 2 Click **New Source Table**.
- 3 In the **Files of type list**, click **ODBC**.
- 4 In the Select Data Source dialog, click the **Machine Data Source** tab.
- 5 Find the SQL Server data source and click **OK**.
- 6 In the SQL Server Login dialog, type your SQL Server login ID and password.
- 7 Click **Options**, and in the **Database** list, click a database.

To import issues from a SQL Server database, click the issue database.

8 Click **OK**.

9 In the Import Objects dialog, select that tables you want to import.

For SQL Server databases, the table to import is dbo.tbIDts.

10 Click **OK**.

RELATED TOPICS

[About Importing Issues](#)

[What Can You Import?](#)

[How New Records are Created](#)

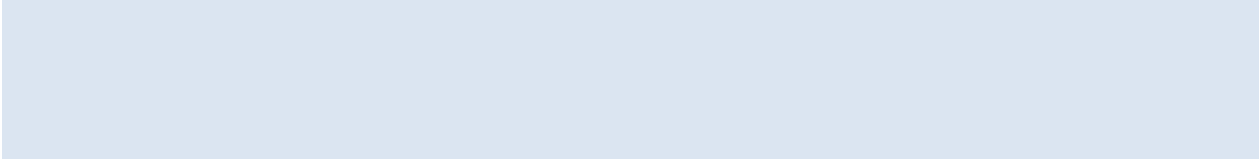
[Before You Import](#)

[Combining Multiple Source Tables](#)

[Importing Issues from Projects](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)



COMBINING MULTIPLE SOURCE TABLES

When you import issues into Vizor, you work with one source table at a time. If your issue data is stored in two or more tables, you'll need to combine those tables into a single source table before you can map the source fields to the destination fields.

First, use the Import Issues dialog to import each file into a separate source table, then use Microsoft Access to define a query that joins these source tables into a single table.

RELATED TOPICS

[About Importing Issues](#)

[What Can You Import?](#)

[How New Records are Created](#)

[Before You Import](#)

[Importing Issues into a Source Table](#)

[Importing Issues from Projects](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)

IMPORTING ISSUES FROM PROJECTS

In Vizor projects, issue data may be split across two tables (tblDts and tblFixInformation) in the issue database.

You need to merge the two tables before you can map the fields.

To import issues from Microsoft Access .dat files:

- 1 Click Add tblDts and tblFixInformation to the list of source tables.
- 2 In the **Source Table** list, click **qryMergeCensusTables**.

This query adds the combined fields from tblDts and tblFixInformation to the **Source Fields** list.

- 3 Click **Auto Map**, and then click **Start**.

To import issues from a SQL Server issue database:

- 1 Add dbo.tblDts and dbo.tblFixInformation to the list of source tables.

The source tables are named dbo_tblDts and dbo_tblFixInformation.

- 2 [Create a new query](#) to merge dbo_tblDts and dbo_tblFixInformation.
- 3 In the **Source Table** list, click the new query.

This query adds the combined fields from dbo_tblDts and dbo_tblFixInformation to the **Source Fields** list.

- 4 Click **Auto Map**, and then click **Start**.

RELATED TOPICS

[About Importing Issues](#)

[What Can You Import?](#)

[How New Records are Created](#)

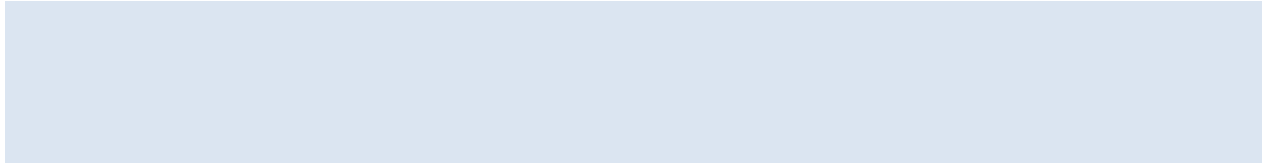
[Before You Import](#)

[Importing Issues into a Source Table](#)

[Combining Multiple Source Tables](#)

[Creating Queries to Merge Source Tables](#)

[Mapping Fields](#)



CREATING QUERIES TO MERGE SOURCE TABLES

The qryMergeCensusTables query merges two source tables named tblDts and tblFixInformation. If the source tables you want to merge have different names, you need to create a new query based on qryMergeCensusTables.

To create a new query for merging source tables:

- 1 Exit Vizor Admin and open Program Files\Vector\ConnectTools\Admin.mdb in Microsoft Access.
- 2 In the **Objects** list, click **Queries**. Click **qryMergeCensusTables** and click **Design**.
- 3 In the **View** menu, click **SQL View**.
- 4 In the SQL statements, replace all occurrences of tblDts and tblFixInformation with the names of the source tables you want to merge.
- 5 In the File menu, click **Save As** and type a name for the new query.
- 6 Close Admin.mdb.
- 7 Open Program Files\Vector\ConnectTools\Admin.mde.
- 8 Open the table tblImport and add a new record. In the tName column, type the name of the query. In the nID column, type a unique, numeric ID for the query (use the next available nID).
- 9 Close Admin.mde.
- 10 The new query is now available in the **Source Table** list of the Import Issues dialog.

RELATED TOPICS

[About Importing Issues](#)

[What Can You Import?](#)

[How New Records are Created](#)

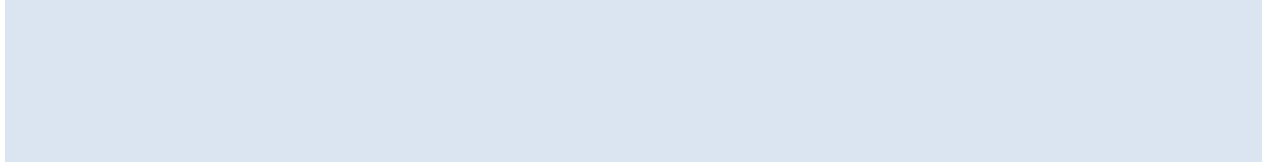
[Before You Import](#)

[Importing Issues into a Source Table](#)

[Combining Multiple Source Tables](#)

[Importing Issues from Projects](#)

[Mapping Fields](#)



MAPPING IMPORT FIELDS

After you create a source table, you must define a mapping between the fields in the source table and the fields in tblDts and tblFixInformation. Vizor uses this mapping to copy the imported data into the new records it creates in tblDts and tblFixInformation.

To map a source field to a destination field:

- 1 Click a field in the **Source Fields** list.
- 2 Click a field in the **Destination Fields** list. The data in the source field is copied to this field when you click the **Start** button.
- 3 Click the right arrow button to map the fields.

To remove a mapping:

- 1 In the **Mapped Fields** list, click a mapping.
- 2 Click the left arrow button.

MAPPING TEXT FIELDS

Text fields, such as tBriefDescription, have a limit of 255 characters. Memo fields store large numbers of characters (in Access, up to 64000 characters; in SQL Server, up to maximum length of 231-1 (2,147,483,647) characters).

MAPPING MEMO FIELDS

Import data into the mDetailedDescription and Activity\$Log fields, not into the Description\$Input and Activity fields. Import data into the mDetailedDescription and Notes\$Log fields, not into the Description\$Input and Notes\$Input fields.

IMPORTING CHOICE LISTS

Import either the choice text or the numeric ID of the choice in the choice table.

IMPORTING USERS, SUBMITTERS, AND CONTACTS

Import either the full name of a user, or the numeric ID for that user in the tblUser table (users database).

CHAPTER 25 - VIZOR LICENSES

ENTERING YOUR LICENSE INFORMATION

During installation, you must enter the license key.

To check the number of available licenses, the license keys and related information, log on to Vizor Web Admin and go to the **Licenses** tab.

Evaluation licenses allow you to use Vizor for a specified period of time. At the end of that period, you can no longer run Vizor unless you provide a valid license key.

Concurrent licenses are permanent licenses that allow a specified number of users to use Vizor simultaneously. With concurrent licensing, you only need licenses for the maximum number of simultaneous users. For example, if you have 30 users who need access to Vizor, but no more than 20 log on at the same time, you need only 20 licenses. When all 20 licenses are in use, no additional users can log on until someone logs off.

Named licenses are per-user licenses. Each license is dedicated to a specific user. Vizor automatically assigns named licenses to users as you create and disable user accounts. With named licensing, you cannot create more users than named licenses (for example, if you have five named licenses, then you can have at most five user).

ENTERING LICENSE KEYS

When you receive a new license key use the Admin tool to enter your license key.

To enter your license key:

1. Log on to **Web Admin**.
2. Go to Licenses & Logons.
3. Click View/Edit Licenses.
4. The **License Keys** list opens. Enter a new key and click Add.

NUMBER OF LICENSES

Shows how many users can run Vizor concurrently. The Access Enabled license key determines the number of concurrent users (to see the number of concurrent users, click **Access Enabled** in the **Product** list).

LICENSE TYPE

Evaluation licenses are temporary licenses that have an expiration date. **Concurrent** licenses are permanent licenses that allow a specified number of users to use Vizor simultaneously.

EXPIRATION DATE

Shows the date at which the evaluation license expire and after which users will not be able to log into VIZOR.

RESERVING LICENSES

VIZOR offers the ability to assign licenses to specific users. This way, the latter will always have access to Web views regardless of the number of users currently accessing the system.

For example, if you have 10 licenses, and you wish to reserve 1 for a specific user (For example, 'Employee'), this means you will only have 9 licenses left for other users to use. When 'Employee' logs in, Vizor will automatically use his reserved license. If 'Employee' logs in a second time, while still using his reserved license, Vizor will use one of the unreserved (floating) licenses.

To assign reserved licenses to a user, set a value for the *Number of Reserved Licenses* field. This is done via the Users and Contacts editor, as described in [Editing User Information](#).

For information on how concurrent licenses are handled, see [Types of Web Views](#).

To view how the licenses are distributed, or whether or not they are in use, go to the **Licenses** area in the Web Admin.

CHAPTER 26 - INTEGRATING REMOTE CONTROL

ABOUT REMOTE CONTROL

Remote Control enables a help desk analyst to take control of a user's computer to investigate problems and provide support from inside a Web view. To start a remote control session, the help desk analyst enters a computer name in a field and clicks the **Remote Control** button adjacent to the field.



The button connects to the specified computer and starts a PC-Duo ActiveX Remote Control session. The remote control session runs in a browser window, and does not require the PC-Duo Control to be installed on the local computer.

The requirements for remote control are:

- The Web view must be running in Internet Explorer.
The Remote Control button loads a page the contains an ActiveX control. This ActiveX control allows the help desk analyst to watch, share, or control the remote computer.
- The PC-Duo Remote Control Client must be installed on the remote computer, and the Client cannot use its license key serial number as a security key.

RELATED TOPICS

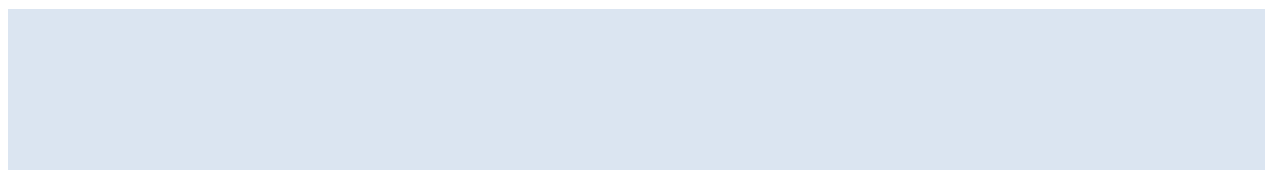
[Setting Up Remote Control](#)

[Adding Remote Control Buttons](#)

[Configuring ActiveX for Remote Control](#)

[Checking Client Security Key Settings](#)

[Removing Remote Control Buttons](#)



SETTING UP REMOTE CONTROL

To add remote control to a Web view:

- 1 Add a text field where users can enter the computer name or IP address.

This field is used to identify the PC-Duo Client on the user's computer. The Vizor project includes a Computer Name field for this purpose.

- 2 Add the Remote Control button to the field.
- 3 Generate the Web view.

RELATED TOPICS

[About Remote Control](#)

[Adding Remote Control Buttons](#)

[Configuring ActiveX for Remote Control](#)

[Checking Client Security Key Settings](#)

[Removing Remote Control Buttons](#)

ADDING REMOTE CONTROL BUTTONS

To add the Remote Control button to a field:

- 1 Log on to the Web View Editor.
- 2 In the shortcut bar, click **Remote Control** .
- 3 In the **Available Fields** list, expand the tab that contains the field.

If you do not see the field, select the **Show All** check box to show all available fields.

- 4 Select the **Can Connect** check box for the field.
- 5 Generate the Web view.

When you select the **Can Connect** check box for a field, the Web View Editor sets the **URL** and **URL Button CSS** attributes of the field.

The **URL** field attribute is set to:

```
../../RC/rcviewer.asp?Client=%fieldvalue%
```

The first part of the URL (../../RC/rcviewer.asp) is the Remote Control URL, which points to this file:

```
ConnectServer\CensusWeb\Views  
    \CensusWebVD\RC\rcviewer.asp
```

The RCviewer.asp file is installed with Vizor. You can customize this file to change the default remote control settings. For example, you can specify the security key used to access clients, and enable encryption or compression.

The second part of the URL is a query string that identifies the computer to control. When a user clicks the Remote Control button, **%fieldvalue%** is replaced with the current contents of the field.

The **URL Button CSS** field attribute specifies the CSS class used for the button.

RELATED TOPICS

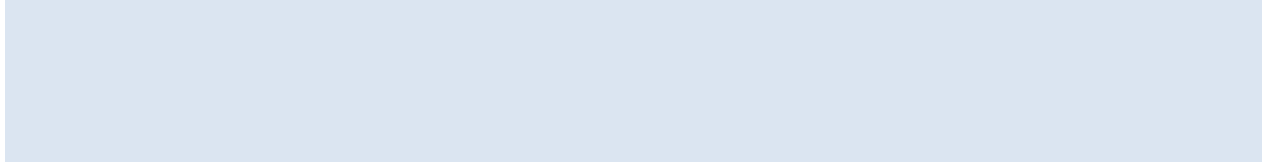
[About Remote Control](#)

[Setting Up Remote Control](#)

[Configuring ActiveX for Remote Control](#)

[Checking Client Security Key Settings](#)

[Removing Remote Control Buttons](#)



CONFIGURING ACTIVEX FOR REMOTE CONTROL

To configure PC-Duo ActiveX Remote Control, you edit the file RCviewer.asp, which is located in the folder

```
ConnectServer\CensusWeb\Views  
  \CensusWebVD\RC\RCViewer.asp
```

RCViewer.asp contains a number of variables that allow you set options and defaults.

After you customize RCViewer.asp, copy the file to

```
ConnectServer\CensusWeb\CUSTOMIZEDFILES  
  \#Project#vit\RC
```

This ensures that the next time you generate a Web view, the customized copy of RCViewer.asp is copied to the output folder.

DEFAULT_SECURITY_KEY

Specifies the security key used by PC-Duo ActiveX Remote Control when it connects to clients.

The ActiveX control does not support licensing, so you cannot type an asterisk to use the license key serial number as the security key.

DEFAULT_COMPRESS

Enables (1) and disables (0) compression. By default, compression is disabled.

DEFAULT_ENCRYPT

Enables (1) and disables (0) encryption. The encryption type is 56-bit DES encryption. By default, encryption is disabled.

DEFAULT_SCALE_TO_FIT

Specifies whether the remote screen is scaled to fit in the browse window when the remote control session starts. Set to 0 to turn off scaling.

Users can turn on scaling after the remote control session starts by selecting the **Scale to Fit** check box.

DEFAULT_SCROLL_BARS

Specifies whether the view window has scroll bars when the remote control session starts. Set to 0 to turn off scroll bars.

Users can turn on scroll bars after the remote control session starts (by selecting the Scroll Bars check box).

DEFAULT_VIEW_TYPE

Specifies the default viewing mode (Watch, Share, or Control). Users can change the viewing mode after the remote control session starts.

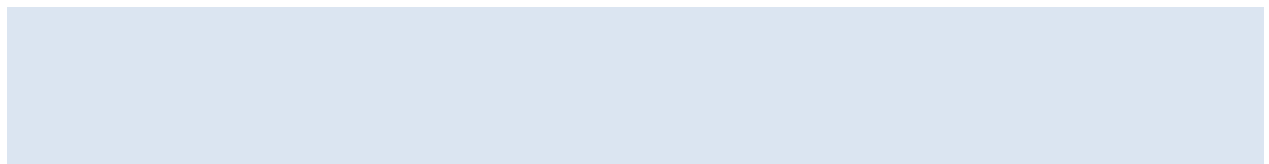
DEFAULT_INIT

Specifies the name displayed by the PC-Duo Client during a remote control session. Allows the remote users to identify who is controlling their computer. "*" means use the computer name.

RELATED TOPICS

[Setting Up Remote Control](#)

[Checking Client Security Key Settings](#)



CHECKING CLIENT SECURITY KEY SETTINGS

PC-Duo ActiveX Remote Control cannot connect to clients that use the license key serial number as a security key.

To check if a client is using the serial number as a security key:

- 1 In the **Client Configurator**, edit the **Master Profile**.
- 2 On the **Users** tab, the **Security Key** box contains the security key.

If the Security Key box contains an asterisk (*), the PC-Duo ActiveX Remote Control will be unable to connect to the client because the client is using the serial number.

RELATED TOPICS

[About Remote Control](#)

[Setting Up Remote Control](#)

[Adding Remote Control Buttons](#)

[Configuring ActiveX for Remote Control](#)

[Removing Remote Control Buttons](#)

REMOVING REMOTE CONTROL BUTTONS

To remove the Remote Control button from a field:

- 1 Log on to the Web View Editor.
- 2 In the shortcut bar, click **Remote Control** .
- 3 In the **Available Fields** list, expand the tab that contains the field.
- 4 Clear the **Can Connect** check box for the field.
- 5 Generate the Web view.

RELATED TOPICS

[About Remote Control](#)

[Setting Up Remote Control](#)

[Adding Remote Control Buttons](#)

[Configuring ActiveX for Remote Control](#)

[Checking Client Security Key Settings](#)

CHAPTER 27 - PROJECTS

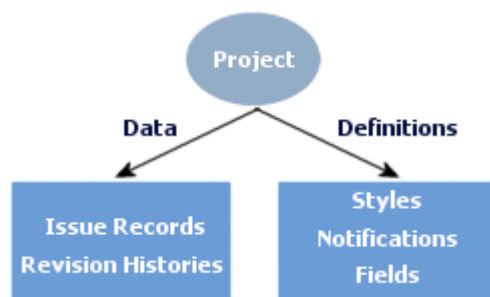
WHAT IS A PROJECT?

In Vizor, projects are used to keep issues in separate databases. How you organize your projects is up to you. You can have one project for tracking all issues, or you can have several projects. To create and manage projects, use [Vizor Admin](#).

A project includes:

- A database of issues and their revision histories, attachments, Service Level Agreements and e-mails.
- A set of project-specific definitions, which include styles (queries, sorts, layouts, reports, and report pages), the fields, e-mail templates, choice lists and notification definitions. These definitions are shared by all users of a project.

PROJECT STRUCTURE



RELATED TOPICS

[Creating New Projects](#)

[Copying Definitions and Styles](#)

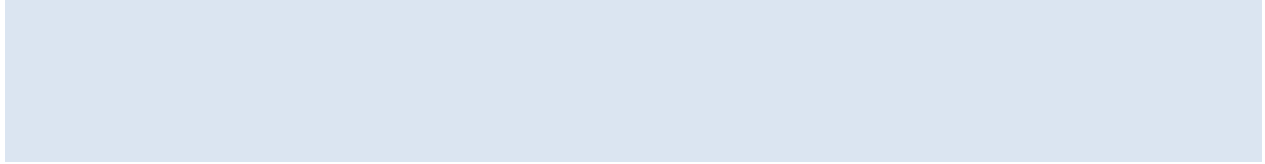
[Building a Project Template](#)

[Deleting Projects](#)

[Relocating Databases to SQL Server](#)

[Relocating Attachments](#)

[Editing Project Properties](#)



CREATING PROJECTS

Typically, you start a new project because you want to start tracking issues in a new database. However, there are other reasons for creating a new project. For example, you may want to:

- Build a template project and use it as a base for all other projects.
- Break a large project into several smaller projects.

New projects are always based on an existing project. When you create a new project, you can copy the issues, styles, notifications, Web views and the fields from the base project.

To create a project:

- 1 Start Vizor Admin, and click **New Project** in the **File** menu.
- 2 In the **Project Name** box, type the name of the new project.
- 3 In the **Base Project** list, click the project you want to use as a template for the new project.
- 4 In the **Groups Allowed to Open Project**, click the user groups that are allowed to access the project through Vizor Admin, Vizor Web Admin, the Web View Editor, or Web views.

Note that access to individual Vizor components can be selectively disabled. For example, a group can have access to Web views of a project, but no access to Vizor Admin, Vizor Web Admin, and the Web View Editor.

- 5 Under **Global Styles and Notifications**:

- Click **None** to create a project with no queries, sorts, layouts, reports, or notifications. Only the fields are copied from the base project.
- Click **Copy from the Base Project** to copy the definitions of fields, queries, sorts, layouts, reports, and notifications to the new project.

If you are creating your first project, we recommend you use one of the sample projects that come with Vizor as the base project. Copy the styles and notifications, but do not copy the issues and revision history. Also, read [Building a Project Template](#) before you start tracking issues.

- 6 If you want to copy the issues from the base project, click the **Copy issues and revision history from base project** check box.
- 7 If you don't want to copy the Web views from the base project, clear the **Copy Web views from the base project** check box.

Copying Web views from the base project can save time. For example, copying the Web views from Vizor means you don't have to manually set up timestamping in a new Web view.

RELATED TOPICS

[What is a Project?](#)

[Copying and Sharing Definitions](#)

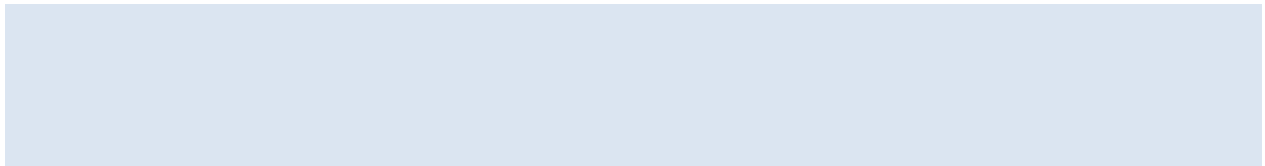
[Building a Project Template](#)

[Deleting Projects](#)

[Relocating Databases to SQL Server](#)

[Relocating Attachments](#)

[Editing Project Properties](#)



COPYING DEFINITIONS AND STYLES

A new project is always based on an existing project. The new project can start **empty** or **take a copy** of the styles, notifications, and fields in the base project. In the New Project dialog, select the option that corresponds to what you want to do with the new project:

- **None**

Clear the styles and notifications so you can build your own. Copy the fields from the base project and use it as a starting point.

- **Copy from the Base Project**

Copy the styles, notifications, and fields from the base project and use them as a starting point, or template, for the new project. This is the suggested way.

WHY COPY STYLES AND NOTIFICATIONS?

Copying styles and notifications allows you to configure your new project faster by making changes on existing styles instead of creating them from scratch. For example, a new report may be very similar to a report that exists in the base project, with the only difference being that you need to add or replace new fields that only exist in the new project. In this case, it is faster to copy the styles, and therefore the reports, from the base project and then create the new fields and update the report in the new project.

RELATED TOPICS

[Creating New Projects](#)

[Building a Project Template](#)

BUILDING A PROJECT TEMPLATE

Before you start using Vizor to track issues, you should build a project template. A project template is a project that acts as the base for all new projects. The project template contains styles and definitions common to all projects.

For example, a project template can define a standard set of queries, reports, and notifications. In addition, a project template can include a set of fields customized to match your company's process and terminology.

Using a project template allows you to reuse custom styles, notifications, and fields. Each new project can copy the styles and definitions from the project template.

Before you build a project template, you must evaluate your issue tracking and reporting requirements. For example, you should decide:

- Which custom fields and choice lists are required.
- Which standard queries and reports you want in each project.
- Which notifications are needed to the workflow in your process.

To build the actual project template, create a new project based on the Vizor project, then customize the issue fields and define the custom styles and notifications you need.

RELATED TOPICS

[What is a Project?](#)

[Creating New Projects](#)

[Copying Definitions and Styles](#)

DELETING PROJECTS

TO DELETE A PROJECT:

- 1** In Vizor Admin, click a project in the **Project** list.
- 2** On the **File** menu, click **Delete Project**.

RELATED TOPICS

[What is a Project?](#)

[Creating New Projects](#)

EDITING PROJECT PROPERTIES

The project properties are displayed when you click a project in the **Project** list.

PROJECT LOCATION

Location of the project files, which includes the definitions database.

DEFINITIONS DATABASE

The name of the project definitions database. The definitions database contains the definitions of the fields, queries, sorts, layouts, reports, and notifications for the project.

DATABASE ENGINE

Specifies whether the project uses Microsoft Access or SQL Server for the issue database.

GROUPS ALLOWED TO OPEN PROJECT

List of user groups that are allowed to open the project. Groups allowed to open a project can:

- Log on to Web views of the project (if they also have permission to open the views).
- Edit the project in Vizor Admin (depending on what features are enabled for the group).
- Edit views of that project in Web View Editor, or generate new views of that project.

If the Users group is allowed to open the project, then all groups can open the project. The Admins group is always allowed to open projects.

CURRENT USERS

Displays the users currently logged on to Web views of the project.

Name

Logon name of the user.

Date/Time

Date and time that the user logged on.

Workstation

The user's IP address or computer name.

Project

The user is logged on to a Web view of this project.

Virtual Directory

The virtual directory is the final part of the URL used to access the Web view. For example, if the virtual directory is named *vit*, then the URL is `http://server/vit`.

Virtual directories such as `connect00`, `connect01`, and `connect02` are all accessed through the `//server/vit` URL.

Application

This is always *Web View*. The **Logons Editor** also lists the Web View Editor and Web Admin.

CHAPTER 28 - PLANNING AND DEPLOYING A CUSTOM PROJECT

PLANNING

Before you can set up your Vizor system, you need to understand your business structure and have a clear idea of the process you want to implement. The following sections illustrate the sort of information required to set up a typical Vizor installation.

INFORMATION RECORDING AND TRACKING

- What information do you want to record and track for each issue?
- How do you categorize issues?
- Do you want to provide different views of the issue data?
For example, do you want a summary view optimized for logging issues and categorizing problems, and a detailed view for recording things such as how the issue was resolved, and how long it took?
- Do you want to be able to link issues? Vizor supports parent-child [relationships](#) between issues.

WORKFLOW

- What is your help desk process?
Can you represent the path of an issue through the process as a sequence of steps, or decisions, such as New, Assigned, Verified, Resolved?
- Do you want to enforce this workflow process?
- Who needs permission to make decisions? Who is responsible for carrying out each step in the process?
- Do you need e-mail notifications to enforce ownership and accountability? When issues are submitted or resolved, who needs to be alerted?

ROLES AND RESPONSIBILITIES

- What are the different roles and responsibilities of the people that use Vizor? Can you identify different groups of users?
- Do different groups have different requirements of the issue tracking system? Do you need to restrict access to the system based on group membership?
For example, you probably want to restrict most users to the ability to submit and track issues. Support staff, on the other hand, require greater access to the system.
- Do you need to manage and coordinate work teams? Vizor allows you to define work teams and the assign issues to them rather than a specific individual.

IMPLEMENTATION ROADMAP

This roadmap outlines the major steps in implementing an issue tracking system. Performing the steps in the sequence recommended here save time and simplify the process.

STEP 1. CREATE A PROJECT

A project defines the fields, queries, sorts, layouts, and reports that you see in a Web view.

- 1 Create a new project.
- 2 Customize the fields in the project.
- 3 Edit and define the queries, reports, sorts, and layouts to match the set of fields defined in the project.

STEP 2. DEFINE USER GROUPS AND CREATE USER ACCOUNTS

- 1 Define user groups and set group permissions to control access to data and administrative features.
- 2 Create user accounts so people can log on. You can either import user accounts from Active Directory (or from a Windows domain), or create user accounts in Vizor.
- 3 Assign users to groups.
- 4 Define work teams and assign users to work teams.

STEP 3. DEFINE AND GENERATE WEB VIEWS

After you generate and test your Web views, you can perform more advanced customizations, such as defining workflow and notifications.

STEP 4. DEFINE FIELD DEPENDENCIES (OPTIONAL)

Use the workflow editor in Vizor Web Admin to make the possible values in one choice list depend on the choice selected from another list.

You can apply changes made in the workflow editor without regenerating the Web view.

STEP 5. DEFINE A WORKFLOW (OPTIONAL)

Use the workflow editor in Vizor Web Admin to define workflow rules to emulate your process. For example, you can enforce a sequence of steps by controlling the possible values of the **Progress** field.

You can also define rules that set the value of a field when another field changes. For example, you can define rules that assign an owner based on the problem area:

When (Problem Area = Outlook), then set this value: (Owner = Resident Outlook Wiz).

STEP 6. DEFINE SERVICE LEVEL AGREEMENTS

A [service agreement](#) is an agreement between the help desk and users. The service agreement defines a required level of service for the users.

A service agreements can be with individual users (contacts), with all users in a department, or all users in a company.

A service level is a guarantee of a certain level of service. For example, a basic service level may provide 9-to-5 service during the regular working week, while a more comprehensive service level might provide 24 hour service, 7 days a week.

Each service level has its own hours of service, target response and closure times for issues, and escalation rules.

STEP 7. IMPORT EXISTING DATA (OPTIONAL)

After you finalize the set of fields in a project, you can import information from existing databases.

STEP 8. DEFINE AND SET UP NOTIFICATIONS

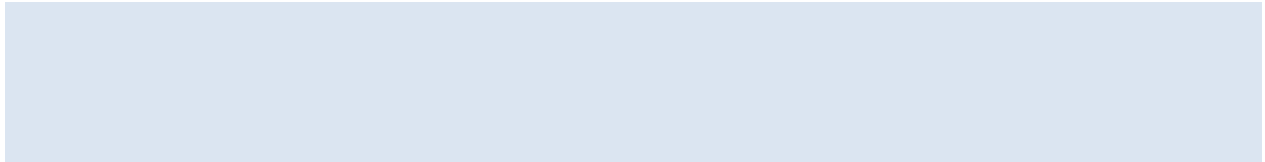
You can define and set up notifications after you generate your Web views, but you should do this before people start using the Web views to submit issues.

STEP 9. GO LIVE

Make your Web views available for general use.

STEP 10. SYSTEM MAINTENANCE (ONGOING)

You should regularly backup your projects, databases, Web views, and other system files.



CREATING A PROJECT

A project defines everything you see in a Web view: the fields, queries, reports, sorts, and layouts.

HOW MANY PROJECTS DO YOU NEED?

Each project has its own issue database.

For example, you can record different types of issue in separate databases by creating separate projects. Separate projects mean smaller databases and better performance. They also mean you can customize the fields, queries, sorts, layouts, and reports, so you can simplify your Web Views.

However, tracking issues in separate projects has some disadvantages:

- You cannot generate reports across all projects.
- You cannot create a single view that covers all projects.
- You cannot link items from different projects
- People who work with all three types of issue can find that separate projects are inefficient and time-consuming.

CHOOSING A BASE PROJECT

A base project serves as a starting point for a new project. The new project inherits the fields defined in the base project, and can optionally inherit the styles (queries, reports, sorts, layouts, and notifications) and Web views as well.

Vizor includes a default Vizor project you can use as the base project. Before you create a new project based on Vizor, you should familiarize yourself with the fields included in Vizor, and understand what you can and cannot do when customizing fields.

Designed for use in development projects, Vizor allows you to record and track project-related tasks, feature and enhancement requests, user suggestions, and change notices.

Vizor includes over 60 fields, along with a complete set of reports, queries, sorts, layouts, and notifications based on those fields. Some of the default fields cannot be deleted, so even if you don't use these fields, they take up extra space in your database.

Before you delete a field, you must remove it from any report, query, sort, layout, or notification that references it. Deleting unused fields helps minimize the size of your database.

CUSTOMIZING FIELDS

You can edit and delete most fields in Vizor. For example, if a field is the right data type and size, but has the wrong caption, you can change the caption. Similarly, you can replace some or all of the entries in choice lists.

Before customizing a project, you should familiarize with the fields, and determine:

- Which of the existing fields you can use as is, or by changing the label, choice table, or tab
- Which fields you need to add
- Which the existing fields you want to delete

You should also familiarize yourself with what you can and cannot do when editing fields.

YOU CAN:

- Add new fields.
- Change field labels.
- Change the list of choices for a choice list and update the queries that test the choice values.
If you change the **Progress** choice list, you should either disable or update the default workflow rules because the default workflow is based on this list.
- Delete most fields.
To delete a field, you must first remove it from any queries, sorts, layouts, reports, and notification that use the field.
Before you delete a field, go through the different queries, sorts, layouts, reports, and notification conditions and either remove the field or delete the style.
- Create linked choice lists so that the contents of one list depends on a selection in another list.
- Move fields between tabs.
- Change tab names, and reorder the tabs.
- Disable fields (make read-only) either for all users or for specific user groups.
- Hide fields. Note that when you hide a field, it is hidden in all Web views.

YOU CANNOT:

- Change field types.
For example, you cannot change a single choice field into a multi-choice field, or a Number field into a Text field.

- Change the field size.
For example, you cannot change the size of a text box from 20 to 30.
- Delete certain fields. Some fields, such as **Owner**, **State**, and **Progress**, cannot be deleted.
When you generate a Web view, you can choose not to include specific fields. So even if you cannot delete them, you can remove them from all views. Of course, the fields still take up space in your database.
- Use multi-choice lists in workflow rules.
- Use multi-choice lists in formulas, charts, or cross-tabs when building custom reports with Crystal Reports.

NOTES

- If you need to change the type or the size of a field, the best thing to do is to copy the field and delete the original. When you copy a field, you can edit its type and size. Note that copying a field does not copy the stored data, and deleting a field destroys any stored data.

WHAT HAPPENS TO STYLES

Styles such as queries and reports are based on fields. For the most part, Vizor Admin automatically updates styles when you customize the fields, but there are some exceptions.

When you add a new field, you need to add it to the following reports:

- Current Issue - Detailed
Used to print the current issue in a Web view.
- Notification - Detailed Record, Notification - Summary
Used to format the contents or attachments to e-mail notification messages.

NOTES

- In a Web view, these reports show only the fields exported to the view.

WHEN YOU RENAME A FIELD

When you change the label appearing on choice lists, Vizor Admin automatically updates any reports and layouts that use the field name as a title.

If you change just the Field Caption label, Vizor Admin does not update any reports and layouts that use the field name as a title.

If you rename a field such as the **Progress** field, you may want to rename any sorts, queries, and layouts whose names are based on the name of the field.

WHEN YOU EDIT A CHOICE LIST

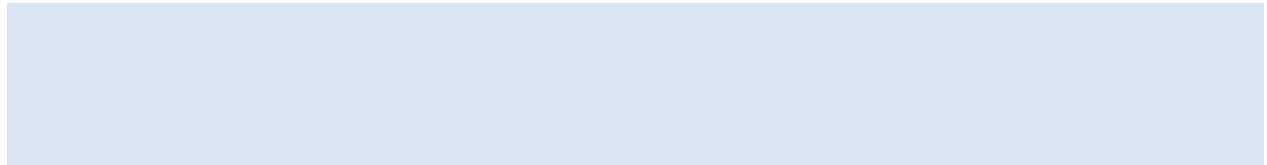
If you delete a choice from a choice table (for example, **Assigned** from **tblSubstate**, the **Progress** choice table), then you have to update any queries that test the choice value. The same is true if you change the choice text (for example, from Assigned to Started).

You may also have to update custom reports that use the choice list. For example, when you change the choice text, custom reports that use a specified sort order add the new choice to the Others category.

You should also check the workflow rules if you edit the **Progress** choice list.

WHEN YOU DELETE A FIELD

Custom reports that use the deleted field do not display properly. For example, if a custom report uses the field for calculations, or as a chart axis, then deleting the field invalidates the report. In many cases the report still works, but the data does not make sense.



CREATING GROUPS AND GRANTING PERMISSIONS

Groups enable you to restrict access to Web views or administrative features. Typically, groups match the different roles in a process. For example, if you have separate groups for employees and help desk staff, you can prevent employees from opening the same Web views used by the help desk staff.

Vizor includes several built-in groups that cannot be deleted: **Users**, **Admins**, and **Guests**. It also includes several sample groups that reflect basic help desk roles: **Employees**, **Vizor-Analysts**, and **Vizor-GroupLeads**.

If a help desk analyst is responsible for administering the your system, you can make the analyst a member of both **Vizor-Analysts** and **Admins**. As a member of **Admins**, the analyst has full access to all administrative features.

If you want to restrict access to Web views or administrative features, use groups. Typically, you create groups that match the different roles in your business process. For example, if you have separate groups for sales and development staff, you can prevent the sales staff from opening the same Web views used by the development staff.

Vizor includes several built-in groups that cannot be deleted: **Users**, **Admins**, and **Guests**. It also includes several sample groups that reflect the basic roles in a product development process: **Developers**, **QA**, **Managers**, and **Customers**.

If a developer is responsible for administering your system, you can make the developer a member of both **Developers** and **Admins**. As a member of **Admins**, the developer has full access to all administrative features.

You can have several different admin-type groups, with each group providing a different level of access to administrative capabilities. For example, a basic admin group could allow members to create new user accounts and update contact information, while other admin groups could:

- Create fields, edit choice lists, rename tabs, change workflow.
- Create reports, queries, sorts, and layouts, define notifications.
- Create projects, delete records, perform database maintenance, set up notifications.

THE BUILT-IN GROUPS

USERS

All users are automatically members of the **Users** group. You cannot remove users from this group, or delete it. Permissions granted to this group apply to all users, and cannot be overridden.

ADMINS

Members of the **Admin** group have complete access to all projects and administrative features. You cannot disable features or projects for the **Admins** group.

GUESTS

The **Guests** group is intended for users who are not full-time employees, such as customers or visitors from other companies.

FEATURES

Features allow you to control:

- Who can use the Ad-hoc Query Editor, generate reports, update contact information, view revision histories, or change their password in a Web view.
- Who can log on to Vizor Admin, Vizor Web Admin, and the Web View Editor.
- Who can use certain tools in Vizor Admin and Vizor Web Admin. For example, who can edit workflow rules in Vizor Web Admin, and who can create reports or edit fields in Vizor Admin.

PROJECTS

When you create a project, you specify which groups can open the project. Groups allowed to open a project can:

- Log on to Web views of the project (if they also have permission to open the view).

- Edit the project in Vizor Admin.
- Edit views of that project in Web View Editor, or generate new views of that project.

Admins can always open a project. Instead of allowing **Users** to open a project, you should allow specific groups. That way you can grant access permissions to views on a group-by-group basis in the Web View Editor.

VIEWS

Groups allowed to open a view can log on to the view. A group must be allowed to open the project before it can be allowed to open a view.

Remember that if members of the Users group are allowed to open the view, then any user can open the view.

USER ACCOUNTS

User accounts provide basic logon security. Vizor supports the following types of account:

- Vizor user accounts.
- Active Directory user accounts.
- Windows user accounts.

The authentication method used for each user is specified in Vizor Web Admin.

A user account must be required to access Vizor Admin, Vizor Web Admin, and the Web View Editor. Users must also log on to access Web Views unless they use a submit-only view.

SUBMIT-ONLY VIEWS

A submit-only view allows an unlimited number of users to enter issues from a single user account. All users automatically log on to the view with this account (in fact, the users never see the logon window, they go straight to the view).

This user account must belong to a group that has permission to open the Web view, and to add and update contacts. The account is used to set the **Submitter** field.

Users submitting issues by this method are recorded as the contact. They must enter the required contact information (name, e-mail, and so on) the first time they submit an issue.

IMPORTING USER ACCOUNTS FROM WINDOWS

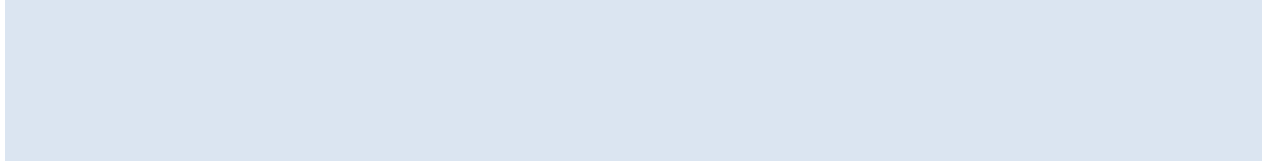
You can import accounts from an Active Directory domain or from a Windows domain. Vizor includes a User Account Manager wizard to help you import user accounts.

You can also schedule imports to keep Vizor synchronized with Active Directory (or Windows domains).

PREDEFINED USER ACCOUNTS

Vizor includes several built-in users: **demo**, **guest**, and **admin**. It also includes sample users that reflect basic help desk roles. You can disable most of these accounts in Vizor Web Admin. The **admin** account cannot be disabled, so you should change the admin password.

Disabled user accounts are still listed in the **Contact**, **Owner**, and **Submitter** lists. You must edit the project database files to delete the users from the database, or to remove them from the choice lists.



GENERATING WEB VIEWS

PARENT-CHILD RELATIONSHIPS BETWEEN ISSUES?

You can add a **Child Issues** tab to a Web view to allow users to link one or more [child issues](#) to a parent issue. For example:

- If you have several bugs that are all symptoms of the same problem, you can make those bugs the children of the main, parent bug for the problem.
- If a task consists of a number of sub-tasks, the sub-task issues can be children of the main task.
- If a bug is present in multiple branches of your code, you can create a child issue for each code branch.

By default, the parent issue controls the substate of the child issues. For example, when a user changes the substate of the parent to **Fixed**, the children are also marked **Fixed**. If you want the substate of a child issue to be independent from the substate of the parent issue, you must export the [Substate controlled by parent](#) field.

CUSTOMIZED FILES FROM THE BASE PROJECT?

New projects automatically inherit the customized Web view files of the base project including any custom reports.

When you create a new project, everything in:

```
CUSTOMIZEDFILES\#Project#<base-project>\
```

is copied to:

```
CUSTOMIZEDFILES\#Project#<your-project>\
```

WHAT IF YOU DON'T EXPORT A FIELD?

QUERIES

If you don't export a field, don't export queries that reference the field. The query won't work.

For example, suppose you don't export the **Owner** field, but you do export a query like **My Open Issues**, which tests the value of the **Owner** field. When a user tries to run the query, the user gets a message saying that the query could not be completed and the view is rolling back to the previous query.

If the query is the default query for the view, users get a more informative message when they log on.

LAYOUTS

Layouts do not include fields that are not exported.

SORTS

Sorts default to sorting by issue number if a field is missing.

LISTING REPORTS

Listing reports (such as the **Current Issue - Detailed** report, which is used by the **Print** button) include only the fields exported to the view.

CUSTOM REPORTS

Like listing reports, custom reports include only the fields exported to the view. However, unlike listing reports, custom reports do not always handle missing fields properly.

For example, if the **Owner** field is not exported, then, instead of showing owner names along one axis of a chart, the report may show the summary descriptions or the priority values. What the report shows

depends on the order of the fields in the report definition (if a field is missing, the next one in the definition is used).

SHOULD YOU AUTOMATICALLY UPDATE CHOICE LISTS?

If you expect to frequently update a choice list, set the **Automatically Update List** attribute to **Yes**. This means you won't have to regenerate the Web view every time you update the list because the view checks for updates each time a user logs on.

By default, only the **Contact**, **Owner**, and **Submitter** choice lists are automatically updated.

Automatically updating a large number of choice lists may affect performance. You should automatically update only the choice lists that you plan to frequently change.

CHANGING FIELD LABELS

To change field labels, set the **Caption** attribute in the Web View Editor. This allows you to override the label specified in the project.

The **Caption** attribute only changes the label in the HTML form. Everywhere else in the Web view (such as in reports and the Ad-hoc Query Editor), the label defined in the project is used.

WHEN DO YOU HAVE TO REGENERATE?

When you:

- Add a new field or query to a project. Remember to move the new field or query to the **Export to View** list before you generate.
- Change a choice list (for example, add a new choice, or change the text of a choice), unless **Automatically Update List** is set to **Yes**.
- When you change any of the attributes for a field (such as when you set **Automatically Update List** to **Yes**).

You don't have to regenerate when you:

- Change the definition of a query, sort, or layout.
- Define new sorts and layouts. All sorts and layouts are automatically exported to views of the project.
- Create or edit reports.

- Add a group to, or remove from, the list of groups allowed to open the view. You just have to save the view.
- Enable or disable features.

NOTES

Web view users must exit and log back on to see the changes.

HOW TO CONTROL THE LAYOUT OF FIELDS

The order of the fields in the **Export to View** list determines the order in which they are laid out in the two columns of the HTML form. For example, here's how the first six fields in the **Export to View** list would be arranged, if:

- None of the first five fields are Memo fields.
- Field 3 has **Column Span** = 2.
- Field 6 is a Memo field and has **Column Span** = 2 and **CSS Class** = **MemoFieldWidth**. (Memo fields automatically have **Column Span** set to 2 and **CSS Class** = **MemoFieldWidth**.)

1	2
3	
4	5
6	

In this example, fields 1 through 5 have the CSS Class set to either **ComboBoxWidth** or **TextBoxWidth**. You can change the default widths by editing the CSS classes in CensusMain.css.

SETTING UP NOTIFICATIONS

Notifications allow you to alert users when issues are updated. For example, you can:

- Notify analysts and developers when issues are assigned to them.
- Notify support staff when employees update an issue.
- Notify employees and customers when issues are resolved.

Use notifications to alert users to changes to critical fields such as **Owner**, **Progress**, and **Priority**. Defining too many update notifications can overwhelm users with e-mail, resulting in users missing important notifications.

CHAPTER 29 - CUSTOMIZING VIZOR

POWER CUSTOMIZATIONS

This section covers a variety of ways to customize VIZOR, all of which are considered power customizations.

Power customizations often require directly accessing the SQL Server databases and a deeper technical expertise. Some customizations require writing code scripts in languages such as Javascript and Visual Basic, as well as other advanced techniques.

ABOUT THE DATABASES

ISSUE DATABASE

Stores the issues and their revision histories.

PROJECT DEFINITIONS DATABASE

Stores all project-related definitions, including fields, queries, sorts, layouts, reports, and notifications.

USERS DATABASE

Stores user accounts, contacts, and the information entered on the **Contact** tab.

Database	Jet/Access	SQL Server
Issue	<project>01.dat	<project>01_DAT

Project definitions	<project>02.def	<project>02_DEF
Users	users.mdb	USERS_MDB

ADVANCED WEB VIEW SETTINGS

There are many advanced settings that can be customized per Web view. To change them, you typically need to customize the file:

ViewConstants.js

Available Settings

Area	Setting	Description
CommandBar		
	EnableSaveOnlyLastTab	If true, disables the OK (Save) button until the user is in the last tab. Used for Wizards. Default = false
	EnableApplyInNewRec	If true, the Apply button is visible even in forms for creating NEW items, such as "Submit Ticket" in HelpDesk. Default = false

LOGON PAGE

SIMPLIFY THE URL

If your VIZOR is hosted by , this has been already done for you.

If you host your own VIZOR server and the Web server does not run other pages on the main Web site, you can automatically redirect users from the main Web site to the VIZOR URL:

So, if the URL is:

`http://myVizorServer`

if will be redirected to:

`http://myVizorServer/connect`

In order to do that:

1. Copy the file **Default.aspx** from the folder <Server>\CensusWeb\Templates\Project and paste it in the folder of the main web site, which is typically "C:\inetpub\wwwroot".
2. Delete or rename the existing file iisstart, so it is not the default document.

IF NOT USING INTEGRATED AUTHENTICATION - SSO

Are your users never going to log on to VIZOR with their Windows or Network/Active Directory accounts integrated with the Web browser? This is also referred to as SSO (Single-Sign On).

To remove the link to log on so your users are not confused:

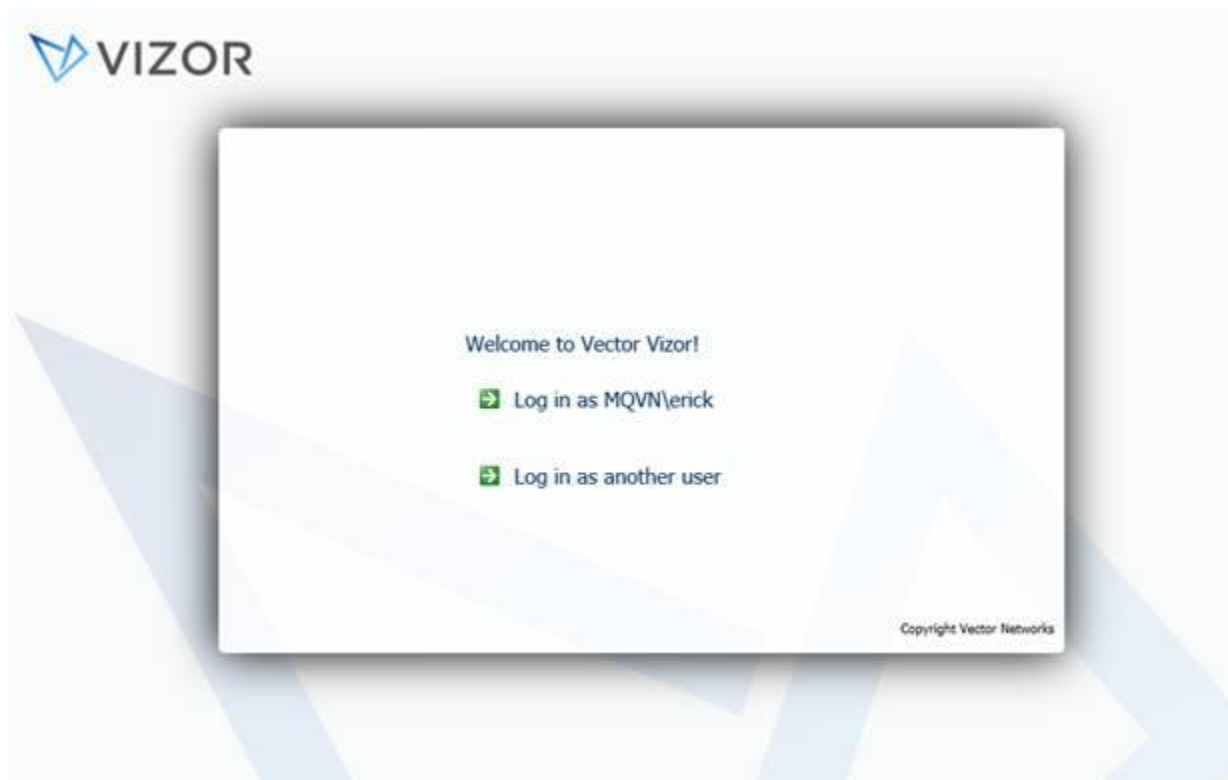
1. Go to VIZOR Admin > Tools > Options

2. Look for the option **Show "Logon with my network account..."**.
3. Select **No**.
4. Click OK on the editor.
5. Regenerate one Web view (any in that URL).

IF MOSTLY USING INTEGRATED AUTHENTICATION

Do most of your users use Windows authentication integrated with the browser?

Would you prefer having a logon page that favours those users?



1. Customize the web.config file located in "ConnectServer\CensusWeb\Templates\Project" and move it to your CustomizedFiles.

Change the default logon page to be "logoff.asp". Add the following line before the entry for "logon.asp":

```
<add value="logoff.asp" />
```

It should look like this:

```
<defaultDocument enabled="true">

  <files>

    <clear />

    <add value="logoff.asp" />

    <add value="logon.asp" />
```

2. Save the file.
3. Regenerate one Web View (any in that URL).

TOTAL EFFORT TIME

The Total Time or Total Effort time field is used to keep track of all the effective time the issue has taken from all the people involved in attending it or resolving it.

This includes the Owner of the ticket and any other user that has assigned tasks or that works on the ticket, usually field engineers, technicians and other support team members.

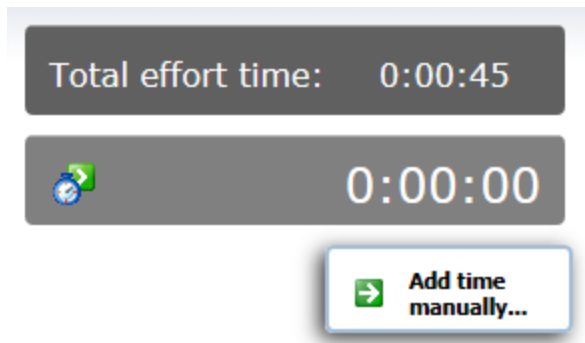
The Total Effort Time Field is characterized by:

Data Type: numeric = 102

Container Type=30

The special characteristic in this case is the container type **30**, which includes the following elements:

- Read-only Field to display the total effort time in a dd:hh:mm format.
- A Stop Watch for counting active working time as it occurs, adding the time to the total when stopped.
- A button to add any time manually, in minutes.

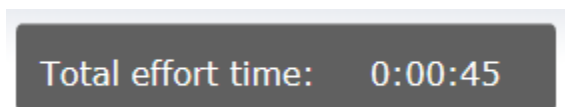


When a field is Type=102 and ContainerType=30, the HTML field template fld_102_30.html is applied.

DISPLAYING THE TOTAL EFFORT TIME IN ANOTHER TAB

To display the field in exactly the same manner as the original, with the 3 components, simply follow the usual procedure to copy a field to another tab. Since the data type and the container type will be 102_30, the correct field template will be used and the intended UI displayed.

To display only the amount in another tab, in read-only mode:



- Copy the field for another tab as usual (like the previous option)
- Make the field read-only, so fEnabled=**False**.
- Change the Container Type to **31**.

Setting the Container Type to 31 will make it use the HTML field template for TimeSpan fields **fld_102_31.html**.

The TimeSpan represents a time interval. Its two typical uses are:

- Reflecting the time interval between two date and time values. For example, to store the time between the creation time and closure time.
- Measuring elapsed time. For example, for the Stopwatch or the total effort time to reflect the time interval that has elapsed since the start of something an event.

CREATING AN ATTACHMENTS FIELD

TO CREATE AN ATTACHMENTS FIELD:

- 1 Open the project definitions database.
- 2 Open the tblDtsFields table.
- 3 Copy the record for the **Attachments** field and paste it into a new record.
- 4 Change the nID, tCaption, tLabelCaption, tBoundControlName and tWhere fields.
- 5 If you want to put the new attachments field on a different tab, change the **nDestinationTabID** field.

tblDtsFields field	Description
nID	Unique ID for the field.
tCaption	Label used in choice lists.
tLabelCaption	Default label for the field in a Web view.
tBound ControlName	Internal name for the field.
tWhere	Run the query: SELECT tWhere FROM tblDtsFields WHERE tName='tAttachments' AND tTableName='tblAttachments' ORDER BY tWhere DESC

Add 1 to the highest number returned by the above query, and enter this as the tWhere.

CUSTOMIZING URL BUTTONS

URL buttons are implemented as A elements with background images.

Each of the default URL buttons (URLButton  , MailToButton  , RemoteControl ) has four CSS classes, along with four associated :hover classes. The Web View Editor automatically determines which CSS class to use based on the field type and the field attribute settings. For example, the CSS classes used for URLButton are determined as follows:

When	These CSS classes are used
Column Span=1	URLButtonState1 URLButtonState1: hover
Column Span=2 CSS Class = MemoFieldWidth	URLButtonSpanTwoState1 URLButtonSpanTwoState1: hover
Column Span=2 CSS Class = SpanTwoWidthOne	URLButtonTwoWidthOneState1 URLButtonSpanTwoWidthOneState1: hover
Field already has a button (for example, a date field).	URLButtonSecondButtonState1 URLButtonSecondButtonState1: hover

These CSS classes are defined in the CensusMain.css style sheet in CensusWeb\Views\CensusWebVD\Theme\css.

Each different pair of CSS classes positions the button differently, by changing the value of the CSS **left** property.

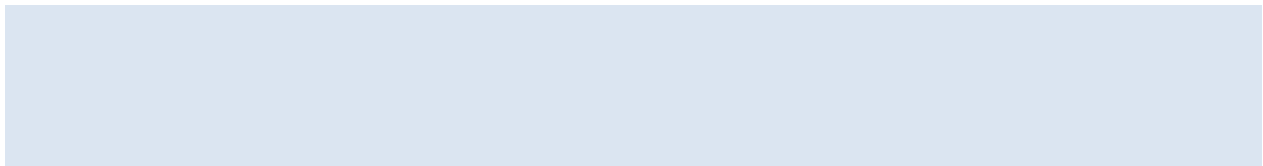
The `:hover` pseudo classes change the background image to give a rollover effect.

To define your own buttons, copy the `URLButton` classes, change the class names, and point the `background-image` property to your button images. For example:

```
.MyButtonState1
{
    background-image:
        url(../images/buttons/mybutton_off.gif);
    background-repeat:no-repeat;
    display:block;
    position:absolute;
    left:80%;
    width:24px;
    height:24px;
    top:0px;
}
```

For button rollovers, you'll need two versions of your button image. Your button images should have a one or two pixel margin (so there's some space between the field and the button). Alternatively, you could adjust the **left** property.

To use these new classes, use `MyButton` as the value of the **URL Button CSS** attribute.



CUSTOMIZING STANDARD HTML REPORTS

Web views use HTML to display Listing reports. You can customize this HTML by editing the XML files that define the HTML page template.

- **TmplRpt_listing_<type>.xml**
HTML template for an HTML report page, including the table that contains the report data.
- **ValueCell_listing_<type>.xml**
HTML template for a table cell that displays a field value.
- **CaptionCell_listing_<type>.xml**
HTML template for a table cell that displays a field caption.

where **<type>** is either tabular or multi-column (the two types of listing reports).

You can find these files in the **Reports** folder of a Web view, for example:

```
C:\Program Files\Vector
  \ConnectServer\CensusWeb
    \Views\CensusWebVD\vit_vit\Reports
```

These XML files are dynamically created the first time you view a report.

Vizor generates the HTML reports by transforming raw XML data into HTML using an XSL style sheet. The three .xml files are templates for the XSL style sheet.

CHANGING THE CSS STYLES

The CSS styles used by the HTML reports are defined in the `TmplRpt_listing_<type>.xml` file.

Notes

Reports are used to format attachments to e-mail notifications. Using an embedded style sheet ensures that notification attachments are formatted. If you want to use an external style sheet, you must use an absolute URL to locate the style sheet.

PLACEHOLDERS

The empty elements, such as `<ReportHeader/>` and `<TableCells/>`, in `TmplRpt_listing_<type>.xml` are placeholders. When Vizor generates a report, it replaces these placeholders with report-specific data.

- `<ColumnHeaders/>` is a placeholder for the field titles.
The file `CaptionCell_listing_<type>.xml` contains the HTML template for a field title, where `<CellWidth />` and `<CaptionCell />` are the width and title from the Report Editor.
- `<TableCells/>` is a placeholder for the field values.
The file `ValueCell_listing_<type>.xml` contains the HTML template for a field value.
- `<ReportCaption />` is the name given to the report in the Report Editor.
- `<ReportHeader />` is the title text from the Page Editor. `<ReportHeaderCSS />` is the formatting applied to the title text.
- `<PageHeader />` is the header text from the Page Editor. `<PageHeaderCSS />` is the formatting applied to the header text.
- `<PageFooter />` is the footer text from the Page Editor. `<PageFooterCSS />` is the formatting applied to the footer text.

NOTES

To see the generated HTML, open a report in a Web view and then view the page source.

Deleting Users

Disabled users are not deleted from the database because existing issues may refer to those users. If you don't want to see disabled users in the **Contact**, **Owner**, and **Submitter** fields:

- Delete the user from the users database.
- Remove disabled users from the **Contact**, **Owner**, and **Submitter** choice lists.

However, if an existing issue references the user in its **Owner**, **Submitter**, or **Contact** field, the field will be empty.

If existing issues reference disabled users, it may be impractical to delete the user accounts for ex-employees or contractors, because you want to keep a historical record of who reported, submitted, and resolved issues. In that case, you can prevent these users from being 'selectable' by modifying the WHERE condition of the choice list of the field in question to exclude disabled accounts.

If existing issues do reference the user, you can always reassign open issues, but closed issues can be a problem. You may want to keep a historical record of who reported, submitted, and resolved an issue. If so, you have to keep the disabled users.

REMOVING DISABLED USERS FROM CHOICE LISTS

Instead of deleting users from the users database, you can prevent the **Owner** and **Contact** lists from showing disabled users. However, if an existing issue references the user in its **Owner** or **Contact** field, the field will be empty.

To remove disabled users from choice lists:

- 1 Open the project definitions database.
- 2 Open **tblDtsFields** and find the owner (nID = 11) and contact (nID = 13) records.
- 3 In the **tWhere** field, add **And fDeleted = 0** to the **Where** clause. For example:

```
Where (([nID] >= 0 And [tName] <> '') Or [tName] = '<User>') And fOriginator = 0 And fDeleted = 0
```


ADDING A CHOICE LIST OF USERS OR CONTACTS

To add a choice list of users:

- 1 In the Field Editor, create a new field.
- 2 In the **Type** list, click **Multi-Choice List** or **Single Choice List**.
- 3 In the **Table Containing Choices** list, click **tblUser**.

tblUser cannot be edited in the **Choices Editor**.

4 In the tWhere field, type:

```
Where ([nID] >= 0 And [tName] <> '') Or  
[tName] = '<User>') And fOriginator = 0
```

The [nID] >= 0 condition removes the macros defined in **tblUser** (the users database):

User ID	User Name
-4	<Contact>
-3	<User>
-2	<Previous Owner>
-1	<Owner>

The user with nID = 0 is the **<None>** macro.

The fOriginator = 0 condition removes contacts, so that the list contains users only.

5. Apply the changes.

OPTIONAL BEHVAIOURS IN HELPDESK

HelpDesk has some special behaviours or business rules that can be enabled easily.

For example, you can:

- Filter Owners base on selected Work Team
- Filter Owners base on Category, so the skills of the person are matched
- Set the Work Team base on selected Owners

ENABLING TIMESTAMPING IN NEW PROJECTS

When you create a new project based on the Vizor project, the new project inherits the original timestamping fields and new Web views are created based on the Web views of the original project.

When a new Web view is copied from an existing one, timestamping fields retain their behaviour. However, if a new, blank Web view is created by clicking **New View** , then additional steps need to be taken to enable timestamping.

To make timestamping work in Web views of the new project, you must update the **Html code after control** attribute of the **Description**, **Description Log**, **Activity**, and **Activity Log** fields.

You need to set the **Html code after control** attribute of the timestamping fields. The required HTML is included below. Note that it is formatted for readability—the attribute itself takes a single line of text.

Description

```
<script type='text/javascript'>
    document.write(
        parent.objCustomCode.getCodeAfterField(
            '%fieldname%',
            'mem_29_txtDetailedDescription' )
    )
</script>
```

Description Log

```
<script type='text/javascript'>
document.write(
    parent.objCustomCode.getCodeAfterField(
        '%fieldname%',
        '' )
    )
```

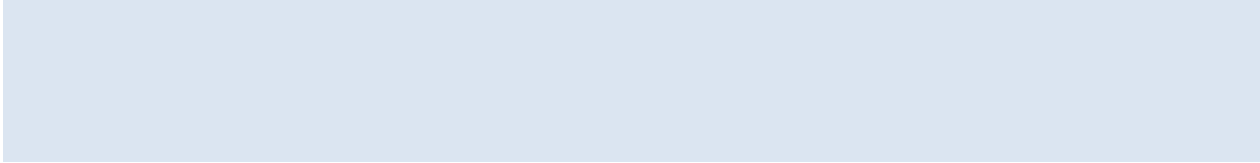
```
)  
</script>
```

ACTIVITY

```
<script type='text/javascript'>  
document.write(  
    parent.objCustomCode.getCodeAfterField(  
        '%fieldname%',  
        'mem_56_mem_56_ActivityLog' )  
    )  
</script>
```

Activity Log

```
<script type='text/javascript'>  
document.write(  
    parent.objCustomCode.getCodeAfterField(  
        '%fieldname%',  
        '' )  
    )  
</script>
```



ADDING TIMESTAMPING

Changes to a memo field can be time-stamped. For example, you can time-stamp the changes, comments, notes, and follow-ups that are added to an issue description over time.

Timestamping requires two memo fields, one for input and one for storing the timestamped log. When an issue is saved, the Web view automatically copies the contents of the input field to the log field.

1 Use the Field Editor to create a memo field for input of new data, and a field for displaying the timestamped log. For the input field, clear the **Show in Choice Lists** and **Maintain Revision History** check boxes.

2 Find the internal name of the timestamp log field. The internal name looks like *mem_58_mem_58_FieldName*.

If you have Microsoft Access, you can find the internal name by opening the project definitions database and looking in the `tblDtsFields` table. The field name is composed from the values in the **nID** and **tBoundControlName** fields:

```
mem_<nID>_<tBoundControlName>
```

The **tBoundControlName** field includes a repeat of the *mem_<nID>* string. That is why the string appears twice in the name.

If you don't have a copy of Microsoft Access, generate a Web view for the project, then get the value of the **name** attribute of the text area control associated with the timestamp log field. You can find this in the **tmplRecord.html** file located in the HTML folder of the view:

```
CensusWebVD\<project>_<view>\HTML
```

3 Set the HTML code after control attributes:

Before you generate a Web view, you must export the two fields and set their HTML code after control attributes.

a For the input field, enter this HTML code (as a single line of text):

```
<script type='text/javascript'>document.write
( parent.objCustomCode.getCodeAfterField( '%fieldname%', '<timestamp-log-field-name>' ) )
</script>
```

<timestamp-log-field-name> is the field name you found in the previous step. Note that the javascript is case sensitive.

b For the timestamp log field, enter this HTML code (as a single line of text):

```
<script type='text/javascript'>document.write
( parent.objCustomCode.getCodeAfterField( '%fieldname%', '' ) ) </script>
```

CUSTOMIZING TIMESTAMPING

By default, each time a new entry is added to the timestamp log, a line that looks like this is added:

```
December 11, 2001 4:38 PM    analyst    (New)
```

You can customize the way the timestamp log is formatted by editing **MoveControlData()** in **Js\CustomCode.js**. For example, you can insert a line of dashes to separate entries, add tabs (\t), or add more line breaks (\r):

```
objTargetControl.value="-----\r" +  
  
                                                                getDate() + "\r" +  
  
                                                                objSourceControl.value +  
  
"\r\r"  
  
                                                                objTargetControl.value +  
  
"\r";
```

where:

getDate() returns a string that includes the date, time, name of the current user, and current value of the Progress field.

objSourceControl.value is the new input to add to the timestamp log

objTargetControl.value is the content of the timestamp log.

TIMESTAMPING WITH ONE MEMO FIELD

You can implement timestamping with just one memo field. To do this, use the [HTML code before control](#) attribute to insert a text area control above the timestamp log field. This text area control is not tied to any database field. It exists only to allow input, which is copied into the timestamp log field when you save the issue. The advantage to this approach is that you avoid storing an extra memo field in the issue database. The disadvantage is that you cannot use Vizor Admin or the Web View Editor to customize the input text area (for example, to change the caption or position of the control).

HTML CODE AFTER CONTROL:

```
<script type='text/javascript'>
document.write(parent.objCustomCode.getCodeAfterField('%fieldname%', ''));
</script>
```

HTML CODE BEFORE CONTROL:

```
<script type='text/javascript'>
document.write(parent.objCustomCode.getInputMemoField('mem_desc', '%fieldname%'));

</script>
```

NEW FUNCTION IN CUSTOMCODE.JS:

```
function getInputMemoField( inputFld, logFld )
{
    var strOutput;

    strOutput="<textarea  class='MemoFieldWidth'  ";
    strOutput+="name='" + inputFld + "' ";
    strOutput+="wrap=physical  rows=10 cols=73";
    strOutput+="onfocus=\"parent.MemoGotFocus('"
+ inputFld + "')\" ";

    strOutput+="onchange=\"parent.OnMemoRecordChanged('" + inputFld + "')\" ";
    strOutput+="onkeypress=\"parent.OnMemoRecordChanged('" + inputFld + "')\"></textarea><br>";

    strOutput+="<script type='text/javascript'>";
```

```
strOutput+="document.write (parent.objCustomCode.getCodeAfterField('" + inputFld + "', '" + logFld  
+ "'))</script>";  
  
    strOutput+="</script>";  
  
    return strOutput;  
  
}
```

NEW DECLARATION IN CUSTOMCODE.JS:

```
// Declaration of the public functions  
  
    this.getInputMemoField=getInputMemoField;
```

CHANGING THE FORMAT OF ATTACHMENTS

The **Detailed** choice in the **Include** list of the Notification Editor attaches a report to the notification message. This report includes most fields for an issue.

If you do not use Crystal Reports, attachments can be HTML files or text files. If you do use Crystal Reports, attachments are RTF files by default. You can change this to be PDF, text, or a number of other formats. See [Viewing Reports](#) for more information on using Crystal Reports for listing reports.

To change the format of a notification attachment:

- 1 Open the project definitions database.
- 2 In **tblMailContents**, change the **nFormatType** for the attachments. (Attachments have **nType** = 3 in **tblMailContents**).

NFORMATTYPE CONSTANTS:

NoFormat	0	Excel40	20
CrystalReport	1	Excel50	21
DataInterchange	2	Excel50Tabular	22
RecordStyle	3	ODBC	23
RichText	4	HTML32Standard	24
Comma Separated Values	5	Explorer32Extend	25
TabSeparatedValues	6	NetScape20	26
CharSeparated Values	7	Excel70	27

Text	8	Excel70Tabular	28
TabSeparatedText	9	Excel80	29
PaginatedText	10	Excel80Tabular	30
Lotus123WKS	11	PortableDocFormat	31
Lotus123WK1	12	HTML40	32
Lotus123WK3	13	CrystalReport70	33
WordForWindows	14	ReportDefinition	34
Excel21	18	ExactRichText	35
Excel30	18	XML	36

ATTACHING INFORMATION

By default, the **Revision Record** and **Summary** are included in the body of the e-mail notification message. You can include this information as separate attachments.

To do this, open the project definitions database and in **tblMailContents** change the **nType** to 3. **nType** determines where the content is put in the e-mail message.

nType	Description
1	Subject
2	Body of e-mail message
3	Attachment

ADDING CUSTOM MAIL CONTENTS

In the Notification Editor, the **Include** list specifies what to include in an e-mail notification message. You can add items to this list. The items can be used as the message subject, inserted in the message body, or included as attachments to the message.

To add an item to the Include list of the Notification Editor:

1 Define a function in the macroXX.bas file.

This function should return the content you want to include in the e-mail message.
macroXX.bas is located in the project sub-folder of the ConnectServer folder (for example:
`ConnectServer\vit\macro70.bas`)

2 Open the project definitions database, and in the tblMacros table, add a row for the new function.

tName is the name of the macro.

tFunctionName is the name of a function defined in macroXX.bas.

nType specifies what arguments the function takes:

- 1 = No Arguments
- 2 = Issue Record
- 3 = Revision history
- 4 = Expression

3 In the tblMailContents table, add a row for the macro, using the same **tName** as in **tblMacros**.

nType specifies where to include the contents:

- 1 = subject
- 2 = body
- 3 = attachment

nFormatType specifies the format:

- 4 = rtf
- 8 = text
- 14 = word
- 31 = pdf
- 32 = html40

When **nType** = 2, **nFormatType** must be 8 (text), otherwise the contents appears as a plain text e-mail message that contains RTF or PDF, which is never sent. This will hold up all your notifications.

EXAMPLE MACROS

FUNCTION TO RETURN A STRING:

```
Public Function GetCustomSubject(oCensusApplication, lngDtsRecordID, oExpression,
recRevisionHistory)
GetCustomSubject = "Your subject goes here."
End Function
```

FUNCTION TO RETURN THE CONTENTS OF A TEXT FILE:

```
Public Function GetCustomMsgBody( oCensusApplication, lngDtsRecordID, oExpression,
recRevisionHistory )

Dim fsObj
Dim txtStreamObj
Dim emailbody
Dim strFileName

strFileName = "C:\Messages\msg.txt"

Set fsObj = CreateObject(
"Scripting.FileSystemObject")

Set txtStreamObj = fsObj.OpenTextFile(
strFileName, 1, False, False)

emailbody = txtStreamObj.ReadAll

txtStreamObj.Close
GetCustomMsgBody = emailbody
End Function
```

FUNCTION THAT TAKES THE REVISION HISTORY AS AN ARGUMENT AND EXTRACTS

A FIELD VALUE:

```
Public Function GetSubject(oCensusApplication, lngDtsRecordID, oExpression,
recRevisionHistory)

Dim lngLastRevisionNumber
Dim rstValue
```

```
Dim variablename

If oCensusApplication Is Nothing Then
Exit Function
End If

If recRevisionHistory Is Nothing Then
Exit Function
End If

'get the last revision number
GetRevisionNumberRange recRevisionHistory, 0, _
lngLastRevisionNumber

' Get the value for the field from the table
' of the last revision in the given set.

Set rstValue = oCensusApplication.CurrentProject._
DataStores.Item(32)._
GetConnection(1).Execute( _
"SELECT tBriefDescription FROM tblDts_History_
WHERE nID=" & lngDtsRecordID _
& " AND nRevisionNumber=" _
& lngLastRevisionNumber, , 1)

' Build subject line that looks like:
'HELPPDESK [Ticket 37]PROJECT [Issue 37] - Cannot print PDF file
If Not rstValue.EOF Then
If Not IsNull(rstValue.Fields(0).Value) Then
GetSubject = "HELPPDESK [Ticket PROJECT [Issue " & _
lngDtsRecordID & "]" - " & _
rstValue.Fields(0).Value
End If
End If
End Function
```

In the **SELECT** statement, **tBriefDescription** is the internal name for the **Summary** field. You can find the internal field names in the **tName** column of the **tblDtsFields** table in the project definitions database.

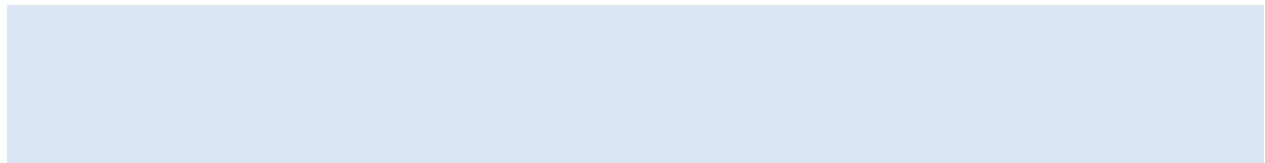
RELATED TOPICS

[Changing the Format of Attachments](#)

[Attaching Information](#)

[Adding Custom Mail Contents](#)

[Adding Notification Reports](#)



ADDING NOTIFICATION REPORTS

Vizor can use reports to format information included in e-mail notifications. For example, the Detailed and Summary items are formatted by reports. You can define new reports and add them to the list of items that can be included in an e-mail notification message.

To add a new report for notifications:

- 1 In Vizor Admin, use the Report Editor to define a new listing report.
- 2 Save and preview the report.
- 3 Exit Vizor Admin.
- 4 Open the project definitions database.
- 5 Open the **tblSystemCustomReports** table, find the record for the report you just created, and get the **nID** value.
- 6 Open the **tblMailContents** table and create a new record.

Field	Description
nID	Next available ID number for a record in the tblMailContents table.
tName	Name displayed in the Include list of the Notification Editor.
nType	2 = include in message body 3 = attachment
tFunction	The GenerateReport function takes care of generating the report for the notification message.
tArguments	<nID>.10, where <nID> is the ID of the report in tblSystemCustomReports .
nFormatType	4 = rtf 8 = text 14 = word

31 = pdf
32 = html40

See [Changing the Format of Attachments](#) for more format type values.

When **nType** = 2, **nFormatType** must be 8 (text).

The report is now available in the **Include** list of the Notifications Editor.

If you're using Crystal Reports for your listing reports, you'll have to create a Crystal Report report file (.rpt) for the new notification report.

TO CREATE AN .RPT FILE FOR YOUR NOTIFICATION REPORT:

- 1 View the report in a Web view.

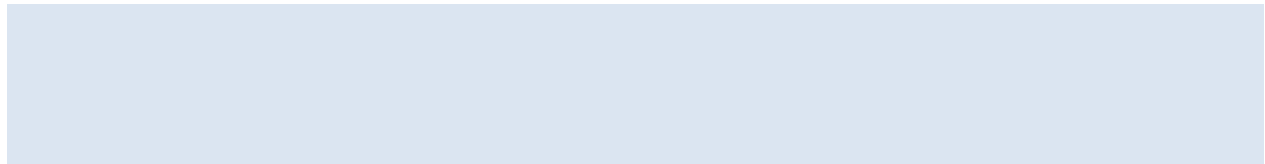
This creates an .rpt file in the CensusWebVD\<project>_<view>\Reports folder. For example:

```
\Program Files\Vector\ConnectServer\CensusWeb\Views\CensusWebVD\HelpDesk_HelpDesk\Reports
```

- 2 Copy the .rpt file to the ConnectServer\<project>\Reports folder. For example:

```
\Program Files\Vector\ConnectServer\HelpDesk\Reports
```

- 3 Rename the file by removing **portrait** (or **landscape**) from the .rpt file name.



SUBMIT-ONLY VIEWS WITHOUT PERSISTENT COOKIES

Submit-only views use persistent cookies to store contact information, such as the user's name and e-mail address. You can set up submit-only views so that submitters do not have to enter their contact information. Consequently, the submit-only views use session cookies only.

To do this, you enter the contact information into Vizor yourself, and then provide each user with their own specific URL.

To set up submit-only views that do not use persistent cookies:

- 1** Use Vizor Web Admin to create contacts if the submit-only view is intended to allow people without Vizor user accounts to submit bugs directly. If the submit-only view is intended for people who have user accounts, the user information already exists in Vizor.
- 2** Send a personal URL for the submit-only view to each person who needs to submit issues.

The personal URL for a contact looks something like this:

`http://server/vit/logon.asp?`

`View=Vizor%2FReport%20Issue&CookieInfo13=14;Emily``http://server/census/logon.asp`

`?View=BugTrk%2FSubmit%20Issue&CookieInfo13=14;Emily`

The first part of the URL is the URL for the Vizor logon page. The second part of the URL (in bold) is the query string. The View= part of the query string specifies the project and view:

`View= <project>%2F<view>`

where:

%2F is the escape sequence for a forward slash (/).

CookieInfo13= specifies the user's contact information:

`CookieInfo13=<User ID>;<Personal Name>`

<User ID> is the numeric ID assigned to the contact or user. You can find this value in the tblUser table of the Users database. **<Personal Name>** is the name of the contact or user.

Spaces in the project, view, and personal name must be replaced with the escape sequence %20.

CHAPTER 30 - BRANDING WEB VIEWS

You can customize the look of a Web view by editing the CensusMain.css style sheet in CensusWeb\Views\CensusWebVD\Theme\css.

Put a copy of your customized style sheet in the CUSTOMIZEDFILES folder. For example, in #AllProjects#\Theme\css or #Project#vit\Theme\css.

REPLACING LOGOS

To replace the logo, edit the **td.Logol** and **div.Logo** rules in CensusMain.css. The **background-image** property specifies the logo image.

Note that Vizor uses two versions of the logo, **logol** for light backgrounds (Web view page) and **logo** for dark backgrounds (logon page).

```
/* logo on a Web view page */
td.Logol
{
    background-image:url(../images/logos/logol.gif);
    background-repeat:no-repeat;
    display:block;
    left:27px;
    width:200px;
    height:60px;
    position:absolute;
}

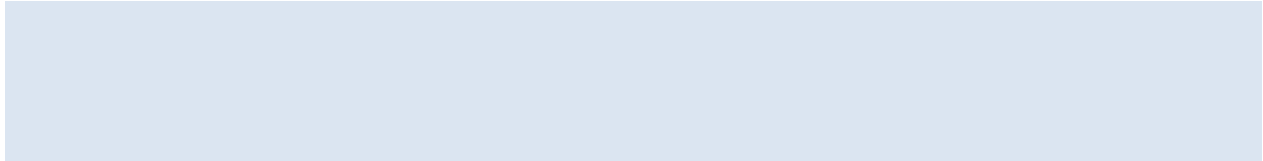
/* logo on the Vizor logon page */
div.Logo
{
    background-image:url(../images/logos/logo.gif);
    background-repeat:no-repeat;
    display:block;
    left:27px;
    width:200px;
    height:60px;
    position:absolute;
    border:3 orange solid;
}
}
```

View Toolbar

The View toolbar is the top part of a Web view, where the logo and the **Help**, **Log Off**, and other buttons are displayed.

body.TopToolbarBody sets the background color.

td.LogoI sets the logo image displayed in the toolbar.



SUMMARY LIST TOOLBAR

The toolbar has three parts: the top (with the Query, Sort, and Layout lists), the middle (the Ad-hoc Query Editor), and the bottom (used for bottom rounded corner images).

For the top part:

- **body.ToolbarBody** sets the background color.
This class is also used by many dialogs, such as **Attach File**, **History**, and **Options**.
- **ToolbarLeft**, **ToolbarCenter**, and **ToolbarRight** set the background images. **ToolbarCenter** also sets the font properties for the captions.

For the middle part:

- **body.ToolbarM** sets the background color.
- **ToolbarMLeft**, **ToolbarMCenter**, and **ToolbarMRight** set the background images. **ToolbarMCenter** also sets the font properties for the captions.

For the bottom part:

- **body.ToolbarB** sets the background color.
- **ToolbarBLeft**, **ToolbarBCenter**, and **ToolbarBRight** set the background images.

SUMMARY LIST HEADER

The appearance of the heading row for the Summary List is controlled by:

- **headerSummaryList**
- **HeaderSummaryListCellCenter**
- **HeaderSummaryListCellLeft**
- **HeaderSummaryListCellRight**

headerSummaryList sets properties for the entire table. The three **HeaderListSummaryCell** classes are used to set background images. If you don't want to use images, you can comment out the **HeaderListSummaryCell** classes and use **headerSummaryList** to set the background and text colors.

SUMMARY LIST

body.SummaryListBody formats the page background and text.

The appearance of links in the Summary List are controlled by **body.SummaryListBody a:link** and **body.SummaryListBody a:visited**.

To change the shading of the rows in the list, edit these CSS classes. The two sets of CSS classes allow for rows with alternating shading.

- **SummaryList1**
- **SummaryList1Hover** (for Internet Explorer)
- **SummaryList1HoverN** (for Netscape)
- **SummaryList2**
- **SummaryList2Hover** (for Internet Explorer)
- **SummaryList2HoverN** (for Netscape)

TABS

The appearance of the selected tab is controlled by:

- **TabsLeftCornerUp**
- **TabsBackgroundUp**
- **TabsRightCornerUp**

The appearance of the unselected tabs is controlled by:

- **TabsLeftCornerDown**
- **TabsBackgroundDown**
- **TabsRightCornerDown**

For example, you could replace the background images or use solid colors instead of images.

To customize the font used for the text on the tabs, edit **TabsBackgroundDown** and **TabsBackgroundUp**.

The area to the right of tabs (where the next/previous page controls are located) can be styled with **tabsBody** (the background color) and **tabsBorder** (the bottom border), and **PageNavigation** (the *Page X of Y* text).

To put a dividing border between the Summary List and the tab area, use **border-top** with **tabsBody**.

FORM

Each tab in a Web view is a form that contains a table. The HTML looks like this:

```
<body class="RecordBody" >
  <form name="Record">
    <table class="RecordTab" >
```

So, for example, to change the background color of the tabs, edit the **body.RecordBody** class in **CensusMain.css**.

To apply styles to the form element, you can use this selector in the style sheet:

```
body.RecordBody form { ... }
```

FIELD CAPTIONS

Field captions in a Web view are formatted with the CSS classes **normalfield**, **requiredfield**, and **disabledfield**.

FORM ELEMENTS

The CSS class applied to a form element is specified by the **CSS Class** field attribute in the Web View Editor. You can customize the default CSS classes or define your own and apply them to fields through the Attribute Editor.

The default CSS classes for form elements are **TextBoxWidth**, **ComboBoxWidth**, and **MemoFieldWidth**, **InputAttachment** (for the **Attachments** box), and **InputCalendar** (for date fields).

COMMAND BAR

The command bar is the bar at the bottom of the page that contains the **Issue** box and the **New**, **Save**, **Cancel**, and **Print** buttons.

body.CommandBarBody sets the background color for the command bar. You can set the text color here (except for the text in the message area).

CommandBarBLeft, **CommandBarBCenter**, and **CommandBarBRight** specify the background images for the command bar.

CommandBarMsgLeft, **CommandBarMsgCenter**, and **CommandBarMsgRight** specify the background images for the message area (For example: *Ready. Click New to create a new issue.*) of the command bar.

ControlsCaptionDark sets the style for the **Current Issue** label.

CurrentRecord and **CurrentRecordHover** set the style for the **Current Issue** text box.

ABOUT BRANDING DIALOGS

Dialogs are pages that users open by clicking a button. This includes the Attachments, Reports, Revision History, Password, Options, and Contacts dialogs.

Most of the dialogs use the same style sheet as the Web view. However, the Contacts and Password dialogs use the CensusMain.css style sheet found in CensusWebVD\Admin\THEME\css.

COMMON STYLES FOR ALL DIALOGS

body.toolbarBody sets the background color for the dialog pages (except for the Reports dialog, which uses **body.ReportsBody**).

The box with rounded corners is a set of background images specified by these classes:

- **BoxTLeft**, **BoxTCenter**, and **BoxTRight**
- **BoxMLeft**, **BoxMCenter**, and **BoxMRight**
- **BoxBLeft**, **BoxBCenter**, and **BoxBRight**

ATTACHMENTS DIALOG

body.toolbarBody sets the background color for the page.

DialogLabel formats the captions.

InputFile and **InputFileHover** format the text boxes.

The list of attached files is displayed in a table. **HeaderSummaryListCellCenter** formats the heading row (where the **File Name** and **Remove** captions are displayed).

SummaryList1 and **SummaryList2** format the list of files.

DialogLabel formats the file names.

REPORTS DIALOG

body.ReportsBody sets the background color for the page.

The appearance of the tabs is controlled by the same CSS classes that apply to the tabs in a Web view (see [Tabs](#)).

The **Report Name** caption uses the **normalfield** class.

The **<select>** element that lists the available reports uses the **ReportList** class.

The **Layout:** and **Viewer Type:** captions use the **AdminNormalTextDarkBG** class. The corresponding **<select>** elements do not use a CSS class, but you can use their IDs (LayoutType and ViewerType) with ID selectors in the CSS file.

REVISION HISTORY DIALOG

This dialog uses only the common styles.

OPTIONS DIALOG

body.toolbarBody sets the background color for the page.

Captions use the **normalfield** class.

Form elements (such as `<input>`) are in a `<TD>` that uses the **normalfield** class, but the form elements themselves have no class.

CONTACTS DIALOG

This dialog uses the CensusMain.css style sheet in CensusWebVD\Admin\THEME\css.

body.RecordBody sets the background color for the page.

The caption for the **Display** list uses the **DialogLabel** class, and the `<select>` element uses **AdminComboBoxWidth**.

On the input form for contact information, the captions use **requiredfield** and **DialogLabel**. The form elements all use **AdminComboBoxWidth**.

PASSWORD DIALOG

This dialog uses the CensusMain.css style sheet in CensusWebVD\Admin\THEME\css.

body.toolbarBody sets the background color for the page.

normalfield formats the captions **Old Password**, **Password**, and **Confirm Password**.

AdminTextBoxWidth and **AdminTextBoxWidthHover** format the text boxes.

AdminNormalTextDarkBG formats the **Password can contain a maximum of 14 characters message**.

ABOUT BRANDING THE LOGON PAGE

The heights of some of the table cells are hard-coded in `CensusWebVD\logon.asp`.

LogonBody sets the background color for the page.

The images and background colors for the box with rounded corners are set by these classes:

Logon-t-left, td.Logon-t-center, div.Logon-t-right

- **Logon-m-left, div.Logon-m-center, td.Logon-m-right**

- **Logon-b-left, td.Logon-b-center, div.Logon-b-right**

Note the mix of **div** and **td** elements. If you want to replace the images with colors and borders, set the left and right classes to the background color and use the center classes to define your border.

div.Logo inserts the Vizor logo.

div.Logon-decor-img inserts the keyboard image with the orange border

LogonForm positions the logon form (the **Logon Name** and **Password** text boxes).

div.LogonField and **div.PwdField** position the form elements.

div.LogonCaption formats the **Logon Name** and **Password** labels.

input.LogonInput and **input.LogonInputHover** format the text boxes.

a.LogonState1 and **a.LogonState1:hover** set the images used for the **Logon** button.

HelpState1 and **HelpState1:hover** set the images used for the **Help** button.

VALIDATE LOGON PAGE

The ValidateLogon page displays the available views. With a few exceptions (documented below), this page uses the same styles as the Logon page.

The heights of some of the table cells are hard-coded in **CensusWebVD\validatelogon.asp**.

ExitState1 and **ExitState1:hover** specify the images used for the **Log Off** button.

div.ViewList-m-center is used instead of **div.Logon-m-center**.

div.ViewList-decor-img is used instead of **div.Logon-decor-img** to display the keyboard image with the orange border

div.ViewLinks positions and sizes the area used to display the list of available views. This class also adds the vertical orange bar.

div.LogonCaption formats the **Click the view you want to open** string.

ViewListCaption formats the DIVs that contain the project names.

ViewList-Link, **ViewList-Link:link**, **ViewList-Link:visited**, and **ViewList-Link:hover** format the view names, which are hyperlinks.

ADDING ROWS

By default, the timesheet project allows users to track weekly time spent for up to four different projects. If you need to track time for more than four different projects or tasks, you can add more rows to the timesheet.

Step 1: [Create fields in the Field Editor](#)

- 1 Create a new choice list for selecting projects. Copy the Project1 field and name it Project5.
- 2 Create the input fields for the days of the week.
 - a Copy the **P1HRsMon** field and name it **P5HrsMon**.
 - b Copy **P5HrsMon** and name it **P5HrsTue**.
 - c Copy **P5HrsTue** and name it **P5HrsWed**.
 - d Continue until you have input fields for all seven days of the week.
- 3 Create a field to hold the weekly total. Copy Proj1TotalWkHrs and name it **Proj5TotalWkHrs**.
- 4 Click **Apply** to add the fields to the project.

Step 2: [Export the new fields to the view](#)

- 1 In the Web View Editor, click the timesheet view in the **Project / View** list.
- 2 Under **Fields**, click the last field (**Product4**) in the **Export to View** list. Click **Attributes**, and delete the value of the Html code after control attribute.
- 3 Move the new fields from the **Available** list to the **Export to View** list.

The new fields are added to the end of the **Export to View** list. The order of the fields does not matter.

- 4 Click the last field in the **Export to View** list. Click **Attributes**, and set the value of the Html code after control attribute to "-->".

Step 3: [Edit timesheet.js](#)

- 1 Open timesheet.js in your favorite text editor. You can find this file in the folder:

```
CensusServer\CensusWeb\CUSTOMIZEDFILES\#Project#Timesheet\#AllWebViews#\Js
```

This is the file to edit if you want to add a row to all Web views of your timesheet project. If you want to add rows to only certain Web views, you'll need to create a new folder. For example:

```
CensusServer\CensusWeb\CUSTOMIZEDFILES
\#Project#Timesheet\#WebViews#Timesheets_MyTimesheets\Js
```

- 2 At the top of the file, declare and initialize a variable that holds the ID of the new row:

```
var PROJ_FIVE_ID=4;
```

3 In the **WriteTimesheetGrid()** function, create a new array that holds the row information, and store this new array in **m_arrTimesheet**:

```
m_arrTimesheet[PROJ_FIVE_ID]=new Array(
    "cbo_97_cbo_97_Project5"
    + INNER_ARRAY_SPLITTER + INNER_ARRAY_SPLITTER+ strProjComboOptions,
    "nbr_98_nbr_98_P5HrsMon",
    "nbr_99_nbr_99_P5HrsTue",
    "nbr_100_nbr_100_P5HrsWed",
    "nbr_101_nbr_101_P5HrsThu",
    "nbr_102_nbr_102_P5HrsFri",
    "nbr_103_nbr_103_P5HrsSat",
    "nbr_104_nbr_104_P5HrsSun",
    "nbr_105_nbr_105_Proj5TotalWkHrs"
    + INNER_ARRAY_SPLITTER + CONTROL_TYPE_TOTAL);
```

4 Use the ID of the new row (PROJ_FIVE_ID) as an index into the **m_arrTimesheet** array.

In the new array, specify the names assigned to form controls in the generated HTML. These names are based on the **tBoundControlName**, which you can find in the **tblDtsFields** table of the project definitions database. For example, if the **tBoundControlName** is **cbo_97_Project5**, then the form control name is **cbo_97_cbo_97_Project5**.

Add a call to **GetTimesheetProjRow()** for the new row. Put this call before the call to **GetTimesheetDayTotalRow()**.

```
// Calls to GetTimesheetProjRow() for previous rows

strHTML+=m_oTimesheet.GetTimesheetProjRow(
m_arrTimesheet[PROJ_FIVE_ID]
);

// call to GetTimesheetDayTotalRow()
```

Step 4: Update layouts, queries, sorts, and reports

Any layouts, queries, sorts, or reports that use values from the rows in a timesheet must be updated.

For example, the Current Issue - Detailed report is used to print the current timesheet. If you don't update this report, it won't show the rows you added.

Step 5: Regenerate the Web views

If you get javascript errors when you try to open the Web view, log off and fix timesheet.js. You don't need to regenerate the Web view, but you must copy timesheet.js from the CUSTOMIZEDFILES folder to the Web view folder.

CHANGING THE TIMESHEET STARTING DAY

The weekly timesheets can start on any day of the week. By default, the weekly timesheets start on Monday.

To change the timesheet starting day:

- 1** In the Web View Editor, click the timesheet view in the **Project / View** list.
- 2** Under **Fields**, click the last field before the fields that make up the actual timesheet. By default, this is the **Weekly Total Hours** field.
- 3** Click **Attributes** and then click the attribute value of the [Html code after control](#) attribute. This should take you to the end of the line of text, where you'll see the `START_MONDAY` constant as the last argument to a function.
- 4** Change this to the new starting day. Use one of these constants:

`START_MONDAY`

`START_TUESDAY`

`START_WEDNESDAY`

`START_THURSDAY`

`START_FRIDAY`

`START_SATURDAY`

`START_SUNDAY`

CREATING A TIMESHEET

When you create a new project based on the **Timesheet** project, you must add the timesheet to a Web view of the new project. After that, you can copy the Web view to create more views of the new project.

Project:	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Project Total
Daily Total:								

To add a timesheet to a Web view:

- 1 Export the fields displayed above the timesheet (by default: **Employee**, **Weekly Start Date**, and **Weekly Total Hours**).
- 2 Enter the following HTML into the **HTML code after control** attribute of the last field currently in the **Export to View** list.

Instead of typing in this HTML, you can copy the **HTML code after control** value from the **Weekly Total Hours** field in the original **Timesheet / Timesheet** Web view.

```
<script type="text/javascript" src="../js/Timesheet.js"></script>
<select name="TS_cbo_vals" class="HiddenObject"><WC@53></WC@53>
</select><script>WriteTimesheetGrid(escape
(document.Record.TS_cbo_vals.innerHTML),START_MONDAY);</script><!--
```

- 3 Export the timesheet fields. The order does not matter.

DailyTotalMon, ..., DailyTotalSun (7 fields)

P1HrsMon, ..., P1HrsSun (7 fields)

P2HrsMon, ..., P2HrsSun (7 fields)

P3HrsMon, ..., P3HrsSun (7 fields)

P4HrsMon, ..., P4HrsSun (7 fields)

Proj1TotalWkHrs, ..., Proj4TotalWkHrs (4 fields)

Project1, ..., Project4 (4 fields)

- 4 Enter the string "-->" (without the quotation marks) into the **HTML code after control** attribute of the last field in the **Export to View** list.

CHAPTER 31 - DATABASES

OVERVIEW

Vizor supports databases in multiple versions of SQL Server.

By default, SQL Server databases are located on the SQL Server computer in the DATA folder or the default folder specified by the SQL Server administrator. By default, Jet/Access databases are stored in the ConnectServer folder (for example, C:\Program Files\Vector\ConnectServer)

Project databases

A project includes two main databases:

- A database of records of the main entity stored in that project (assets, tickets, people, etc.) and their revision histories.
- A database of definitions, which contains defined business rules, queries, sorts, layouts, reports, notifications, fields, and tabs available in the project.

Users database

In addition to the project databases, there is a database of users, contacts, employees and companies. The users database is shared by all projects.

DAT DATABASE

SQL Server

Issues are stored in the **<project>01_Dat** database, where **<project>** is the name of the Vizor project. The database consists of:

- **<project>01_Dat.mdf**
The master data file (MDF) of the database.
- **<project>01_Dat_log.LDF**
Log file that holds the log information used to recover the database. There must be at least one log file for each database, although there can be more than one.

Microsoft Access

Issues are stored in the file **<project>01.dat**, where **<project>** is the name of the Vizor project.

MAIN TABLES

tblDts

Stores most of the information entered for an issue. This includes most of the fields on the **Overview**, **Description**, and **Detail** tabs, as well as the information entered on any custom tabs added to the project.

tblFixInformation

In previous releases of Vizor, new fields added to the **Resolution** tab were stored in **tblFixInformation**. The default fields are stored in **tblDts**.

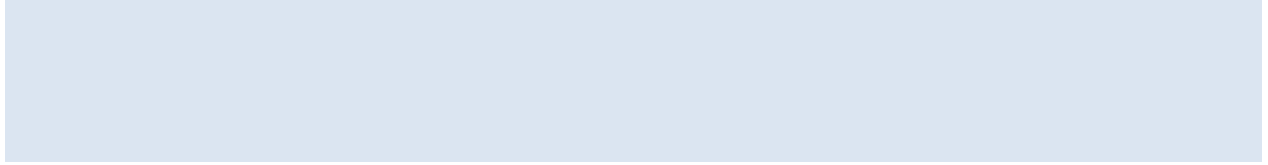
Any new fields added to the **Resolution** tab are stored in **tblDts**. Stores fix-related information entered on the **Fix** tab.

tblAttachments

Stores any files attached to an issue.

tblRevisionHistory

Stores the revision history for issues.



DEFINITIONS DATABASE

SQL Server

The project definitions database is named **<project>02_DEF**, and consists of these files:

- **<project>02_DEF.mdf**
- **<project>02_DEF_log.LDF**

Microsoft Access

The project definitions database is named **<project>02.def**, where **<project>** is the name of the Vizor project.

MAIN TABLES

tblDtsFields

Defines the fields available in a Vizor project. The Field Editor in Vizor Admin provides a user interface for editing most of the columns in this table.

In **tblDtsFields**, the **tWhere** column specifies a **Where** clause that controls the contents of a choice list. The Vizor project uses the **Where** clause to build the **Contact**, **Owner**, and **Submitter** choice lists.

For example, this Where clause removes macros (such as <User>) and contacts from the **Owner** list:

```
Where (([nID] >= 0 And [tName] <> '')Or
[tName] = '<User>') And fOriginator = 0
```

- The **[nID] >= 0** condition removes the macros defined in **tblUser**:

User ID	User Name
-4	<Contact>

-3	<User>
-2	<Previous Owner>
-1	<Owner>

The user with **nID = 0** is the **<None>** macro.

- The **fOriginator = 0** condition removes contacts, so that the list contains users only.
- **tName** is the name of the user.

tblState

Specifies the list of choices for the **State** field. The **State** field is a read-only field that is set based on the **Progress** field.

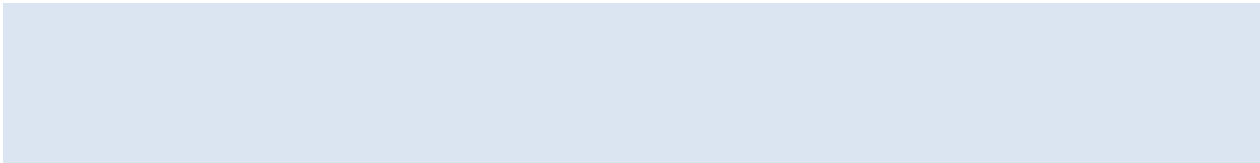
tblSubState

Specifies the list of choices for the **Progress** field. Each possible substate corresponds to a possible **State** value (**Open** or **Closed**).

Choice List Tables

A choice table specifies the possible values for a choice list field. Each row in a choice table includes an **nID**, a **tName** (the choice text), and optionally, an **nChoiceID** (for non-alphabetical choice lists).

In the issue database, multi-choice fields store a semi-colon separated list of the **tName** values (the choice text strings), not the **nID** values.



USERS DATABASE

All Vizor projects share the same database of users. The users database also stores global choice lists.

Microsoft Access

The user database is named **users.mdb**.

SQL Server

The user database is named **USERS_MDB**, and consists of these files:

- USERS_MDB.mdf
- USERS_MDB_log.LDF

MAIN TABLES

tblUser

Lists the users and contacts. Users have logon names and passwords. Contacts are people who report issues, but do not have a logon account.

If you add a field to the **Origin** tab, the field is added to **tblUser**, not **tblDts**. The **Origin** tab is a read-only tab that displays user and contact information. The user and contact information is actually entered in Vizor Web Admin (**Security** tab). This allows user and contact information to be entered once for all issues, instead every time for each issue.

GLOBAL DATABASES

These databases are shared by all projects:

- **CensusWeb**

CENSUSWEB_MDB in SQL Server

Stores Web view definitions. Used by the Web View Editor.

- **Licenses**

LICENSES_MDB in SQL Server

Stores system information, list of projects, features, licenses.

Advanced Topic

DATASTORES

DataStores are databases and other sources of data such as XML files.

Some advanced customizations require working with datastore or database IDs. These are usually entered in tables and scripts.

Global Datastores are defined in the censation.xml file.

Project DataStores are defined in tblProjectDS [Licenses_mdb].

The following is a list of data store types:

DataStore Type	ID	Notes
ClientDB	1	Internal use
Admin	2	Admin Tool database
Library	4	Admin Tool Library (AdminLib)

		database
Licenses	8	Licenses_mdb
Users	16	Users_mdb
Dat	32	Project DAT database of the project in context
Def	64	Project DEF database of the project in context
Usr	128	Project USR database of the project in context
Web	256	Census_Web database of the project in context
SystemDB	512	Security databases for JET logons Censys.mdw
InventoryDB	1024	Integration with Inventory
SecuredDB	65536	Internal use
CenAuth	131072	Reserved authentication use

RELATED TOPICS

[Database Overview](#)

[Issue Database](#)

[Project Definitions Database](#)

[Users Database](#)

[Other Project Files](#)

[Relocating Databases to SQL Server](#)

[Logging Off Users](#)

[Troubleshooting Vizor Web Admin](#)

CHAPTER 32 - BACKUPS

ABOUT BACKING UP

You should regularly back up your Vizor databases. Besides backing up all the issue data entered into the system, you should also (though perhaps less frequently) back up the files and databases that define your projects and Web views.

BACKING UP JET/MS ACCESS DATABASES

By default, Access Enabled databases use the Jet/Access database engine, and are located in the **ConnectServer** folder on Web server. To back up the Jet/Access databases, you just have to back up the databases files. By default, Access Enabled databases use the Jet/Access database engine, and are located in the Vizor Server folder on Web server. To back up the Jet/Access databases, back up the databases files.

To back up MSDE 2000 databases:

- If you have Microsoft Access 2000, you can use the BACKUP command in the Database Utilities menu of an Access project to back up an MSDE database.
- If SQL Client Tools are installed, you can use SQL Enterprise Manager to back up an MSDE database.
- If you don't have Microsoft Access 2000 or the SQL Client Tools, then you can use the Transact-SQL BACKUP DATABASE statement, and run Osql.exe (a command-line Query tool).

See Q241397 in the Microsoft Knowledge Base.

BACKING UP SQL SERVER DATABASES

The default Access databases can be migrated to SQL Server. After migration, the databases are located in an admin-specified folder on a SQL Server computer, which may or may not be the same computer as the Web server

To back up SQL Server databases, you can use the SQL Server backup functionality to do incremental or complete backups of the databases.

LOCATING DATABASES

By default, all databases are stored in the **ConnectServer** folder on your Web server. However, the databases can be relocated to different folders on different computers. For example, the databases may be relocated to a separate SQL Server computer.

To locate the databases, log on to Vizor Web Admin and go to the **Relocation** tab.

ATTACHMENTS

By default, attachments are stored on the Web server. For example, the files attached to issue #3 in the project are stored in the folder

```
ConnectServer\vit\Data\Attachments\3
```

Attachments may be relocated to a separate file server. To check the location of the attachments, log on to Vizor Web Admin, go to the **Relocation** tab, and click **Relocate Attachments**.

WEB VIEWS

By default, Web views are stored on the Web server in the **ConnectServer** folder:

```
ConnectServer\CensusWeb\Views\CensusWebVD\
```

To check the location of your Web views, log on to the Web View Editor and in the shortcut bar click **URL**.

RELATED TOPICS

[Backing Up Issues](#)

[Backing Up Projects](#)

[Backing Up the Users Database](#)

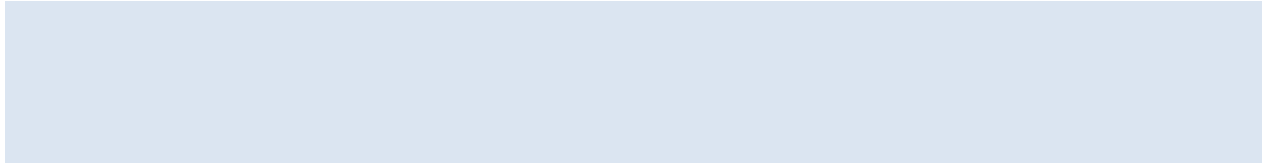
[Backing Up System Files](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Performing Hot Backups](#)

[Restoring Backups](#)



BACKING UP ISSUES

To back up the issue data stored in a project, you need to back up:

- The issue database, which stores all the issue information entered.
- The files attached to issues. These files are stored outside of the issue database, in the file system.

WHAT TO BACKUP

All issue information (except for attached files) is stored in the issue database.

Database engine	Database
Jet/MSAccess	<project>01.dat
SQL Server	<project>01_DAT

By default, attachments are stored in the project folder (for example, C:\Program Files\Vector\ConnectServer\vit\Data\Attachments). Note that the attachments may be stored in a different folder or on a different computer.

NOTES

- To check the location of the databases, log on to Vizor Web Admin and click the Relocation tab.
- To check the location of the attachments, log on to Vizor Web Admin, click the Relocation tab, and click Relocate Attachments.
- You can use Vizor Admin to verify the location of the issue database. In the Project list, click the project to display the **Project Properties**. The **Project Location** specifies the location of the issue database.

RELATED TOPICS

[About Backing Up](#)

[Backing Up Projects](#)

[Backing Up the Users Database](#)

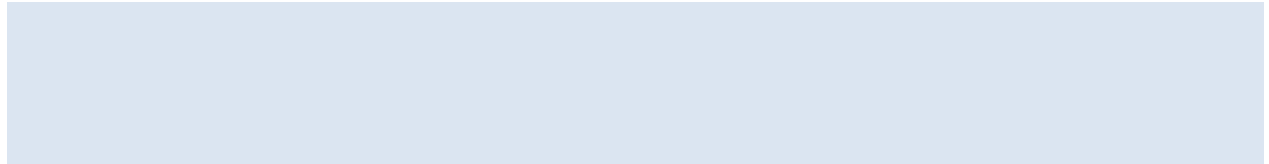
[Backing Up System Files](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Performing Hot Backups](#)

[Restoring Backups](#)



BACKING UP PROJECTS

Projects include the definitions of fields, queries, sorts, layouts, reports, and notifications.

WHAT TO BACKUP

Project definitions are stored in the definitions database.

Database engine	Database
Jet/MSAccess	<project>02.def file
SQL Server	<project>02_DEF database

In addition to the definitions database, there are several related files you should also back up.

Project File	Description
<project>.cen	Used for project initialization and versioning
<project>03.usr	Access database used by Vizor Admin

Note that Vizor automatically backs up the **tblDtsFields** table in the *definitions database*. This table contains the definitions of the fields in the project. The backup copies of the table are stored in the *definitions database*.

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up the Users Database](#)

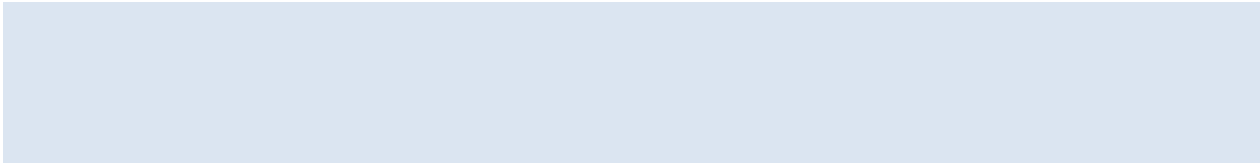
[Backing Up System Files](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Performing Hot Backups](#)

[Restoring Backups](#)



BACKING UP THE USERS DATABASE

The users database contains all user and contact information, and is shared by all projects.

WHAT TO BACKUP

If at least one project uses SQL Server, the user data is stored in a SQL Server database. Otherwise, the user data is stored in a Microsoft Access .mdb file. The users database is located in the ConnectServer folder (for example, C:\Program Files\Vector\ConnectServer).

Database engine	Database
Jet/MSAccess	users.mdb
SQL Server	USERS_MDB

RELATED TOPICS

- About Backing Up
- Backing Up Issues
- Backing Up Projects
- Backing Up System Files
- Backing Up Web Views
- Automatic Backups
- Performing Hot Backups
- Restoring Backups

BACKING UP SYSTEM FILES

We recommend you do a general backup of the files in the ConnectServer folder.

WHAT FILES TO BACKUP

You should backup these files:

- Censys.mdw (security database that contains users, groups, and passwords).
- The license database.

Database engine	Database
Jet/MSAccess	Licenses.mdb
SQL Server	LICENSES_MDB

- CenSession.xml
- CenInfoSession.xml

Vizor automatically creates backup copies of CenSys.mdw and CenSession.xml in the ConnectServer folder when the files change. A mirror of CenSys.mdw is also available (the mirror file is named CenSys_mir.mdw).

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up Projects](#)

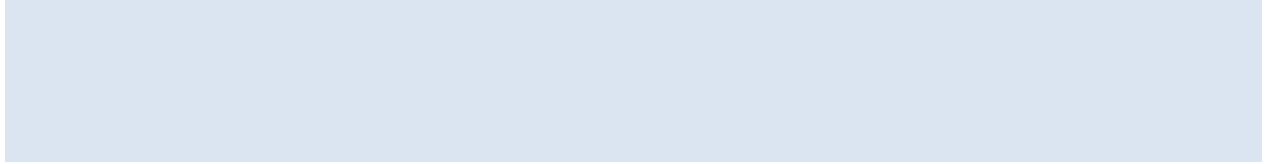
[Backing Up the Users Database](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Performing Hot Backups](#)

[Restoring Backups](#)



BACKING UP WEB VIEWS

WHAT TO BACKUP

You should backup these databases, files, and folders:

- The database of Web view definitions and settings.

Database engine	Database
Jet/MSAccess	censusweb.mdb
SQL Server	CENSUSWEB_MDB

- The CUSTOMIZEDFILES folder. This folder contains all the customized Web view files (for example, custom reports).
- The generated Web view files. Each Web view has its own folder. For example:

```
C:\Program Files\Vector\ConnectServer
\CensusWeb\Views\CensusWebVD\HelpDesk_HelpDesk C:\Program Files\Vector\ConnectServer
\CensusWeb\Views\CensusWebVD\HelpDesk_HelpDesk
```

- The project folders. Each project has its own folder. For example:

```
C:\Program Files\Vector\ConnectServer\HelpDesk
```

The project folder stores attachments, the reports used to format notifications, the macro70.bas file, and the queue of notification messages.

Note that the attachments may be stored on a different computer. To check the location of the attachments, log on to Vizor Web Admin, click the **Relocation** tab, and click **Relocate Attachments**.

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up Projects](#)

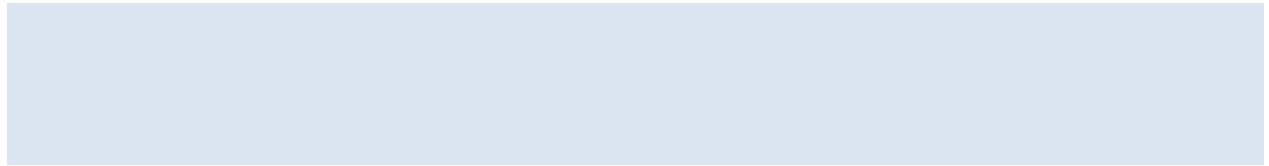
[Backing Up the Users Database](#)

Backing Up System Files

Automatic Backups

Performing Hot Backups

Restoring Backups



AUTOMATIC BACKUPS

Vizor automatically backs up:

- The tblDtsFields table in the project definitions database. This table contains the definitions of the fields in the project.
- CenSession.xml
- CenSys.mdw

You can disable these automatic backups by setting the **Enable Backups** option to **No** (in the **Tools** menu, click **Options**).

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up Projects](#)

[Backing Up the Users Database](#)

[Backing Up System Files](#)

[Backing Up Web Views](#)

[Performing Hot Backups](#)

[Restoring Backups](#)

PERFORMING HOT BACKUPS

Hot backups are backups performed while the Vizor system is in use. Hot backups are ideal for continuous operations.

To perform hot backups, the Vizor administrator has to use database-specific tools:

- To back up data in a SQL Server database, use the built-in SQL Server backup utility.
- If SQL Client Tools are installed, you can use SQL Enterprise Manager to back up an MSDE database.
- To back up Jet/MSAccess databases, use a third-party tool like FMS Total Visual Agent.

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up Projects](#)

[Backing Up the Users Database](#)

[Backing Up System Files](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Restoring Backups](#)

RESTORING BACKUPS

After restoring backups of the issue database and the project definitions database, use the Integrity editor to validate the integrity of fields, relationships, and the issue database.

RELATED TOPICS

[About Backing Up](#)

[Backing Up Issues](#)

[Backing Up Projects](#)

[Backing Up the Users Database](#)

[Backing Up System Files](#)

[Backing Up Web Views](#)

[Automatic Backups](#)

[Performing Hot Backups](#)

CHAPTER 33 - RELOCATING A VIZOR INSTALLATION

ABOUT THE RELOCATION WIZARD

Vizor includes a Relocation Wizard for moving its installations from one computer to another.

When you relocate Vizor, the Relocation Wizard:

- Logs off all users.
- Unloads the virtual directories used by Vizor.
- Stops the Mq Issue Agent service.
- Detaches local SQL Server databases.
- Backs up the ConnectServer and ConnectTools folders before relocating them.
- Copies Vizor to another computer and disables the original installation.
- Regenerates all your Web views after Vizor is relocated. The Web views are all regenerated in the default CensusWeb/Views/CensusWebVD folder.

The Relocation Wizard relocates SQL Server databases if:

- The SQL Server databases are local to the source computer.
- The version of SQL Server on the target computer is greater than or equal to the version of SQL Server on the source computer.

RELATED TOPICS

[Before You Relocate](#)

[Relocating Vizor](#)

[If Something Goes Wrong](#)

[Updating Client Programs](#)

[Enabling Vizor after Relocation](#)

BEFORE YOU RELOCATE

CHECK PERMISSIONS

You must run the Relocation Wizard with a user account that has Full Control permission for the ConnectServer folders on all computers (current location and new location).

CLOSE ALL COPIES OF VIZOR ADMIN AND THE WEB VIEW EDITOR

This includes all copies of the programs, not just those installed on the Web server.

RELOCATE PROJECTS

If you have any projects located outside of the ConnectServer folder, use Vizor Admin to move the projects into the ConnectServer folder. After you relocate Vizor, you can use Vizor Admin to move the projects back outside of the ConnectServer folder.

BACK UP

The Relocation Wizard backs up your ConnectServer and ConnectTools folders, but you may want to create your own backups.

LOG OFF USERS

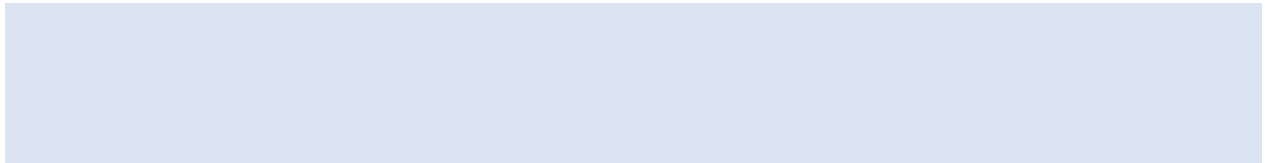
The Relocation Wizard logs off users, but you may want to inform users that Vizor will not be available.

RELATED TOPICS

[Relocating Vizor](#)

[Updating Client Programs](#)

[Enabling Vizor after Relocation](#)



RELOCATING VIZOR

In Vizor, all components are typically installed on one computer: the Web server. The Relocation Wizard allows you to move Vizor to a different computer.

To relocate Vizor:

- 1 Close all copies of Vizor Admin, Vizor Web Admin, and the Web View Editor.
- 2 On the target computer, perform a clean install of Vizor.
- 3 On the target computer, run the Relocation Wizard, called MqRelocateTool.exe, located in ConnectTools\Support\Relocation.
- 4 In the **Logon Name** and **Password** boxes, enter a Vizor user account that belongs to a group with Admins permissions.
- 5 In the **Relocate From** section, enter the location of the current Vizor Server as a UNC path (for example, "\\NANP\ConnectServer").

Make sure the **Computer Name** box contains the name of the computer where the current Vizor Server is located, and the **Local Path** box contains the location of the current Vizor Server relative to a root directory (for example, "C:\Program Files\Vector\ConnectServer").

- 6 In **Relocate To** section, make sure the **Vizor Server** box contains the installation location of the Vizor Server on the target computer, and the **Vizor Tools** box contains the location of Vizor Admin and the Web View Editor on the new computer.
- 7 Select the **Show Advanced Options** check box to show the name of the target computer. If the name is incorrect or missing, enter the name.
- 8 Click **OK** to start the relocation.

When the relocation is complete, the Wizard regenerates all the Web views.

- 9 Update any remote installations of Vizor Admin and Web View Editor to reference the new ConnectServer share.

Check you can open the relocated projects and Web views, and you can run Vizor Admin, Vizor Web Admin, and the Web View Editor.

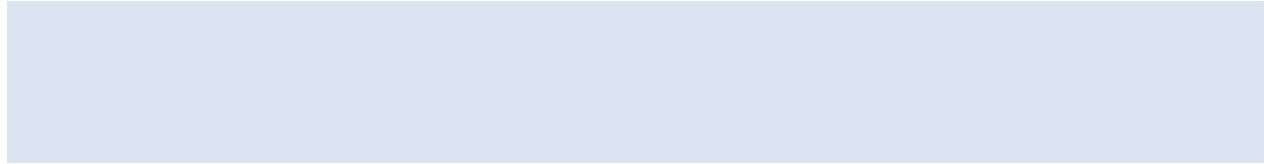
You can now uninstall the old copy of Vizor.

RELATED TOPICS

Before You Relocate

If Something Goes Wrong

Updating Client Programs



IF SOMETHING GOES WRONG

If something goes wrong when you relocate Vizor:

- Check the error log. The Relocation Wizard creates an error log in the ConnectServer\Relocation folder of the computer where you run the wizard.
- The folder name includes the date and time, so the path to the error log looks something like this:

```
ConnectServer
  \Relocation_02-20-2003_10.44.50
    \RelocationErrorLog.txt
```

- Check there is enough free disk space on the new server. Typically, the amount of disk space required is at least twice the size of the current ConnectServer folder.
- If you are relocating SQL projects, reattach the databases to the SQL Server before trying again.

UPDATING CLIENT PROGRAMS

After you relocate Vizor, you must update any client programs. Client programs are copies of Vizor Admin and the Web View Editor installed on computers other than the Web server.

To update the client programs on a computer, you run the MqRelocateClients.exe program on that computer.

When the Relocation Wizard finishes, it puts a copy of the MqRelocateClients.exe program in the ConnectServer\Relocation_<timestamp> folder.

If you send the MqRelocateClients.exe program to users to run, you must also send the INI file located in the same folder as MqRelocateClients.exe.

ENABLING VIZOR AFTER RELOCATION

When the Relocation Wizard copies a Vizor installation to a different computer, it automatically disables the original installation. However, you can re-enable the original installation if you want to continue using it, for example, when you to test an upgrade with live data, while continuing to use the original installation.

To enable Vizor after a relocation:

Apply the following steps to the original Vizor installation.

1 Reattach all SQL Server databases.

To prevent users from using old databases, the Relocation Wizard detaches all SQL Server databases.

Reattach the <project>_DAT databases and the USERS_MDB database using the .mdf or .ldf files in your ConnectServer folder.

2 Rename licenses.mdb.

Locate a file in the ConnectServer folder with a name that looks like Licenses.mdb_2004-06-18_20.23.21.bak. If there are several files like this, use the one with the most recent date in the name.

3 Allow users to log on.

- In the ConnectServer folder, open the file census.ini. Find the [SessionControl] section, and if **Enabled=0**, change it to **Enabled=1**.
- Start Vizor Admin and click **Tools > Integrity**. Select the **Remove Locks** check box and click **Start**.
- In the CensusWebVD folder (by default, ConnectServer\CensusWeb\Views\CensusWebVD), locate a file with a name like Logon.asp_bck_logoff. If this file exists, delete the Logon.asp file and rename Logon.asp_bck_logoff to Logon.asp.
- In the CensusWebAdmin folder (by default, ConnectServer\CensusWeb\Views\CensusWebAdmin), locate a file with a name like Logon.htm_bck_logoff. If this file exists, delete the Logon.htm file and rename Logon.htm_bck_logoff to Logon.htm.

4 Restart the notification service.

If you use notifications, open **Services** and start the **Mq Issue Agent** service.

To open **Services**, click **Start**, click **Control Panel**, double-click **Administrative Tools**, and then double-click **Services**.

INDEX

- %fieldid% variable, 161, 327
- %fieldname% variable, 327, 337
- %fieldvalue% variable, 327, 339, 405
- .cen file, 502
- .def file See project definition, 374
- < less than operator, 104
- <= less than or equal operator, 104
- <Any> macro, 257
- <Empty> macro, 257
- <New Issue> macro, 252
- <None> macro, 103, 127
- <Today> macro, 103
- <User in Group> macro, 246, 252, 289
- <User> macro, 246, 252, 289
- <Value of Field> macro, 156
- > greater than operator, 104
- >= greater than or equal operator, 104
- 404 errors, 382
- ActiveX, 403
- Adding Pop-up Editors, 328
- Adding URL buttons, 339
- Ad-hoc Query Editor, 172, 322
- admin user account, 149
- Admins group, 414, 426, 430
- Alignment attribute, 329
- Always verify locked databases option, 380
- Analyst user account, 149
- Anonymous user account, 350
- Assigned Work Team field, 154
- Attachments, 366, 381
- Attachments command, 366
- Attachments field, 189, 446
- Attributes field, 325
- Authentication method, 163
- Auto Repair, 377
- Automatically setting value, 285
- Automatically Update Choice List attribute, 330, 435
- Backups, 507, 510, 514, 519, 524
- Bandwidth usage, 371

- Bar charts, 127
- Base projects, 414
- Branding, 469, 486
- Byte field size, 196
- Captions, 184, 192, 331, 427, 435
- Categories, 157
- CenInfoSession.xml, 515
- CenSession.xml, 515
- CensusAdminsGroup group, 99, 360
- CensusMain.css, 333, 358, 468, 470, 473, 476, 479, 482, 486
- CensusUser account, 350, 360
- CensusUser user account, 99, 360
- CensusWebVD folder, 356
- CenSys.mdw, 515
- Changing type, 427
- Check boxes, 193
- Child Issues tab, 324
- Choice Editor, 184
- Choice lists, 104, 112, 152, 192, 193, 201, 203, 239, 330, 332, 342, 390, 427, 435, 453, 497, 500
- Choice lists,Allow Choice Order Sort, 203
- Choice lists,automatically setting value, 239
- Choice lists,Automatically updating, 330
- Choice lists,creating relationships, 239
- Choice lists,custom reports, 427
- Choice lists,database tables, 497
- Choice lists,defined, 193
- Choice lists,editing, 201, 427
- Choice lists,field labels, 192
- Choice lists,global, 152, 500
- Choice lists,importing, 390
- Choice lists,Number of Visible Items attribute, 342
- Choice lists,of users and contacts, 453
- Choice lists,Progress field, 201, 427
- Choice lists,searching, 104
- Choice lists,showing selected items, 342
- Choice lists,sorting, 112, 203
- Choice lists,tooltips, 342
- Choice lists,updating in Web views, 330, 435
- Choice order sort, 104
- Client programs, 529
- Closed Date field, 204
- Closed Time field, 204
- Column Span attribute, 332, 435
- ComboBoxWidth CSS class, 333
- Compression, 407
- Contact field, 149, 435
- Contact tab, 137, 191
- contacts, 87, 137, 149, 153, 310, 388, 390, 453
- contacts,adding choice lists of, 453

contacts,assigning to work teams, 153

contacts,defined, 87

contacts,importing, 388, 390

contacts,submit-only views, 149, 310

Contains operator, 105

Cookies, 467

Copying HelpDesk, 530

Cross-tab reports, 127

Crystal Reports, 117

CSS classes, 333, 435, 447

CSS style sheets, 358, 468, 470, 473, 476, 479, 482, 486

Currency fields, 193

Current Issue - Detailed report, 427, 435

Current users, 419

, 126

CUSTOM REPORT ENGINE OPTION

Custom reports, 125, 427, 435

Custom reports,choice list changes, 427

Custom reports,deleted fields, 427

Custom reports,fields not exported, 435

Custom reports,for specific Web views, 125

Custom reports,multi-choice lists, 427

CustomCode.js, 337, 456, 457

CustomizedFiles folder, 125, 353, 435, 517

CustomizedFiles folder,backing up, 517

CustomizedFiles folder,explained, 353

CustomizedFiles folder,new projects, 435

Customizing Fields, 427

Data corruption, 371

databases, 84, 371, 373, 376, 380, 384, 419, 446, 495, 500, 507, 510, 512, 514, 519

Databases command, 363

databases,backing up, 507, 510, 514, 519

databases,compacting, 371

databases,DAT, 495

databases,definitions, 84

databases,described, 371

databases,editing tables, 446

databases,field definitions, 512

databases,importing issues, 384

databases,issue, 84

databases,locked, 380

databases,project definitions, 419

databases,releasing, 380

databases,removing locks, 373

databases,synchronizing issues with project definitions, 376

databases,users, 500

Date fields, 103, 193, 388

Date format, 359

- Default projects, 414
- Default values, 150, 352
- Definitions database, 419
- Description attribute, 313
- Destination Tab, 184, 191
- Detailed Record report, 427
- DHTML, 337
- Display All Read-Only Fields As Text attribute, 313, 334
- Display Read-Only Field As Text attribute, 334
- DNS Error, 381
- Domain user accounts, 350
- Double field size, 196
- Editor attribute, 328
- E-mail Conversation and Queue, 226
- employee account, 149
- Employee user account, 149
- Enable Auto Repair command, 377
- Enable Backups option, 519
- Encryption, 407
- Error log, 528
- Errors, 381, 382
- Errors, 404 File not Found message, 382
- Errors, Cannot find server or DNS Error message, 381
- Errors, The page cannot be displayed message, 381
- Escalation levels, 268
- Escalation rules, 269
- Escape sequences, 346
- Features, 171, 430
- Features, described, 430
- Features, enabling and disabling, 171
- Field Editor, 184
- Fields, 184, 186, 187, 188, 189, 190, 191, 192, 193, 196, 198, 199, 201, 239, 306, 313, 322, 325, 328, 329, 330, 331, 332, 333, 334, 336, 339, 341, 342, 345, 352, 427, 435, 455, 456, 512
- Fields, adding to issues, 186
- Fields, Aligning attribute, 329
- Fields, Alignment attribute, 329
- Fields, attributes, 325
- Fields, automatically setting value, 239
- Fields, captions, 192, 331, 427, 435
- Fields, changing size, 427
- Fields, changing type, 427
- Fields, choice lists, 193, 330, 435
- Fields, choice lists, automatically updating, 330
- Fields, choice lists, defined, 193
- Fields, choice lists, updating in Web views, 435
- Fields, Column Span attribute, 332
- Fields, CSS Class attribute, 333
- Fields, currency, 193
- Fields, custom reports, 435

- Fields,customizing, 427
- Fields,date, 193
- Fields,decimal numbers, 196
- Fields,default values, 352
- Fields,deleting, 198, 427
- Fields,Destination Tab, 184, 191
- Fields,disabled, 188
- Fields,Display Read-Only Field As Text attribute, 334
- Fields,editing, 427
- Fields,editing choice lists, 201
- Fields,Editor attribute, 328
- Fields,exporting to Web views, 322, 435
- Fields,form elements, 333
- Fields,hiding, 199, 345
- Fields,integer numbers, 196
- Fields,labels, 184, 331, 427
- Fields,layout in Web views, 435
- Fields,Layouts, 322
- Fields,Maintain Revision History, 190
- Fields,memo, 193, 334
- Fields,number, 193
- Fields,queries, 435
- Fields,read-only, 184, 188, 313, 334
- Fields,related, 239
- Fields,renaming, 427
- Fields,required, 187, 322
- Fields,security, 184, 188
- Fields,Show in Choice Lists, 189
- Fields,Show Selected Items in Text attribute, 342
- Fields,Show Selected Items In Tooltip attribute, 342
- Fields,size, 196
- Fields,sorts, 435
- Fields,spanning columns, 332
- Fields,tabs, 191, 427
- Fields,tblDtsFields, 512
- Fields,text, 193, 336, 341
- Fields,time, 193
- Fields,timestamping, 455, 456
- Fields,types of, 193
- Fields,updating styles, 427
- Fields,URL Button CSS attribute, 339
- Fields,URL buttons, 339
- Fields,Visible attribute, 199, 345
- Fields,Width attribute, 336
- Fields,yes/no, 193
- File uploads, 381
- Form elements, 333
- Form elements,CSS classes, 333
- Global choice lists, 152, 500
- Global databases, 363

- Group leader user account, 149
- group membership, 167
- Groups, 140, 164, 166, 171, 181, 414, 419, 430
- Groups command, 164
- Groups,adding users, 166
- Groups,Admin, 430
- Groups,creating, 164
- Groups,defined, 140
- Groups,deleting, 164
- Groups,feature permissions, 171
- Groups,Guests, 430
- Groups,project access permissions, 181, 414, 419, 430
- Groups,removing users, 166
- Groups,setting permissions, 430
- Groups,users, 430
- Groups,Web view access permissions, 430
- Guests group, 430
- HelpDesk, 523, 530
- HelpDesk Admin, 96, 201, 377
- HelpDesk project, 414
- HelpDesk Web Admin, 98, 201, 360
- HelpDesk,copying to another computer, 530
- HelpDesk,moving to another computer, 523
- HelpDeskServer folder, 514
- hiding in projects, 159
- Html code after control, 337, 455, 457
- Html code after control,defined, 337
- Html code after control,timestamping, 455, 457
- Html code before control, 337, 455, 457
- Html code before control,defined, 337
- Html code before control,timestamping, 457
- Import Issues, 384
- Importing users, 142
- Integer field size, 196
- Integrity command, 372, 530
- Inventory, 449
- Inventory,customizing reports list, 449
- issue database, 84, 374, 376, 384, 495, 510
- issue database,backing up, 510
- issue database,DAT, 495
- issue database,described, 84
- issue database,importing, 384
- issue database,synchronizing with project definitions, 374
- issue database,synchronizing with revision history, 376
- issue database,tblAttachments, 495
- issue database,tblDts, 495
- issue database,tblFixInformation, 495
- Issueagent user account, 295
- Issues, 125, 324, 368, 384, 388, 427, 435

Issues per Page attribute, 313

Issues,deleting, 368

Issues,importing, 384

Issues,linking, 324

Issues,numbers, 388

Issues,printing, 125, 427, 435

Issues,recycling numbers, 368

Javascript, 337

keyword searches, 105

Knowledge Base views, 310, 346

Knowledge Base views,described, 310

Knowledge Base views,URLs, 346

Labels, 184, 192

Labels,changing, 192

Labels,choice lists, 192

Labels,Field Caption, 184, 192

Layouts, 114, 322

Layouts,defining, 114

Layouts,exported fields, 322

Licenses, 379, 515

Licenses command, 378, 379

Licenses database, 515

Like operator, 107

Line charts, 127

Load Issues Not In Query attribute, 313

Locks, 373, 380, 530

Logging off users, 369, 524

Logon accounts, 137, 149

Logon sessions, 419

Logon.asp, 346, 354, 468

Logons Editor, 373

Logons Editor command, 369

Logos, 468, 486

Long integer field size, 196

Macros, 103, 127, 156, 246, 252, 257, 289, 462

Macros,<Any>, 257

Macros,<Empty>, 257

Macros,<New Issue>, 252

Macros,<None>, 103, 127

Macros,<Today>, 103

Macros,<User in Group>, 246, 252, 289

Macros,<User>, 103, 289

Macros,<Value of Field>, 156, 257

Macros,defining, 462

Mail server options, 295

MAPI, 295

Maximum Notification Retries option, 299

Memo fields, 193, 328, 334, 455, 456

Memo fields,defined, 193

Memo fields,timestamping, 455, 456

MemoFieldWidth CSS class, 332, 435

MIME types, 382

Monitor Interval, 371

Moving attachments, 366

Moving HelpDesk, 523

Mq Issue Agent, 295

Mq Issue Agent service, 380

Mq.Evaluator service, 275

Multi-choice lists, 193, 427

Network traffic, 371

New projects, 435

Normal Web views, 310

Not Like operator, 107

Notification - Detailed Record report, 427

Notification - Summary report, 427

Notification service, 295

Notifications, 190, 273, 285, 288, 289, 293, 295, 299, 380, 414, 427, 439, 459, 462

Notifications,attachments, 459

Notifications,bounces, 299

Notifications,contents, 462

Notifications,copying to new projects, 414

Notifications,e-mail contents, 288

Notifications,escalation rules, 273

Notifications,examples, 285

Notifications,field updates, 190

Notifications,file formats, 459

Notifications,formatting, 427

Notifications,Include macros, 462

Notifications,mail server options, 295

Notifications,maximum retries, 299

Notifications,Mq Issue Agent service, 380

Notifications,retries, 299

Notifications,send failures, 299

Notifications,setting up, 295

Notifications,sharing between projects, 414

Notifications,user account, 295

Notifications,when conditions, 289

Notifications,workflow, 293

Number fields, 193

Number of Characters attribute, 341

Number of Visible Items attribute, 342

, 126, 295, 299, 370, 371, 380, 439, 519

OPTIONS

Options,Always verify locked databases, 380

, 126

OPTIONS,CUSTOM REPORT ENGINE

Options,Enable Backups, 519

Options,Maximum Notification Retries, 299

Options,Monitor Interval, 371

Options,Send immediate shutdown message, 370

Options,Send Invalid Notifications To, 299

Options,Server Mail Interval, 295

Options,Server Mail Sender, 295

Options,Server Mail System, 295

Options,Server MAPI Profile, 295

Options,SMTP Server, 295

, 126

OPTIONS,STANDARD REPORT ENGINE

Owner field, 149, 388, 435

Pages, 130, 313

Pages,report, 130

Pages,Summary list, 313

Passwords, 161, 172, 295, 350

Performance, 371, 426, 435

Performance,number of projects, 426

Performance,optimizing, 371

Performance,updating choice lists, 435

Permissions, 524

Priority field, 104

Problem Area field, 104

Progress controlled by parent field, 324

Progress field, 201, 204, 239, 324, 388, 427, 497

Progress field,automating workflow, 239

Progress field,child issues, 324

Progress field,choice table, 427

Progress field,database table, 497

Progress field,editing, 201, 204

Progress field,importing, 388

Progress field,relationship with State, 204

Progress field,renaming, 427

Project definitions, 374, 419, 427, 453, 462, 497, 512

Project definitions,.def file, 497

Project definitions,backing up, 512

Project definitions,described, 419

Project definitions,macros, 462

Project definitions,Progress choice table, 427

Project definitions,synchronizing with issue database, 374

Project definitions,tables, 497

Project definitions,tblDtsFields, 497, 512

Project definitions,tblState, 497

Project definitions,tblSubstate, 427, 497

Project definitions,user choice lists, 453

Project folder, 510

projects, 84, 159, 181, 363, 368, 384, 412, 414, 416, 417, 418, 419, 426, 430, 435, 510, 517, 524

projects,access permissions, 426

projects,administering, 84

projects,base projects, 426

projects,creating, 414, 435

projects,current users, 419

- projects,customized files, 435
- projects,database engine, 419
- projects,databases, 363
- projects,defined, 412
- projects,definitions database, 84, 419
- projects,deleting, 418
- projects,deleting issues, 368
- projects,folders, 510, 517
- projects,Groups Allowed to Open, 181, 414, 419, 430
- projects,HelpDesk, 414
- projects,hiding users, 159
- projects,importing issues, 384
- projects,location, 419
- projects,one vs. many, 414
- projects,properties, 419
- projects,relocated, 524
- projects,styles and notifications, 416
- projects,templates, 417
- Properties of a project, 419
- Queries, 101, 103, 104, 105, 107, 156, 323, 435
- Queries,<None> macro, 105
- Queries,<Today> macro, 103
- Queries,<User> macro, 103
- Queries,Contains operator, 105
- Queries,dates, 103
- Queries,defining, 101
- Queries,exporting to Web views, 323
- Queries,keyword searches, 105
- Queries,relational operators, 104
- Queries,Revision Number field, 103
- Queries,rolling back, 435
- Queries,Web views, 435
- Queries,wildcards, 107
- Queries,work teams, 156
- RCviewer.asp, 405
- Read-only fields, 188, 313, 334
- Read-Only For, 184, 188
- Read-only Web views, 310
- Regional settings, 359
- Regular expressions, 107
- Relationships between issues, 324
- Relocate, 363, 366, 523
- Relocate,attachments, 366
- Relocate,Relocate command, 363, 366
- Relocation Wizard, 523
- Remote control, 403, 404, 405, 407, 409, 410
- Remote control,about, 403
- Remote control,adding button, 405
- Remote control,Client security keys, 409
- Remote control,compression, 407
- Remote control,encryption, 407

- Remote control,removing buttons, 410
- Remote control,requirements, 403
- Remote control,security key, 407
- Remote control,setting up, 404
- Removing locks, 530
- Repair and Compact command, 371
 - , 126
- REPORT CREATION API
- Reports, 117, 125, 127, 130, 322, 323, 427, 435
- Reports,adding fields, 427
- Reports,Current Issue - Detailed, 125, 427, 435
- Reports,customizing, 125
- Reports,Notification - Detailed Record, 427
- Reports,Notification - Summary, 427
- Reports,Pages, 130
- Reports,time, 127
- Reports,trend, 127
- Reports,viewing, 117
- Required fields, 322
- Revision history, 190, 376
- Revision history,enabling/disabling, 190
- Revision history,synchronizing, 376
- Revision Number field, 103, 285
- Security, 137, 140, 164, 171, 181, 184, 188, 313, 430
- Security keys, 403, 407, 409
- Security,fields, 184, 188
- Security,logon accounts, 137
- Security,permissions, 171, 181, 313, 430
- Security,permissions,discussed, 430
- Security,permissions,features, 171
- Security,permissions,projects, 181
- Security,permissions,Web views, 313
- Security,user groups, 140, 164, 181
- Send immediate shutdown message option, 370
- Send Invalid Notifications To option, 299
- Server Mail Interval option, 295
- Server Mail Sender option, 295
- Server Mail System option, 295
- Server MAPI Profile option, 295
- service level agreements, 265, 266, 267, 268, 269, 275, 276, 278, 280, 281, 282
- service level agreements,creating, 278
- service level agreements,defined, 265
- service level agreements,disabling, 280
- service level agreements,escalation levels, 268
- service level agreements,operational hours, 276
- service level agreements,reporting, 278, 281
- service level agreements,running programs, 275
- service level agreements,service types, 282
- service level agreements,setting up, 266

- service level agreements,target times, 267 , 126
- Service types, 282
- Services, 275, 295, 380
- Services,Mq.Evaluator, 275
- Sessions, 369
- Set Up Global Group command, 99
- Show Selected Items in Text attribute, 342
- Show Selected Items In Tooltip attribute, 342
- Shutdown Process, 380
- Single choice lists, 193
- Single field size, 196
- size, 196
- SMTP Server option, 295
- sorts, 111, 112, 203, 435
- sorts,choice order, 112, 203
- sorts,defining, 111
- sorts,exported fields, 435
- spanning columns, 332
- SpanTwoWidthOne CSS class, 332
- SQL Query Analyzer, 446
- SQL Server, 510, 512, 514
- SQL Server,DEF database, 512
- SQL Server,issue database, 510
- SQL Server,USERS_MDB database, 514
- STANDARD REPORT ENGINE OPTION
- State field, 204, 388
- Style sheets (CSS), 333, 468, 470, 473, 476, 479, 482, 486
- Styles, 412, 414, 427
- Styles,copying to new projects, 414
- Styles,defined, 412
- Styles,editing fields, 427
- Styles,sharing between projects, 414
- Submit-only views, 149, 310, 346, 467
- Submit-only views,contacts, 149, 467
- Submit-only views,cookies, 467
- Submit-only views,described, 310
- Submit-only views,logon accounts, 149
- Submit-only views,URLs, 346, 467
- Submitter field, 149, 257, 388, 435
- Summary List, 313
- Summary report, 427
- tabs, 191, 206, 427
- tabs,adding fields, 191
- tabs>Contact, 191
- tabs,editing, 206
- tabs,moving fields, 427
- tblAttachments, 386, 446, 495

tblDts, 384, 452, 495

tblDtsFields, 453, 497, 512

tblFixInformation, 495

tblRevisionHistory, 495

tblState, 497

tblSubstate, 427, 497

tblUser, 239, 390, 452, 500

templates, 417

Text area form elements, 193

Text fields, 193

Text reports, 127

TextFieldWidth CSS class, 333

Time fields, 193, 388

Time formats, 359

Timeout attribute, 313

Timesheet.js, 488

Timesheets, 488, 490, 491

Timesheets projects, 488, 490, 491

Timesheets,adding rows, 488

Timesheets,adding to Web views, 491

Timesheets,changing starting day, 490

timestamping, 455, 456

tmplInventoryTab.rec, 449

tmplRecord.html, 455

Tools, 371, 380

Tools menu, 97, 98, 149, 164, 303, 360, 363, 366, 369, 372, 377, 379, 519, 523

Tools menu,Attachments command, 366

Tools menu,Databases command, 363

Tools menu,Enable Auto Repair command, 377

Tools menu,Integrity Editor command, 372

Tools menu,Licenses command, 379

Tools menu,Logons Editor command, 369

Tools menu,Options command, 519

Tools menu,Relocate command, 363, 366

Tools menu,Repair and Compact command, 523

Tools menu,Web > Groups command, 164

Tools menu,Web > Users command, 149

Tools menu,Web command, 97, 98, 303, 360

Tools,Options, 371, 380

traffic, 371

Troubleshooting, 371, 373, 377

Troubleshooting,Admin problems, 377

Troubleshooting,performance, 371

Troubleshooting,removing locks, 373

Update test, 190, 285

Uploaded files, 381

URL attribute, 339

URL Button CSS, 339, 405, 447

URL buttons, 339

URLs for Web Views, 346, 467

User Categories, 157

Users, 137, 142, 149, 150, 152, 153, 157, 159, 160, 161, 163, 166, 167, 295, 310, 315, 350, 369, 373, 380, 388, 390, 422, 452, 453, 500, 514, 524

Users command, 149

Users database, 500

Users group, 315, 430

Users,adding choice lists of, 453

Users,adding to groups, 166

Users,admin account, 149

Users,analyst account, 149

Users,anonymous access account, 350

Users,assigning to work teams, 153

Users,authentication, 149, 163

Users,backing up database, 514

Users,categories, 157

Users,CensusUser account, 350

Users,database, 500

Users,default values, 150

Users,defined, 137

Users,deleting, 452

Users,disabling, 149, 160

Users,editing user information, 295

Users,employee account, 149

Users,group leader account, 149

Users,group membership, 167

Users,hiding in projects, 159

Users,importing, 142, 388, 390

Users,issueagent, 295

Users,logging off, 369, 380, 524

Users,logon accounts, 149

Users,MDB database, 514

Users,passwords, 161, 295

Users,removing from choice lists, 452

Users,removing from database, 452

Users,removing from groups, 166

Users,removing locks, 373

Users,roles and responsibilities, 422

Users,submit-only Web views, 310

Users,Web view permissions, 315

Users,Windows account, 350

Users,work teams, 152

Users.mdb, 500

USERS_MDB database, 500

USR database, 502

validatelogon.asp, 487

Vertical Alignment attribute, 329

Virtual directories, 348, 369, 381

Virtual directories,creating, 348

Virtual directories,defined, 348

Virtual directories,Logons Editor, 369

Virtual directories,re-creating, 381

Visible attribute, 199, 345

Web Admin command, 98, 360

Web View Editor, 97, 303

Web View Editor command, 97, 303

Web views, 84, 125, 149, 154, 172, 301, 303, 307, 308, 309, 310, 313, 315, 322, 323, 325, 330, 333, 334, 337, 345, 346, 348, 352, 353, 358, 359, 369, 430, 435, 468, 470, 473, 476, 479, 482, 486, 488, 490, 491, 517

Web views,access permissions, 315, 430

Web views,columns, 435

Web views,copying, 307

Web views,creating, 303

Web views,CSS, 358

Web views,custom reports, 125

Web views,CustomizedFiles folder, 125, 353, 435, 517

Web views,CustomizedFiles folder,backing up, 517

Web views,CustomizedFiles folder,custom reports, 125

Web views,CustomizedFiles folder,explained, 353

Web views,CustomizedFiles folder,from base project, 435

Web views,customizing, 337, 353, 468, 470, 473, 476, 479, 482, 486

Web views,date formats, 359

Web views,default values, 352

Web views,defined, 84, 301

Web views,deleting, 308

Web views,description, 313

Web views,disabling features, 172

Web views,Display All Read-Only Fields As Text attribute, 334

Web views,exporting fields, 435

Web views,exporting queries, 323

Web views,generating, 330, 435

Web views,groups allowed to open, 313, 430

Web views,hiding fields, 345

Web views,inserting custom code, 337

Web views,Knowledge Base, 310

Web views,layout of fields, 325, 435

Web views,layouts, 322

Web views,Load Issues Not In Query, 313

Web views,logging off users, 369

Web views,normal, 310

Web views,permissions, 172

Web views,printing issues, 435

Web views,queries, 435

Web views,read-only, 310

Web views,read-only fields, 334

Web views,regenerating, 435

Web views,renaming, 309

Web views,style sheets, 358, 468, 470, 473, 476, 479, 482, 486

- Web views,styling form elements, 333
- Web views,submit-only, 149, 310
- Web views,Summary List, 313
- Web views,time formats, 359
- Web views,timeouts, 313
- Web views,Timesheets, 488, 490, 491
- Web views,URLs, 346
- Web views,virtual directories, 348
- Web views,work teams, 154
- When conditions, 289
- Width attribute, 336
- Wildcards in queries, 107
- Windows account, 350
- Windows NT domains, 142
- Windows Server 2003, 381
- Work teams, 151, 152, 153, 154, 156
- Work teams,assigning users, 153
- Work teams,defining, 152
- Work teams,overview, 151
- Work teams,queries, 156
- Work teams,web views, 154
- Workflow, 151, 239, 246, 248, 249, 252, 253, 255, 256, 257, 260, 262, 263, 293, 422
- Workflow,<New Issue> macro, 252
- Workflow,applying changes to web views, 262
- Workflow,automating, 239
- Workflow,conditions, 248
- Workflow,dependent values rules, 239
- Workflow,disabling, 263
- Workflow,example rules, 239
- Workflow,macros, 257
- Workflow,notifications, 293
- Workflow,performance costs, 246
- Workflow,planning, 422
- Workflow,possible values rule, 239
- Workflow,projects, 246
- Workflow,related fields, 239
- Workflow,rules, 239, 248, 249, 253, 255, 256, 260, 263
- Workflow,rules,default names, 256
- Workflow,rules,defining, 253, 255
- Workflow,rules,deleting, 255
- Workflow,rules,described, 249
- Workflow,rules,disabling, 263
- Workflow,rules,evaluating, 260
- Workflow,rules,examples, 239
- Workflow,rules,format of, 248
- Workflow,rules,renaming, 256
- Workflow,Submitter field, 257
- Workflow,templates, 249, 256, 260, 263
- Workflow,templates,creating, 249
- Workflow,templates,described, 249

Workflow,templates,properties, 256, 260, 263

Workflow,test operators, 248

Workflow,user groups, 252

Workflow,work teams, 151

yes/no fields, 193