

FACIO processes

Quick Visual Tour

pdqm_info@pdqm.eu

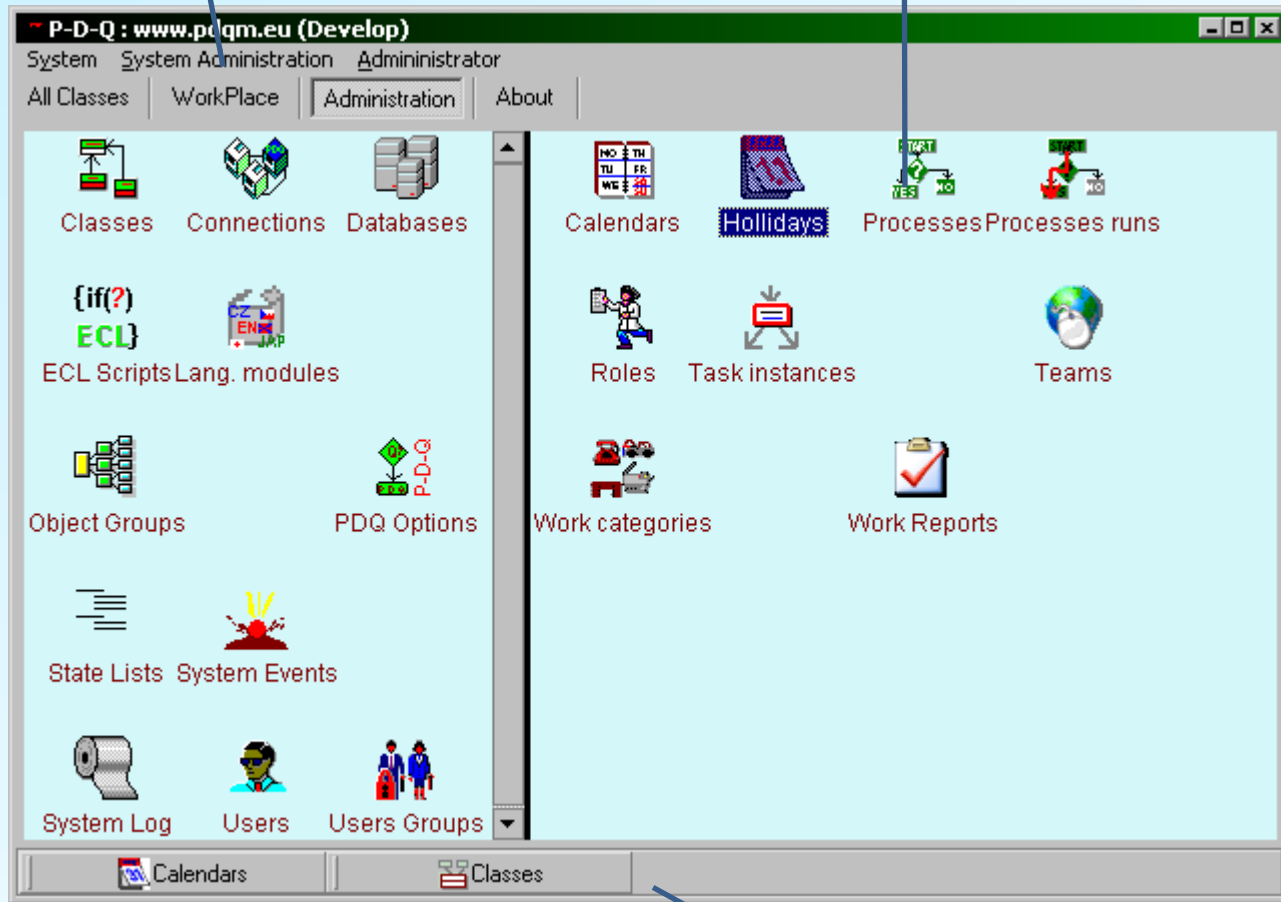
User interface

FACIO processes

Main application desktop

Personal working place

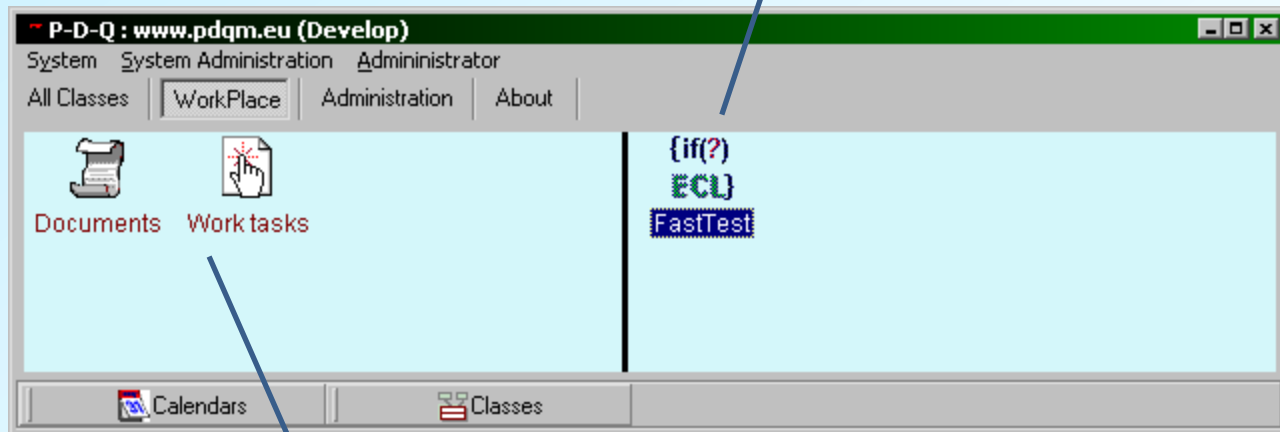
Icons for working objects



Line with open windows

Personal desktop

User can put objects on the desktop for fast access



User has access to the objects that relates with his/her work

Active tasks

Work organisation is easy with the filtering possibilities

User has overview about all active tasks

The screenshot shows a window titled "Work tasks" with a toolbar containing icons for adding, deleting, and filtering tasks. Below the toolbar is a table with columns for search, priority, name, assigned role, assigned team, and assigned user. The table contains eight rows of tasks, with four rows highlighted in red to indicate overdue tasks.

	nw	name	assignedrole	assignedteam	assigneduser
🔍	▲	Definujte kritéria pro rozhodování		TestTeam	
🔍	▲	Vyhodnotte alternativy definovanými met		TestTeam	
🔍	▲	Definujte měřtko kritérií		TestTeam	
🔍	▲	Definujte měřtko kritérií		TestTeam	
🔍	▲	Vytvořte metody vyhodnocení kritérií		TestTeam	
🔍	▲	Definujte kritéria pro rozhodování		TestTeam	
🔍	▲	Revidujte přípravu pro rozhodování		TestTeam	
🔍	▲	Definujte kritéria pro rozhodování		TestTeam	

Tasks has defined priority and actual importance

Overdue tasks are marked red

Task detail

Detail of task includes all details

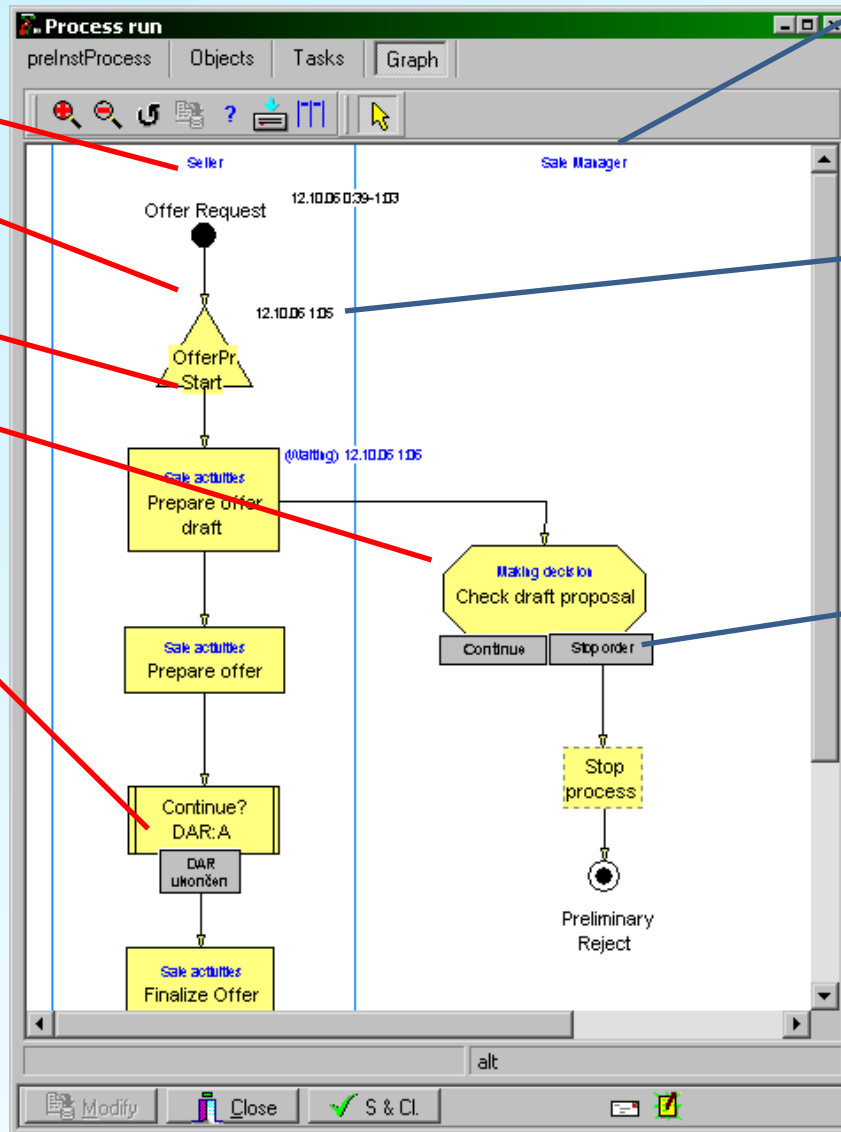
Access to the process overview for fast review

Quick access to the all related objects

Buttons to finish the task

Button for sending the object to the colleague

Process execution overview



Process start

Milestone for process monitoring

User task

User decision

Subprocess

Swimlanes for process diagram organization

Information about the task execution date and time

Variant end states



Developer and administrator interface

FACIO processes

Business object definition

Developer defines object according to the business

System creates tables including indexes and constraints

```
Execute command: CREATE TABLE saleCustomer (  
  IsaleCustomerId integer not null primary key,  
  Address varchar (200),  
  City varchar (100),  
  Name varchar (100) NOT NULL,  
  Prominence smallint,  
  Region smallint NOT NULL)  
ALTER TABLE saleCustomer ADD CONSTRAINT c_saleCustomer_obj FOREIGN KEY (IsaleCustomerId) REFERENCES  
SYSOBJECT(LSYSOBJECTID) ON DELETE CASCADE  
create index I_saleCustomer_2 on saleCustomer (Name)  
?
```

Yes No Cancel

Name	Type	Value
Address	string	
City	string	
Name	string	
Offers	Link saleOffer[]	
Prominence	ItemListVal saleCustProminence	
Region	ItemListVal saleRegions	

Attr. Name: Prominence
Type: itemListval
Itemized List: saleCustProminence
Save History: Not save History
Visibility: Both list and det
Feature: NotNull Indexed Calculated
Array: Read Only Unique

Def Value:
Crypt Key:
Description:

Items	No Access	Read Only	Read Write
Region	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Prominence	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Offers	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Name	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
City	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Methods: No Exec Can Exec

OpenList	<input checked="" type="radio"/> No Exec <input type="radio"/> Can Exec
OpenDetail	<input checked="" type="radio"/> No Exec <input type="radio"/> Can Exec
Delete	<input checked="" type="radio"/> No Exec <input type="radio"/> Can Exec
Create	<input checked="" type="radio"/> No Exec <input type="radio"/> Can Exec

Rights are defined for each field and function

Business object editor

Automatically generated dialog can be modified by the developer

Debugger makes development as simple as possible

saleCustomer

Address

City

Name

Offers

Prominence

Region

OK Cancel

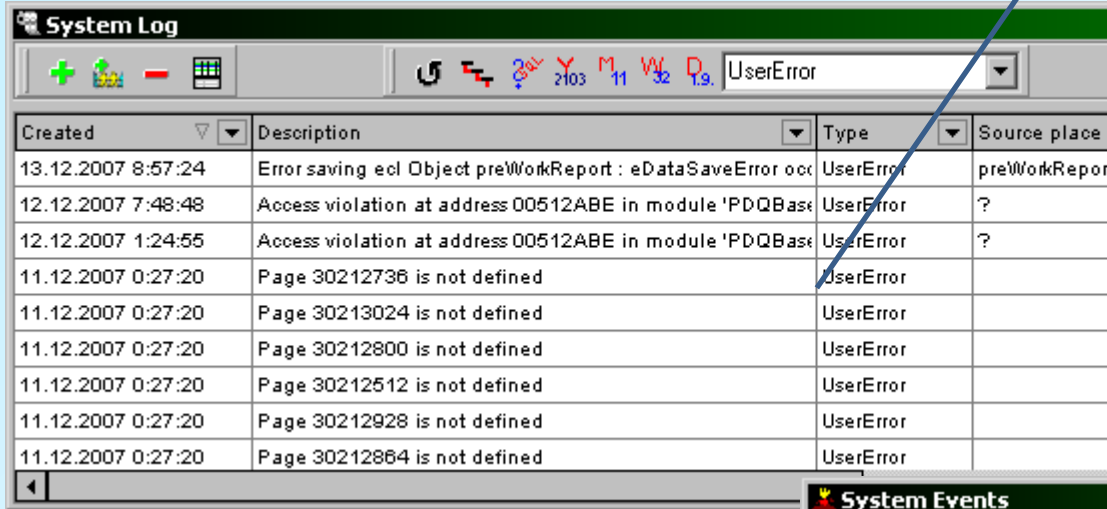
```
string Floa
//Returns
//Options
integer Get
//Returns
integer Get
//Returns
itemlistval
//Returns
string IntT
//Returns
boolean IsC
//Returns
integer It
//Returns
string Item
//Returns name of the Item field defined by list name and ider
integer ItemNameToIdent (string listname; string itemname)
//Returns ItemIdent code as and integer value of the Item from
integer ItemParamInt (itemlistval freeitemlist itemval)
//Returns integer parameter of the itemized list
string ItemParamS (itemlistval freeitemlist itemval)
//Returns String parameter of the itemized list
link sysobject RetypeLink (link sysobject variable)
//Allows changing the object link type
date StrToDate (string strdate)
//Converts string to date
```

```
void (link preInstProcess PrInstance; link preInstTask InstTask; link preInstLink InLink)
1 link preInstWorkAssign IWA := preInstWorkAssign.create;
2 IWA.WorkCategory := WorkCategory;
3 IWA.Priority := PrInstance.Process.Priority;
4 IWA.AssignedTeam := PrInstance.Team;
5 IWA.Name := InstTask.Task.Name;
6 IWA.InstTask := InstTask;
7 // The state of the work is set according to the process state
8 IWA.WRRequired := WRRequired(PrInstance, InstTask, InLink);
9 if (PrInstance.state = 'Beyond Warning') IWA.WorkState := 'Beyond Warning'
10 else if (PrInstance.state = 'Beyond Hard Limit') IWA.WorkState := 'Beyond Hard Limit'
11 else IWA.WorkState := 'Active';
12
13 InstTask.State := 'Waiting'; // Execute should always change state of the task to some c
14 InstTask.WaitingFor := IWA;
15 if (!IsNull(WorkCategory.Calendar)) {
16     timeinterval lint := ExpectedDuration;
17     if (lint > 0)
18         IWA.dtExpectedEnd := WorkCategory.Calendar.GetEndOfWI(CurrentDateTime, lint);
19 };
20
```

Classes are described in C++ like object build-in language

Function help increases productivity

Administration

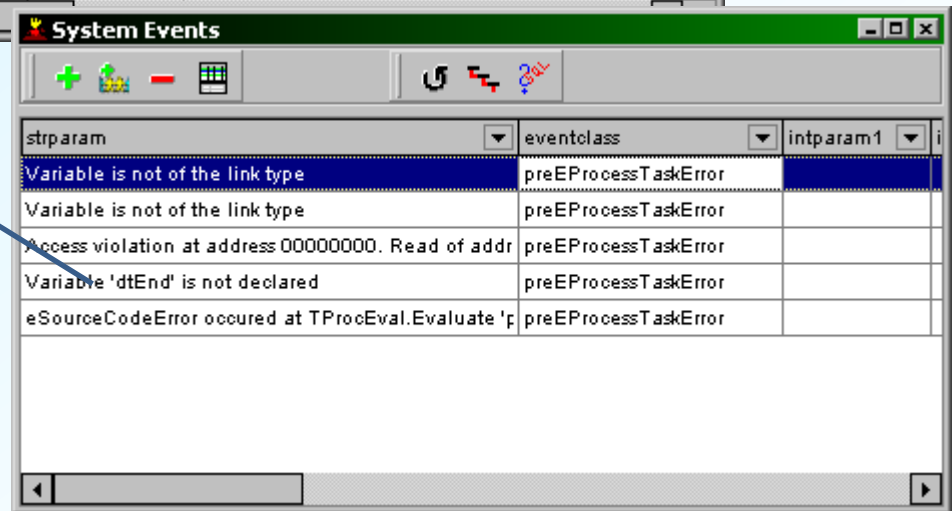


The screenshot shows the 'System Log' window with a search filter set to 'UserError'. The table below represents the data shown in the log.

Created	Description	Type	Source place
13.12.2007 8:57:24	Error saving ecd Object preWorkReport : eDataSaveError occ	UserError	preWorkReport
12.12.2007 7:48:48	Access violation at address 00512ABE in module 'PDQBase	UserError	?
12.12.2007 1:24:55	Access violation at address 00512ABE in module 'PDQBase	UserError	?
11.12.2007 0:27:20	Page 30212736 is not defined	UserError	
11.12.2007 0:27:20	Page 30213024 is not defined	UserError	
11.12.2007 0:27:20	Page 30212800 is not defined	UserError	
11.12.2007 0:27:20	Page 30212512 is not defined	UserError	
11.12.2007 0:27:20	Page 30212928 is not defined	UserError	
11.12.2007 0:27:20	Page 30212864 is not defined	UserError	

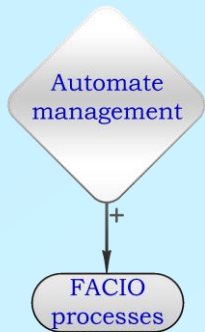
System log reports all important incident and occurrences important for administration

Event generated by the scripts allows both debugging and system monitoring



The screenshot shows the 'System Events' window with a list of error events. The table below represents the data shown in the events log.

strparam	eventclass	intparam1	i
Variable is not of the link type	preEProcessTaskError		
Variable is not of the link type	preEProcessTaskError		
Access violation at address 00000000. Read of addr	preEProcessTaskError		
Variable 'dtEnd' is not declared	preEProcessTaskError		
eSourceCodeError occurred at TProcEval.Evaluate 'p	preEProcessTaskError		



Summary

- Fast business domain description
- Good database management procedures
- Open system prepared for iterative development
- Standard modules for email, communication and file management
- Auditable process execution and data modification
- Strong security rules
- Native window-based user interface – users can work without system training

