

TIBCO iProcess® Server Objects (.NET)

Object Model

Table of Contents



sBase..... 1	sWorkQ / xWorkQ..... 12-14	vCriteriaRequest
vNode	vAuditMsgDef	vACaseCriteria
vNodeId	vAWorkItem	vPredictionCriteria
sSession / xSession..... 2	vAWorkItemListState	vSortField
vNodeCtx	vCDQP	vWICriteria
Exceptions..... 3	vForm	vSummary
vException	vWICriteria	vWISummary
vExceptionDetail	vWIFieldGroup	vWILocked
vEx<ObjectClass>	vWorkItem	sIPEConfig / xIPEConfig..... 24
sNode / xNode..... 4-7	vWorkItemListState	vDatabaseConfig
vANode	vWorkQ	sClientLog..... 25-26
vActiveUser	vWorkQDeltaJMSId	vClientLog
vClientInfo	vWorkQId	sProcManager / xProcManager..... 27-35
vConfigInfo	sWorkQManager / xWorkQManager..... 15-17	vAccessUserRef
vNodeLocale	vAWorkQ	vAction
vSALInfo	vAWorkQListState	vAddressUserRef
vThreadInfo	vCDQPDef	vAProc
vWQSessionInfo	vDate	vCommand
vAGroup	vDateTime	vConditional
vAGroupListState	vParticipation	vDeadline
vAttribute	vRedirection	vDeadlineValue
vAttributeDef	vTime	vDuration
vListValidation	vWorkQ	vDurationValue
vNode	vWorkQListState	vFieldDef
vNodeId	vWorkQId	vFMarking
vOSUser	vWorkQIdListState	vForm
vOSUserListState	sCaseManager / xCaseManager..... 18-20	vAFMarking
vPreference	vACase	vFConditional
vRole	vACaseListState	vFMarking
vTable	vAuditStep	vFont
vTableField	vCase	vFRow
vUser	vCaselId	vLabel
vUserListState	vField	vPermission
vWorkQId	vGraftSubTask	vPluginForm
vWorkQIdListState	vOutstandingItem	vPredictedItem
sNodeManager / xNodeManager..... 8-9	vDynamicSubProcItem	vPriority
vANode	vEAllItem	vProc
vActiveUser	vEventItem	vProcAudit
vClientInfo	vExternalGraftProcess	vProcid
vConfigInfo	vGraftItem	vProcDef
vNodeLocale	vNormalItem	vProcSummary
vSALInfo	vSubProcCallItem	vPublicStep
vThreadInfo	vSubProcCase	vPublicField
vWQSessionInfo	vTransactionControlItem	vStepId
vNode	vPredictedItem	vAutoStep
vNodeId	vPredictedItemListState	vComplexRouterStep
sUser / xUser..... 10-11	vProcid	vDynamicSubProcCallStep
vAttribute	sPageableList / sPageableListR /	vEAllStep
vAuditMsgDef	sPageableListJ..... 21-23	vEISStep
vGroup	vContentRequest	vEventStep
vPreference	vACaseContent	vGraftStep
vProc	vAGroupContent	vNormalStep
vProcid	vAProcContent	vOpenClientStep
vUserId	vAWIContent	vScriptStep
vWorkQId	vAWorkQContent	vSubProcCallStep
vWorkQ	vGroupContent	vTransactionControlStep
vWorkItem	vOutstandingItemContent	Custom Types 1..... 36
	vProcDefContent	Custom Type 2..... 37
	vStepContent	Index..... 38
	vUserContent	
	vWICContent	
	vWIFGContent	

This Server Object, which is inherited by the Server Objects that start user sessions (that includes all Server Objects except sNodeManager, sPageableList, and sClientLog), provides a Dispose method that should be called to end the session when the Server Object is no longer needed.

sBase					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
GetNodeId	vNodeId (p1)		X		
GetNode	vNode (p1)		X		
Disconnect	void	LO	X		
SessionId	int		X		
Dispose	void		X		
Log	void		X		
ClientMsgInterfaceVer	String		X		
ServerMsgInterfaceVer	String		X		

vNode	
vNodeId	
Name	
ComputerName	
IPAddress	
TCPPort	
IsDirector	
Status	
SEOSrvDesc	
SEOSrvName	
SEOSrvVersion	
SWVersion	
ClientCnt	
ClusterId	
Instance	

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.

This Server Object provides the ability to create multiple Server Objects (sNode, sUser, etc.) that all share the same user session. This allows you to share the TCP connection among multiple Server Objects, rather than have a TCP connection for every Server Object that is created.

sSession					
Extends sBase					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
Create_sNode	sNode (p4)		X		
Create_sUser	sUser (p10)		X		
Create_sWorkQ	sWorkQ (p12)		X		
Create_sWorkQManager	sWorkQManager (p15)		X		
Create_sProcManager	sProcManager (p27)		X		
Create_sCaseManager	sCaseManager (p18)		X		
Create_sIPEConfig	sIPEConfig (p24)		X		
KeepAliveWorkQDeltaJMSPublications	void	DKA	X		



This XML Server Object, which is only available in the XML interface, is used to construct all other XML Server Objects (xNode, xUser, etc.).

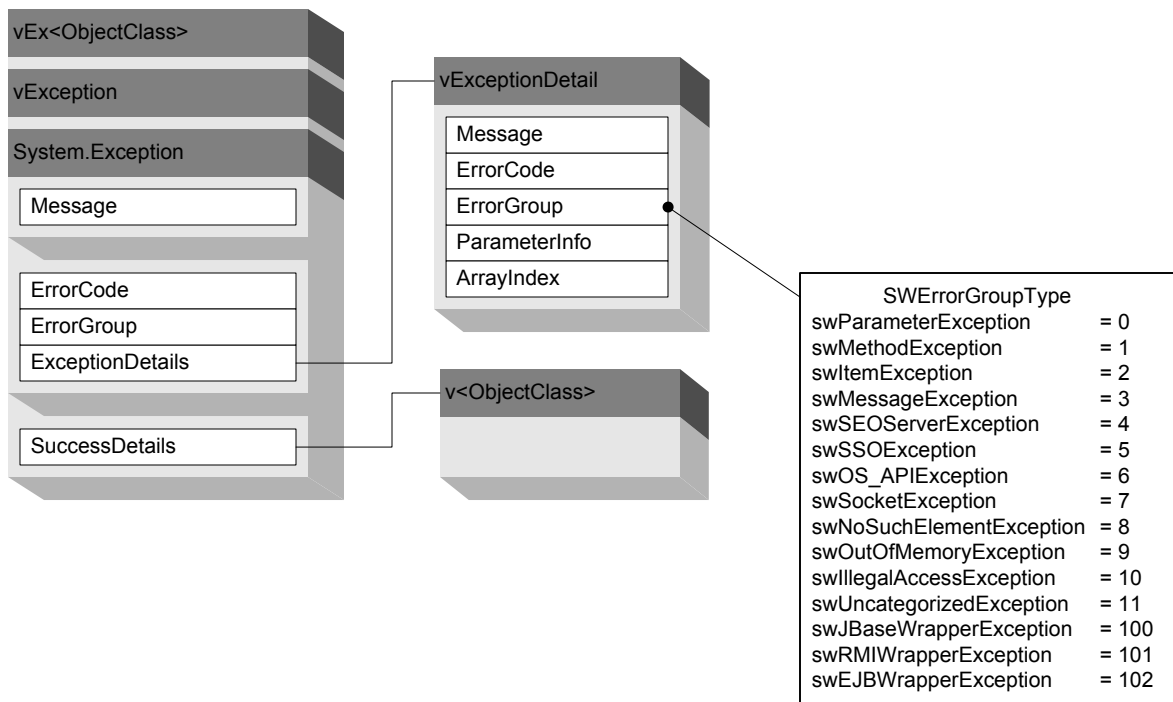
xSession					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
Create_xNode	xNode (p4)			X	
Create_xUser	xUser (p10)			X	
Create_xWorkQ	xWorkQ (p12)			X	
Create_xWorkQManager	xWorkQManager (p15)			X	
Create_xProcManager	xProcManager (p27)			X	
Create_xCaseManager	xCaseManager (p18)			X	
Create_xIPEConfig	xIPEConfig (p24)			X	
GetXMLResults	void			X	
KeepAliveWorkQDeltaJMSPublications	void	DKA		X	
ssoSerializeXML	void			X	
ssoDeserializeXML	Object			X	
Disconnect	void			X	

The vNodeCtx object is used when constructing the sSession and xSession objects.

vNodeCtx	
NodeId	
UserName	
Password	

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.


















This Server Object provides access to information stored on a node (a TIBCO iProcess Engine), such as users, groups, attributes, roles, etc.



Use the sNode object if programming to the base .NET interface; use the xNode object if programming to the XML interface.

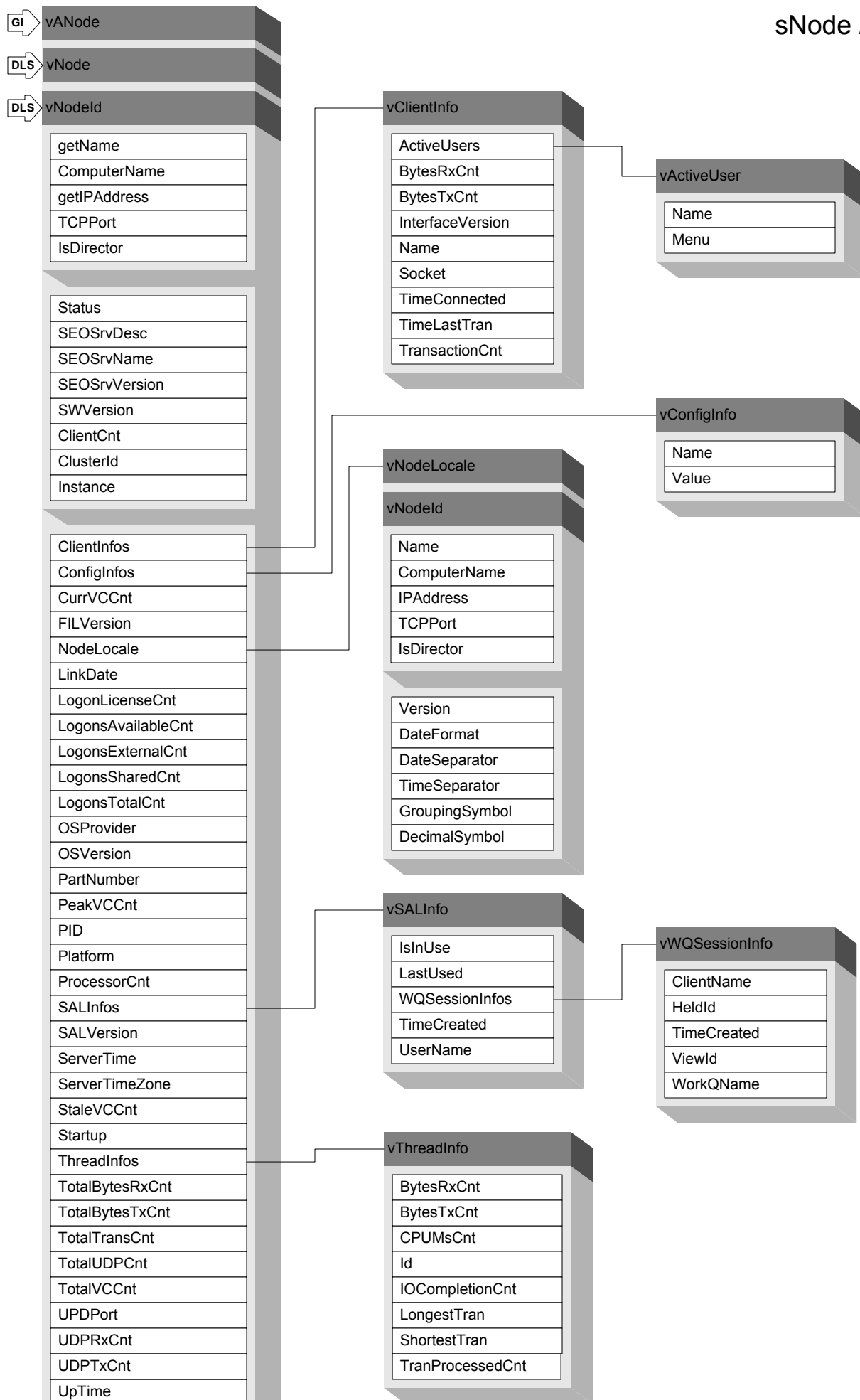
sNode / xNode

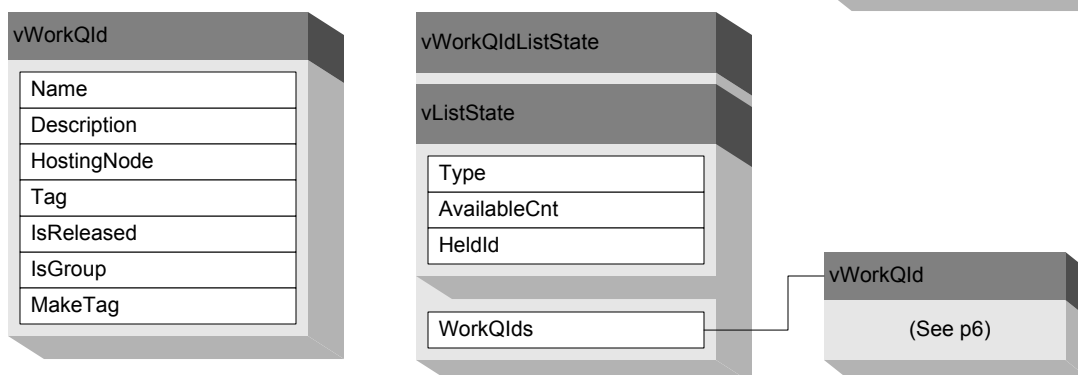
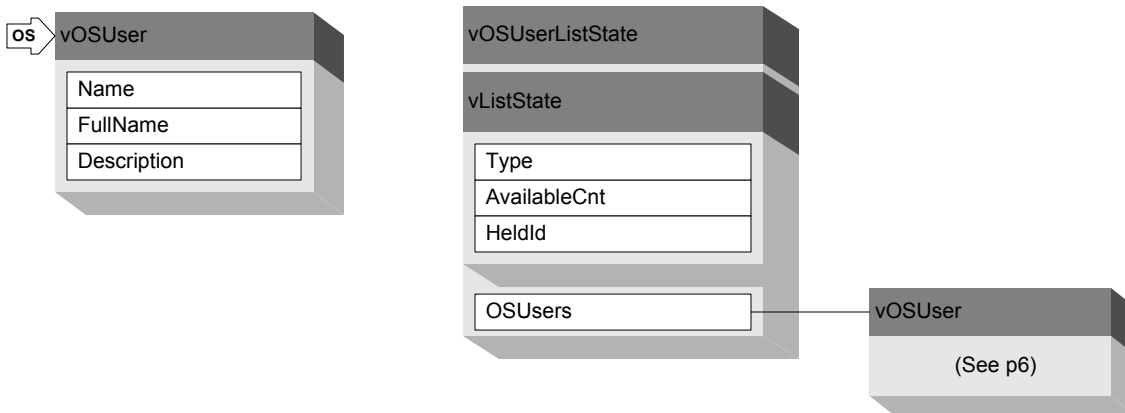
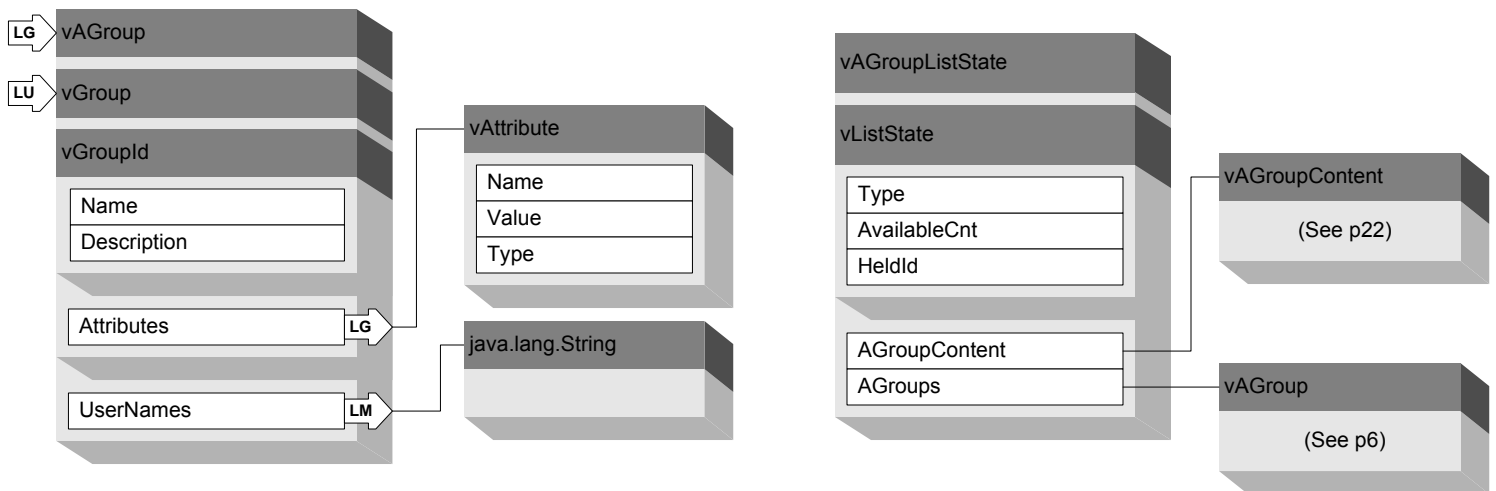
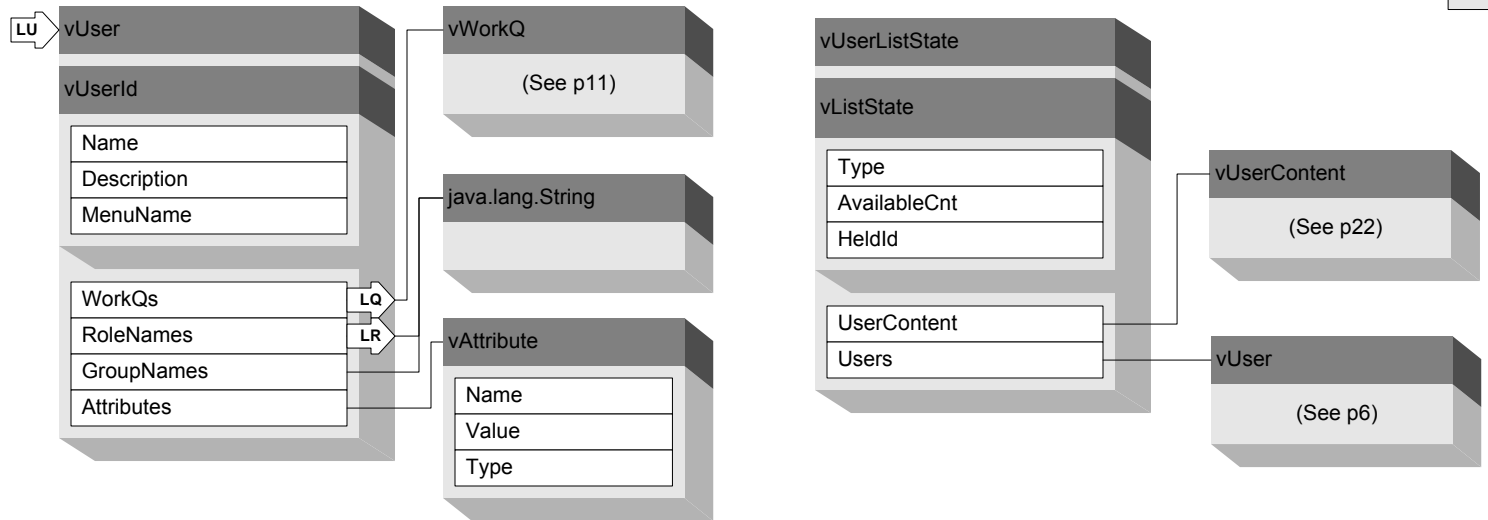
sNode extends sBase

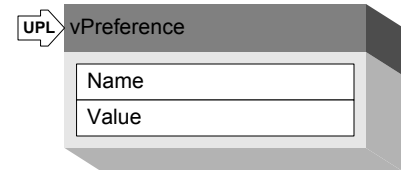
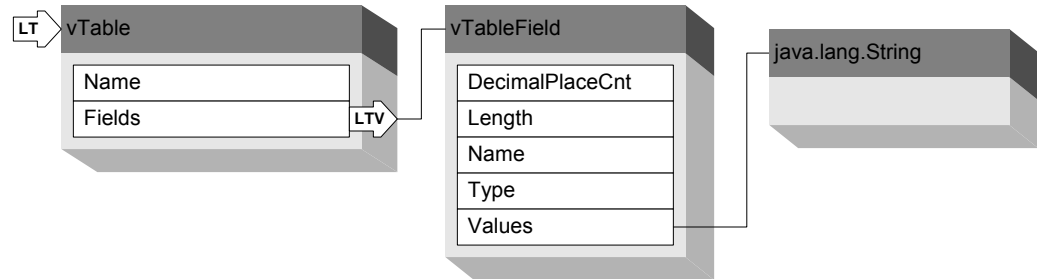
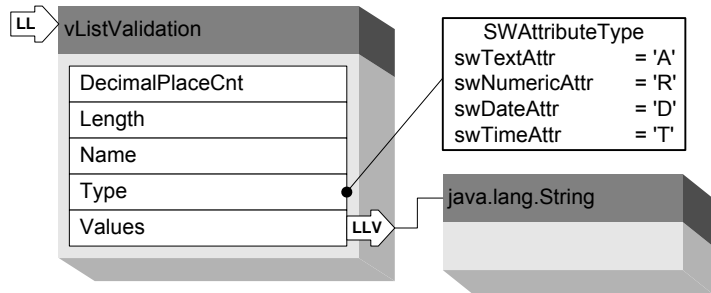
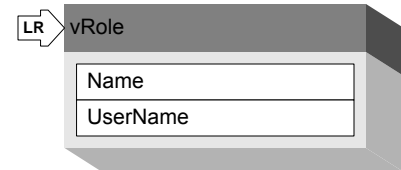
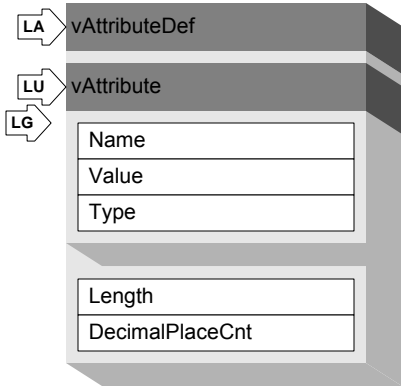
Methods	Returns *	Messages	Available In Interfaces:	
			Base	XML
GetANode	vANode (p5)	DLS, GNL, GI	X	X
GetUserAttributes	vAttribute [] (p6)	LU 	X	X
GetGroupAttributes	vAttribute [] (p6)	LG 	X	X
ChangeAttributes	void	CA 	X	X
GetAttributeDefs	vAttributeDef [] (p7)	LA 	X	X
CreateAttributeDefs	void	AA 	X	X
DeleteAttributeDefs	void	RA	X	X
GetAGroups	vAGroup [] (p6)	LG,LM 	X	X
GetAGroupList	sPageableList – vAGroup (p21/6)	LGW,LGR,(LG),LM	X	
GetAGroupListHeld	sPageableList – vAGroup (p21/6)	LGW,LGR,(LG)	X	
MakeAGroupList	vAGroupListState (p6)		X	X
FetchAGroupList	vAGroupListState (p6)		X	X
CreateGroups	void	AG 	X	X
DeleteGroups	void	RG	X	X
AddUsersToGroups	void	AM 	X	X
RemoveUsersFromGroups	void	RM 	X	X
GetOSUsers	vOSUser [] (p6)	OS 	X	X
GetOSUserList	sPageableList – vOSUser (p21/6)	OSW,OSR,(OS)	X	
GetOSUserListHeld	sPageableList – vOSUser (p21/6)	OSW,OSR,(OS)	X	
MakeOSUserList	vOSUserListState (p6)		X	X
FetchOSUserList	vOSUserListState (p6)		X	X
GetUsers	vUser [] (p6)	LU,LQ,LR 	X	X
GetUserList	sPageableList – vUser (p21/6)	UNW,UNR,(UN),LQ,LR	X	
GetUserListHeld	sPageableList – vUser (p21/6)	UNW,UNR,(UN),LQ,LR	X	
MakeUserList	vUserListState (p6)		X	X
FetchUserList	vUserListState (p6)		X	X
CreateUser	void	AU 	X	X
DeleteUsers	void	RU	X	X
GetRoles	vRole [] (p7)	LR 	X	X
CreateRoles	void	AR 	X	X
DeleteRoles	void	RR	X	X
GetTables	vTable [] (p7)	LT,LTV 	X	X
GetListValidations	vListValidation [] (p7)	LT,LTV	X	X
GetWorkQIdList	sPageableList – vWorkQId (p21/6)	QXW,QXR,(QX)	X	
GetWorkQIdListHeld	sPageableList – vWorkQId (p21/6)	QXW,QXR,(QX)	X	
MakeWorkQIdList	vWorkQIdListState (p6)		X	X
FetchWorkQIdList	vWorkQIdListState (p6)		X	X
MoveSysInfo	void	MSI	X	X
SetUserPreference	void	UPS	X	X
GetUserPreference	vPreference (p7)	UPL	X	X
DeleteUserPreference	void	UPD	X	X

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses) For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.









This Server Object is used for TIBCO iProcess Objects Server / Director discovery, setting TIBCO iProcess Objects Server/Director discovery options, and setting TIBCO iProcess Objects Server Log options.

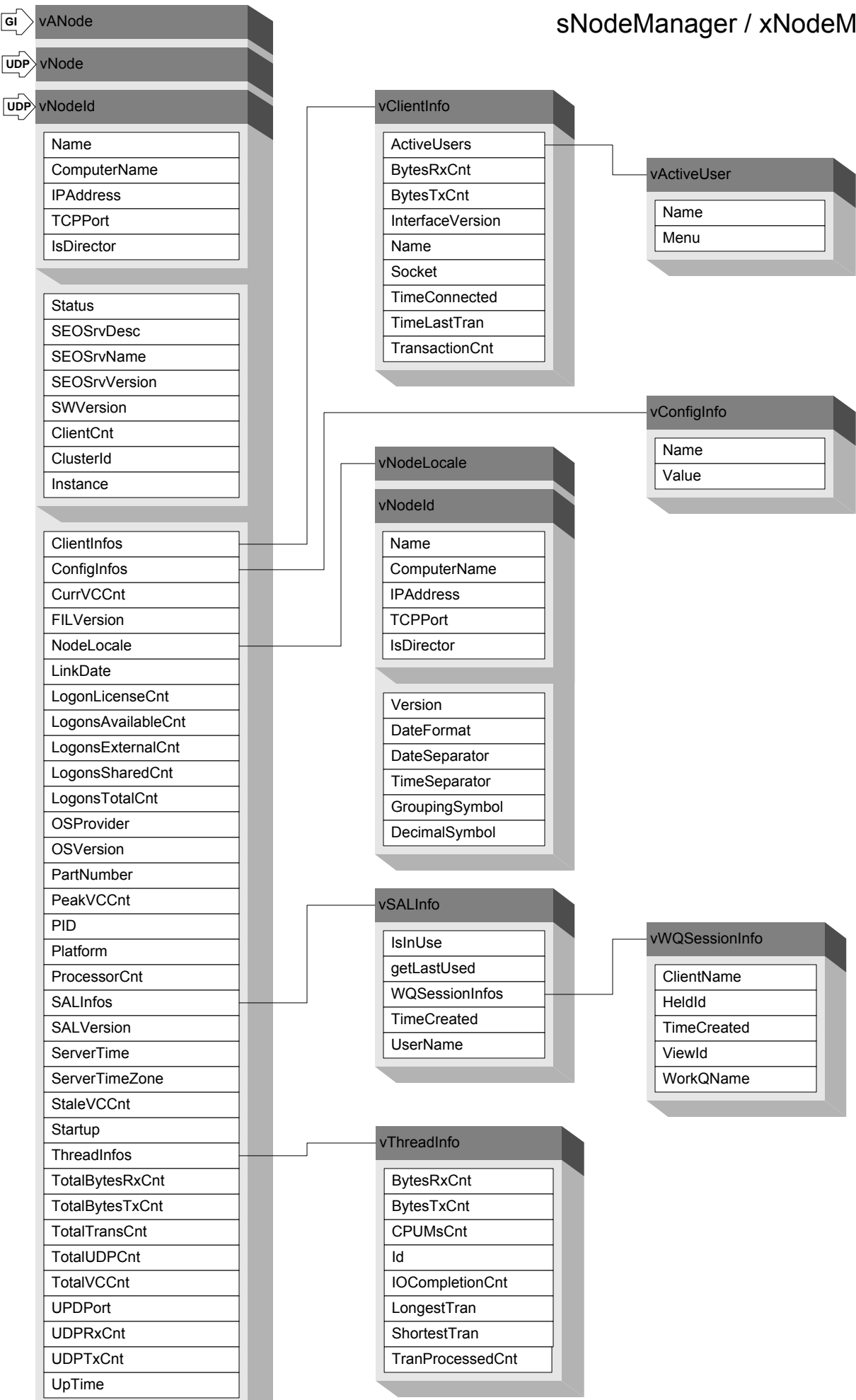
Use the sNodeManager object if programming to the base .NET interface; use the xNodeManager object if programming to the XML interface.

sNodeManager / xNodeManager					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
UDPPortNumbers	int []		X		
Refresh	void	UDP	X		
VerifyNode	vNode (p9)	UDP	X	X	
GetNodes	vNode [] (p9)	UDP	X	X	
GetANode	vANode (p9)	UDP	X	X	
GetManagedNodes	vNodeId [] (p9)		X		
SetSrvLogOptions	void	LOG	X	X	
ResetSrvLog	void	LOG	X	X	
GetXMLResults	void				X

* On xNodeManager, all methods return void; you must call xNodeManager.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.






Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.





This Server Object provides information that is relevant to a single user. It provides access to the user's work queue and work item information, as well as methods to start a case or add an entry to the audit trail.

Use the sUser object if programming to the base .NET interface; use the xUser object if programming to the XML interface.

sUser / xUser					
sUser extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
StartCase	string	SC,SC2	X	X	X
GetAuditProclDs	vProclD[] (p11)	AP	X	X	X
GetStartProclDs	vProclD[] (p11)	LS	X	X	X
GetWorkItems	vWorkItem[] (p11)	QIS 	X	X	X
GetUserId	vUserId (p11)		X	X	X
GetProcs	vProc[] (p11)	LP 	X	X	X
GetSupervisedQlds	vWorkQld[] (p11)	LAQ	X	X	X
GetWorkQlds	vWorkQld[] (p11)	LQ	X	X	X
GetWorkQs	vWorkQ[] (p11)	LQ 	X	X	X
GetAttributes	vAttribute[] (p11)	LU 	X	X	X
GetGroups	vGroup[] (p11)	LU, LG 	X	X	X
GetRoleNames	string[]	LR	X	X	X
Create_sWorkQ	sWorkQ (p12)		X		
ChangePassword	void	SP	X	X	X
IsPasswordExpired	bool	EP	X	X	X
AddCaseAuditEntry	void	AAE	X	X	X
GetCustomAuditMsgDefs	vAuditMsgDef [] (p11)		X	X	X
SetUserPreference	void	UPS	X	X	X
GetUserPreference	vPreference (p11)	UPL	X	X	X
DeleteUserPreference	void	UPD	X	X	X

* On xUser, all methods return void; you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.

vUserId	
Name	
Description	
MenuName	

vAttribute	
Name	
Value	
Type	

vGroup	
vGroupId	
Name	
Description	
Attributes	LG

vProc	
vProcId	
Name	
Description	
HostingNode	
MajorVersion	
MinorVersion	
Tag	
MakeTag	
ProcNumber	
StartStepName	
Status	
CaseDescOpt	

vWorkQ	
vWorkQId	
Name	
Description	
HostingNode	
Tag	
IsReleased	
IsGroup	
MakeTag	
FirstDeadline	
DeadlineCnt	
UnopenedCnt	
UrgentCnt	
WorkItemCnt	
WorkQParam1Name	
WorkQParam2Name	
WorkQParam3Name	
WorkQParam4Name	

vAuditMsgDef	
MsgNumber	
MsgTemplate	












vPreference	
Name	
Value	

vWorkItem	
vWorkItemId	
vCase	
(See p19)	
Tag	
MailId	
CaseTag	
MakeTag	
AddrToName	
Arrived	
Deadline	
DeltaStatus	
IsDeadline	
IsDeadlineExp	
IsKeepOnWithdrawal	
IsForwardable	
IsLocked	
IsLongLocked	
IsOrphaned	
IsReleasable	
IsUnopened	
IsUrgent	
IsSuspended	
LockedBy	
Priority	
StepName	
StepDescription	
CDQPs	
WorkQParam1	
WorkQParam2	
WorkQParam3	
WorkQParam4	

This Server Object provides information and functionality relevant to a single work queue. It allows you to:

- ▶ obtain the work items in the work queue (GetWorkItems, GetWorkItemList, etc.)
- ▶ process work items in the work queue (LockItems, ReleaseItems, etc.)
- ▶ start a new instance of a procedure (StartCase)
- ▶ add items to a case's audit list (AddCaseAuditEntry)


This object does not provide methods to configure the work queue; use the sWorkQManager object to configure a work queue. Use the sWorkQ object if programming to the base .NET interface; use the xWorkQ object if programming to the XML interface.


sWorkQ / xWorkQ					
sWorkQ Extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
GetWorkQ	vWorkQ (p14)	LQ	X	X	
GetWorkItems	vWorkItem[] (p13)	QIS 	X	X	
GetWorkItemList	sPageableListR - vWorkItem (p21/13)	QQW, QQR	X		
GetWorkItemListHeld	sPageableListR - vWorkItem (p21/13)	QQW, QQR	X		
MakeWorkItemList	vWorkItemListState (p13)		X	X	
FetchWorkItemList	vWorkItemListState (p13)		X	X	
FetchWorkItemListDelta	vWorkItemListState (p13)		X	X	
FetchWorkItemListIfChanged	vWorkItemListState (p13)		X	X	
GetAWorkItems	vAWorkItem[] (p13)	QIS 	X	X	
GetAWorkItemList	sPageableListR - vAWorkItem (p21/13)	QQW, QQR	X		
GetAWorkItemListHeld	sPageableListR - vAWorkItem (p21/13)	QQW, QQR	X		
GetAWorkItemListJMS	sPageableListJ	QDW, QDR	X		
GetAWorkItemListJMSHeld	sPageableListJ	QDW, QDR	X		
MakeAWorkItemList	vAWorkItemListState (p13)		X	X	
MakeAWorkItemListJMS	**	QDW	X	X	
FetchAWorkItemList	vAWorkItemListState (p13)		X	X	
FetchAWorkItemListJMS	vAWorkItemListState (p13)		X	X	
FetchAWorkItemListDelta	vAWorkItemListState (p13)		X	X	
FetchAWorkItemListIfChanged	vAWorkItemListState (p13)		X	X	
StartCase	string	SC	X	X	
StartWorkQDeltaJMSPublish	vAWorkItemListState (p13)	QDS	X	X	
LockItems	vWIFieldGroup[] (p13)	LI 	X	X	
LockFirstWorkItem	vWILocked (p23)	LFI 	X	X	
LockFirstAWorkItem	vWILocked (p23)	LFI 	X	X	
KeepItems	void	KR 	X	X	
ReleaseItems	void	KR 	X	X	
UndoItems	void	KR 	X	X	
UnlockItems	void	KR 	X	X	
GetForwardToWorkQIds	vWorkQId[] (p14)	FL	X	X	
ForwardItems	void	FI 	X	X	
GetDefaultCriteria	vWICriteria (p14)	LQ	X	X	
ChangeDefaultCriteria	void	QC	X	X	
ClearDefaultCriteria	void	DC	X	X	
GetForm	vForm (p14)	PF, ME 	X	X	
AddCaseAuditEntry	void	AAE	X	X	
GetCustomAuditMsgDefs	vAuditMsgDef[] (p14)	AMT	X	X	

Key

(pXX) Page in this document showing returned object.

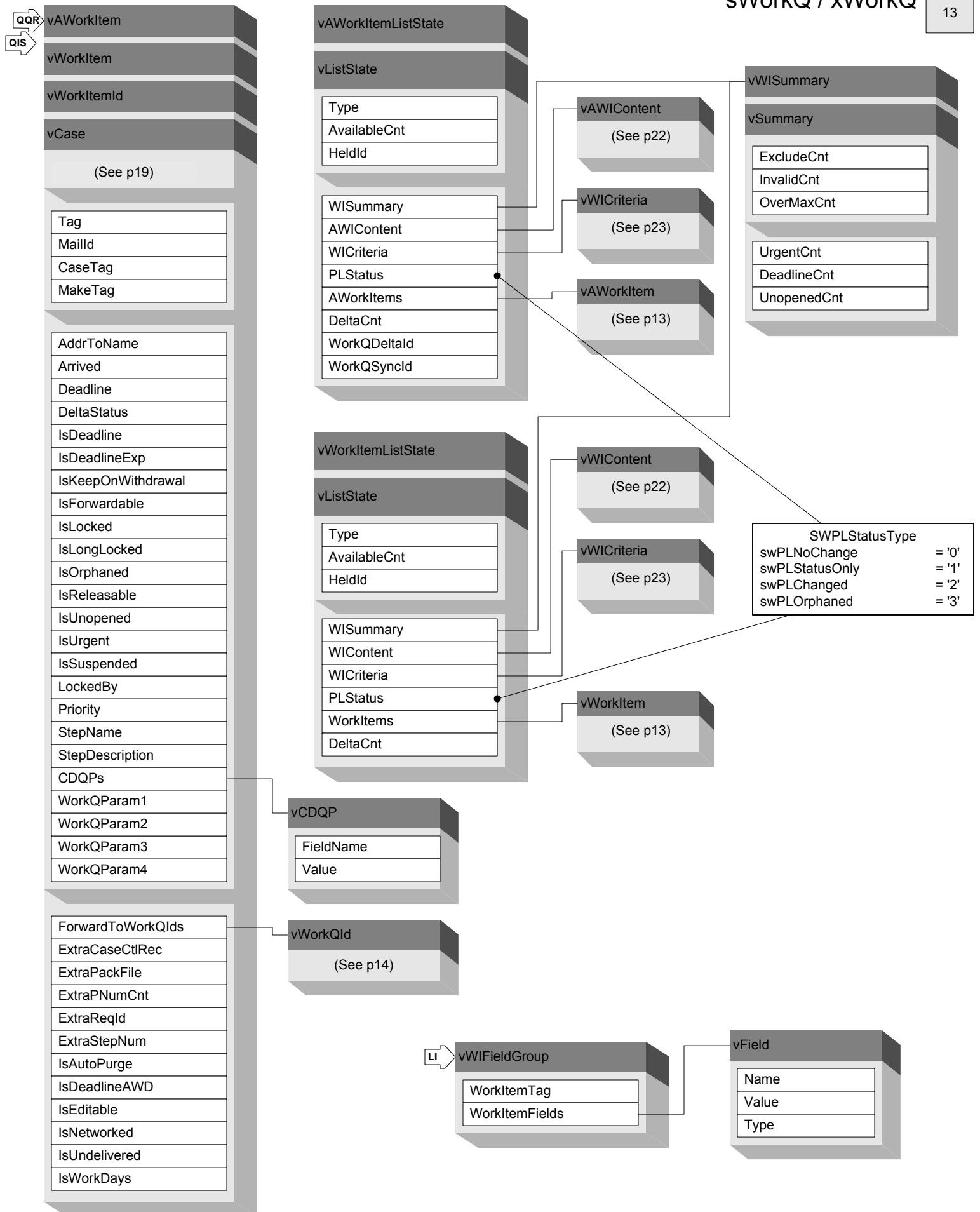
(msg) (Msg in parentheses) For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.

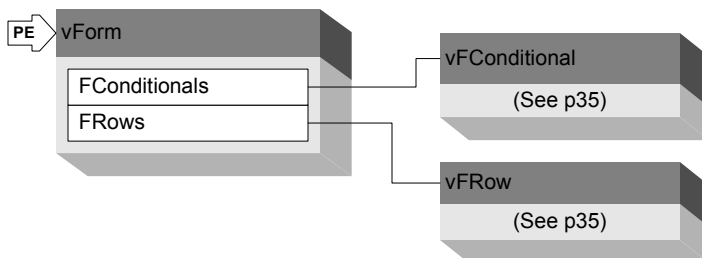
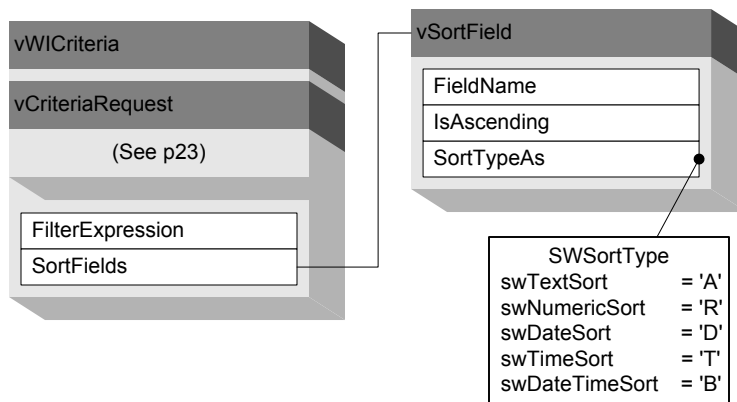
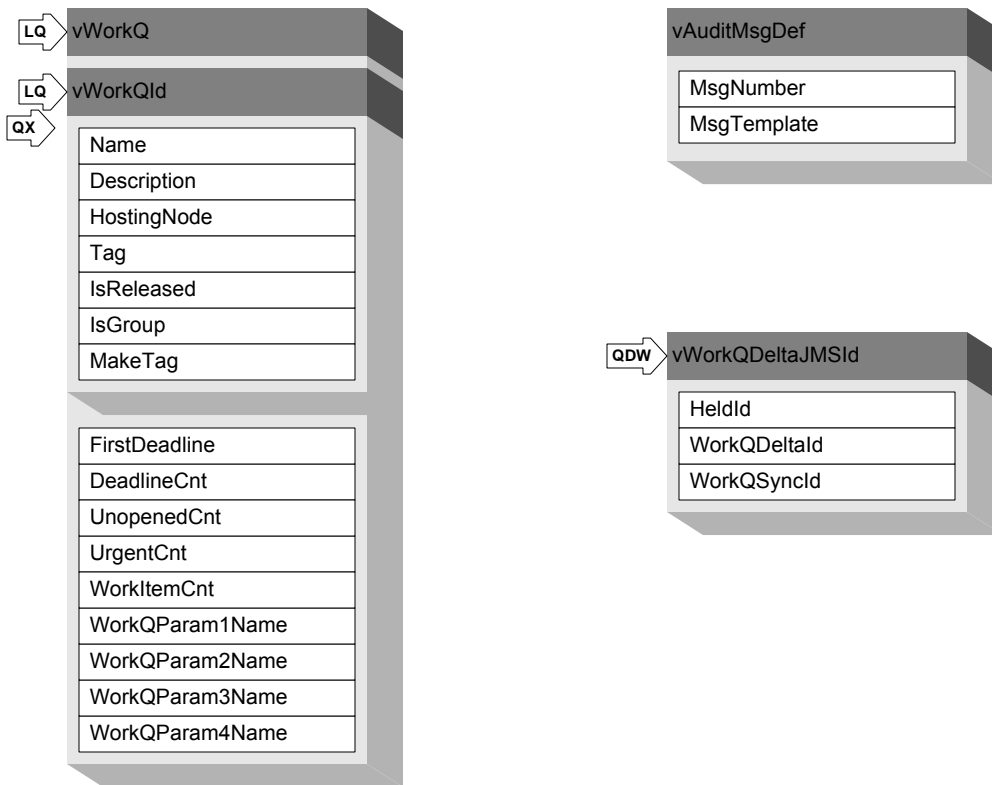
 Loop through message for each item.

 Get entire list and loop through to find items.

* On xWorkQ, most methods return void (the "Make" methods are an exception); you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.

** On sWorkQ, MakeAWorkItemListJMS returns a vAWorkItemListState object (p13); on xWorkQ, it returns a vWorkQDeltaJMSId object (p14).





This Server Object provides access to work queues for configuration, administrative, and reporting purposes.

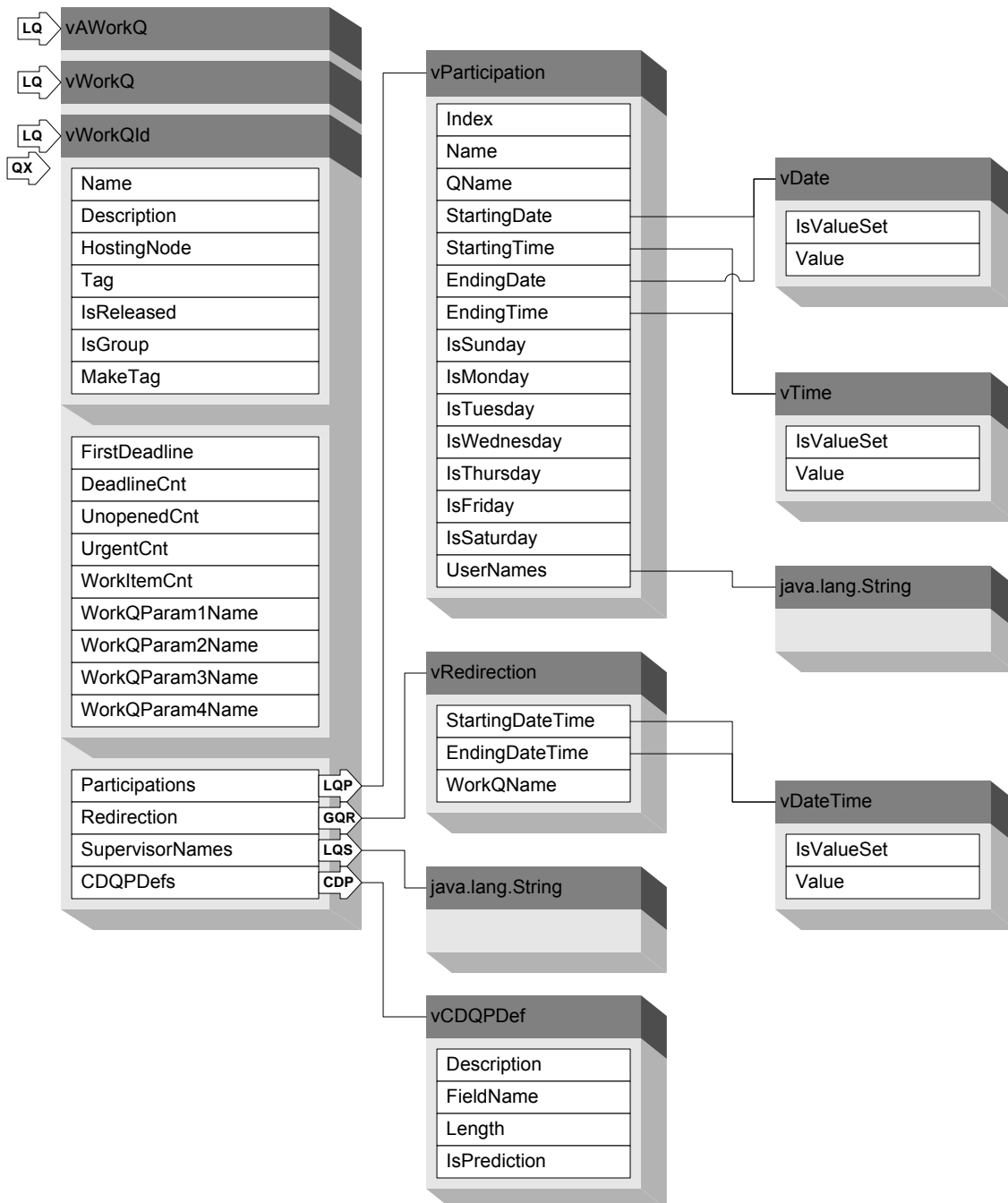
Use the sWorkQManager object if programming to the base .NET interface; use the xWorkQManager object if programming to the XML interface.

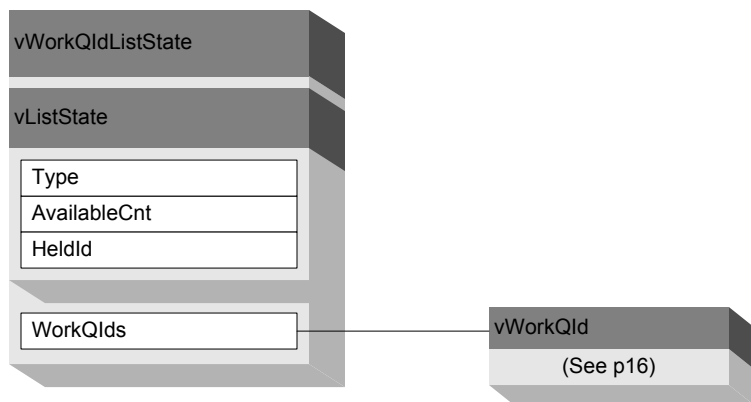
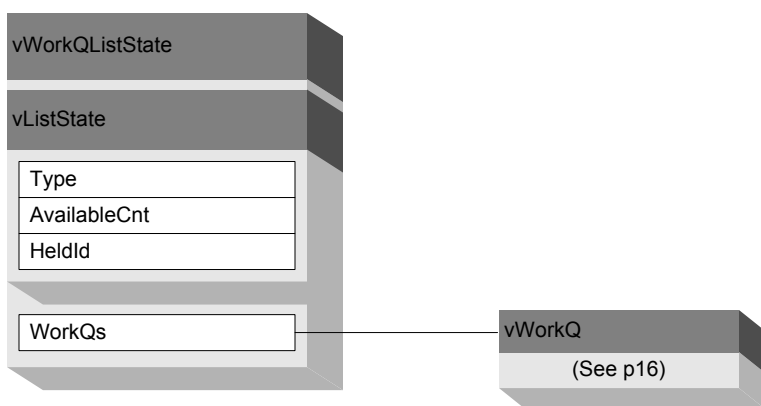
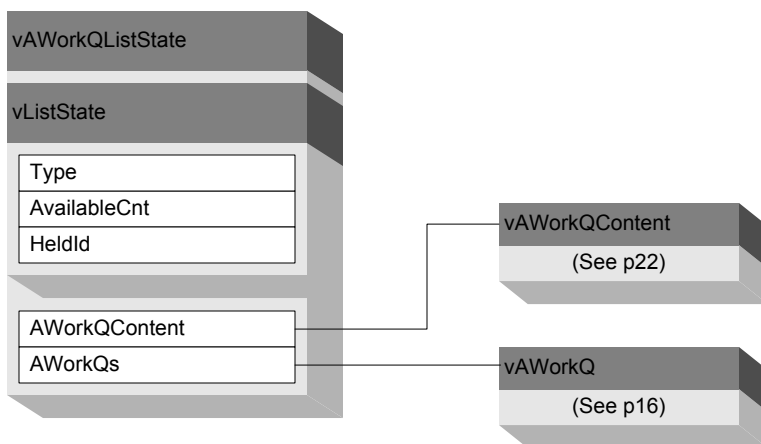
sWorkQManager / xWorkQManager					
sWorkQManager extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
GetWorkQIds	vWorkQId[] (p16)	LQ	X	X	
GetWorkQIdList	sPageableList - vWorkQId (p21/16)	QXW,QXR,(QX)	X		
MakeWorkQIdList	vWorkQIdListState (p17)		X	X	
FetchWorkQIdList	vWorkQIdListState (p17)		X	X	
GetWorkQs	vWorkQ[] (p16)	LQ	X	X	
GetWorkQList	sPageableList - vWorkQ (p21/16)	LQW,LQR,(LQ)	X		
GetWorkQListHeld	sPageableList - vWorkQ (p21/16)	LQW,LQR,(LQ)	X		
MakeWorkQList	vWorkQListState (p17)		X	X	
FetchWorkQList	vWorkQListState (p17)		X	X	
GetAWorkQs	vAWorkQ[] (p16)	LQ,LQP,GQR,LQS,CDP	X	X	
GetAWorkQList	sPageableList - vAWorkQ (p21/16)	LQW,LQR,(LQ),LQP,GQR,LQS,CDP	X		
GetAWorkQListHeld	sPageableList - vAWorkQ (p21/16)	LQW,LQR,(LQ),LQP,GQR,LQS,CDP	X		
MakeAWorkQList	vAWorkQListState (p17)		X	X	
FetchAWorkQList	vAWorkQListState (p17)		X	X	
GetSupervisorNames	string[]	LQS	X	X	
AddSupervisors	void	AQS	X	X	
RemoveSupervisors	void	RQS	X	X	
GetRedirection	vRedirection (p16)	GQR	X	X	
ChangeRedirection	void	SQR	X	X	
CancelRedirection	void	SQR	X	X	
GetParticipations	vParticipation[] (p16)	LQP	X	X	
CreateParticipations	void	AQP,CQP	X	X	
ChangeParticipation	void	LQP,MQP,CQP	X	X	
RemoveParticipations	void	LQP,RQP,CQP	X	X	
GetCDQPDefs	vCDQPDef[] (p16)	CDP	X	X	

* On xWorkQManager, most methods return void (the "Make" methods are an exception); you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.

Key








- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
- Loop through message for each item.
- Get entire list and loop through to find items.







This Server Object is used to manage live cases of a procedure.

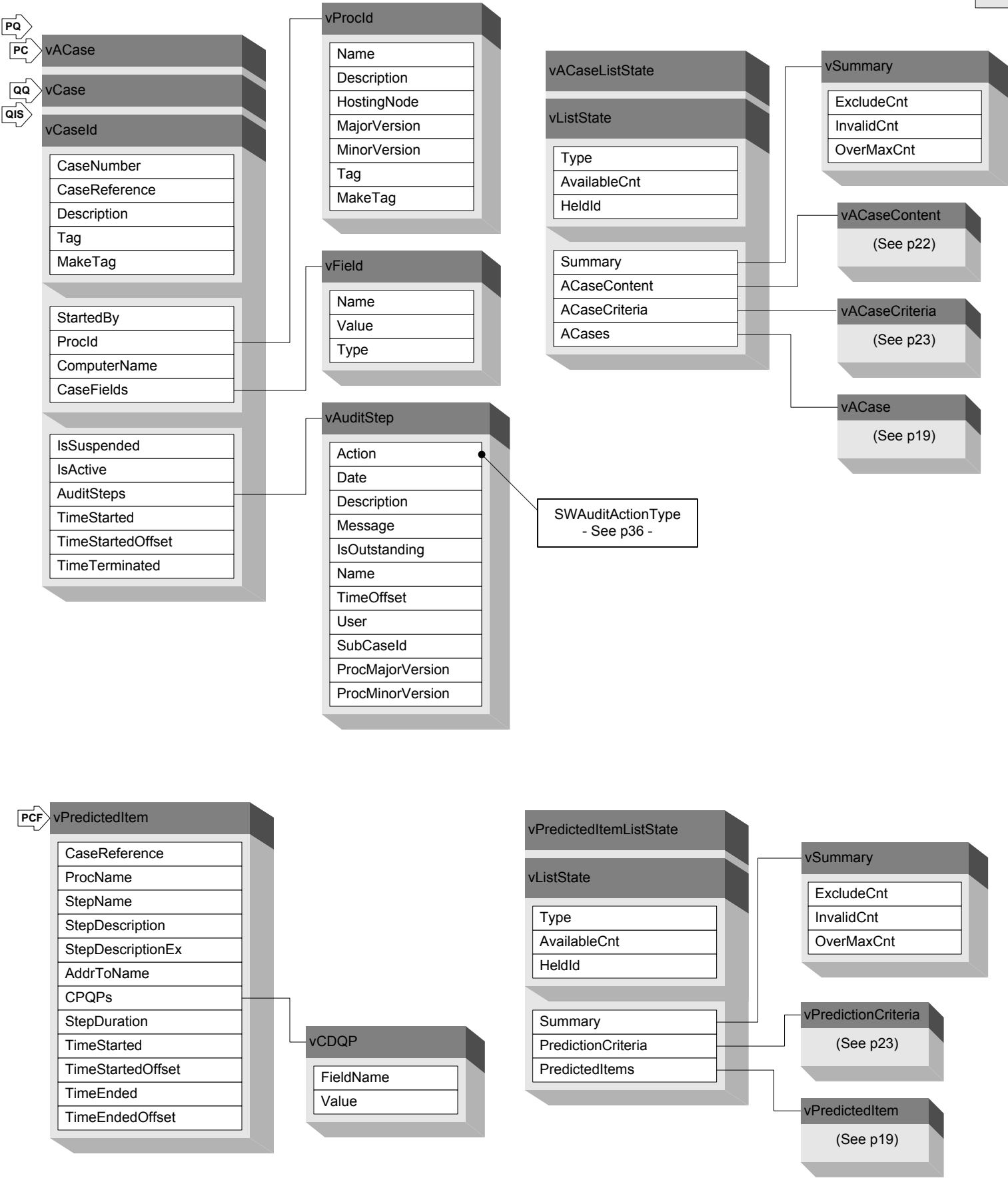
Use the sCaseManager object if programming to the base .NET interface; use the xCaseManager object if programming to the XML interface.

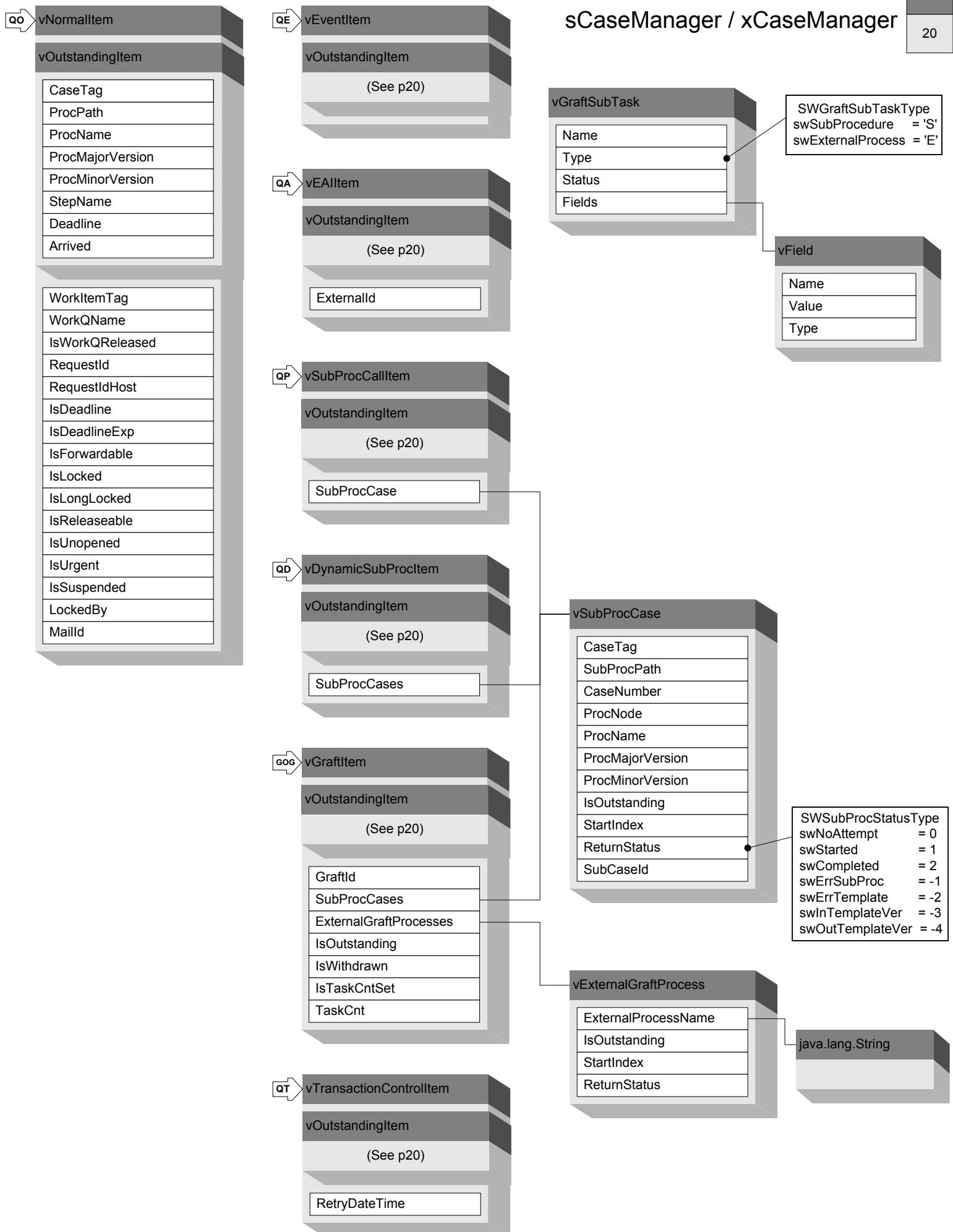
sCaseManager / xCaseManager					
sCaseManager extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
GetProcIds	vProcId[] (p19)	LP	X	X	
GetACases	vACase[] (p19)	PC 	X	X	
GetACaseList	sPageableList - vACase (p21/19)	PQW,PQR,(PQ)	X		
GetACaseListHeld	sPageableList - vACase (p21/19)	PQW,PQR,(PQ)	X		
MakeACaseList	vACaseListState (p19)		X	X	
FetchACaseList	vACaseListState (p19)		X	X	
GetAuditSteps	vAuditStep[] (p19)	PC	X	X	
SetCaseData	void	SCD	X	X	
SetCaseSuspended	void	SS 	X	X	
GetCaseCnt	long	PQC	X	X	
GetOutstandingItems	vOutstandingItem[] (p20)	QO,QP,QA,GOG,QD,QE	X	X	
GetSubProcCases	vSubProcCase[] (p20)	QS	X	X	
JumpTo	void	JT	X	X	
TriggerEvent	void	TE	X	X	
ReleaseEAllItem	void	REW	X	X	
SetGraftTaskCnt	void	SGC	X	X	
StartGraftTask	void	SGT	X	X	
DeleteGraftTask	void	DGT	X	X	
ExternalGraftProcessComp	void	EPC	X	X	
PredictCase	vPredictedItem[] (p19)	LP,PCF	X	X	
GetPredictedItemList	sPageableList - vPredictedItem (p21/19)	PWW,PWR	X		
GetPredictedItemListHeld	sPageableList - vPredictedItem (p21/19)	PWW,PWR	X		
MakePredictedItemList	vPredictedItemListState (p19)		X	X	
FetchPredictedItemList	vPredictedItemListState (p19)		X	X	
CloseCases	void	CP 	X	X	
CloseCasesByCriteria	void	CE 	X	X	
PurgeCases	void	CP 	X	X	
PurgeCasesByCriteria	void	CE 	X	X	
PurgeAndReset	void	PR 	X	X	

* On xCaseManager, most methods return void (the "Make" methods are an exception); you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.





This Server Object provides access to information that has been returned from the TIBCO iProcess Objects Server on a pageable list.



sPageableList					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
ItemsPerBlock	int		X		
AvailableCnt	int	??W	X		
LocalCnt	int	??W	X		
Summary	vSummary (p23)	??W	X		
Hold	string	XLP,XDP	X		
HeldId	string		X		
GetItem	System.Object	??R, PQ*	X		
Type	SWPageableListType (p37)		X		
IsKeepLocalItems	bool		X		
Clear	void		X		
ContentRequest	vContentRequest (p22)		X		
CriteriaRequest	vCriteriaRequest (p23)		X		
Dispose	void	DVW,DDW	X		

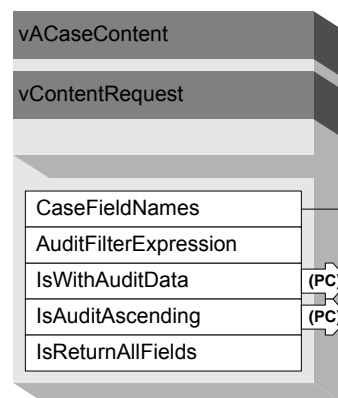
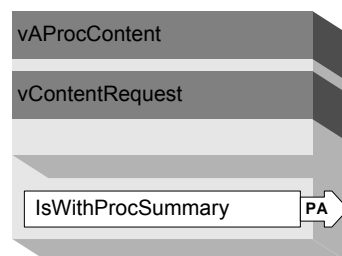
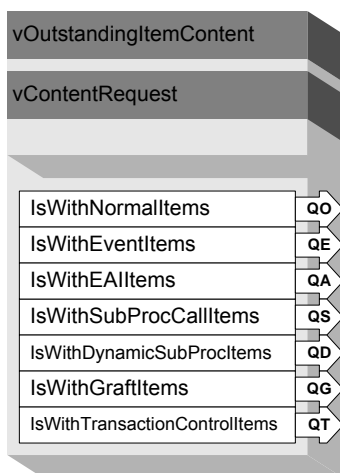
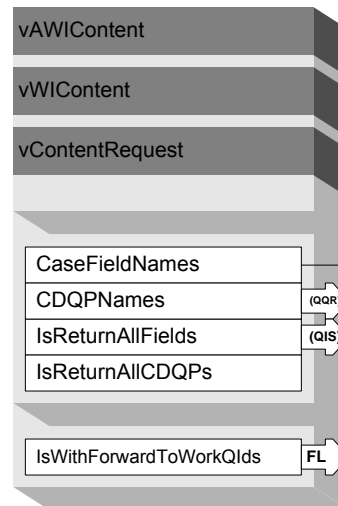
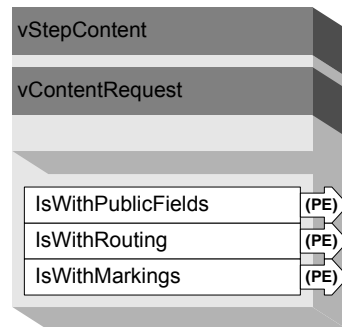
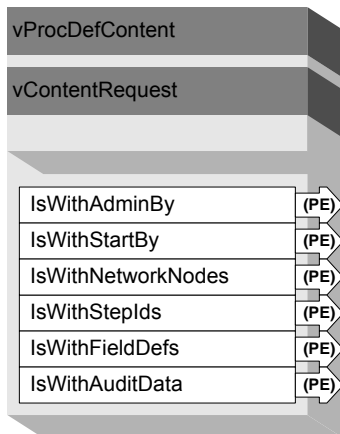
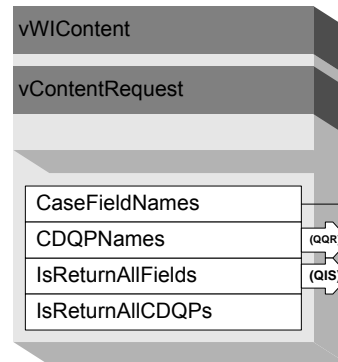
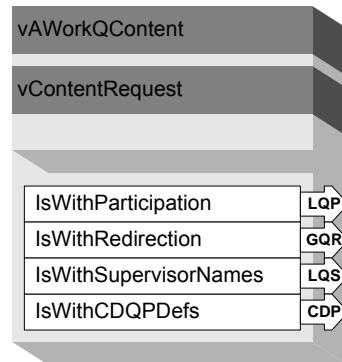
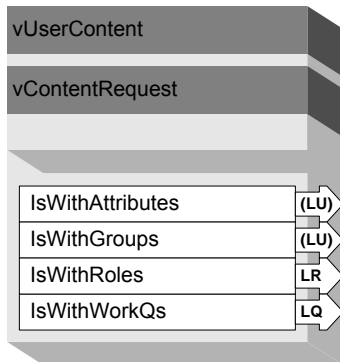
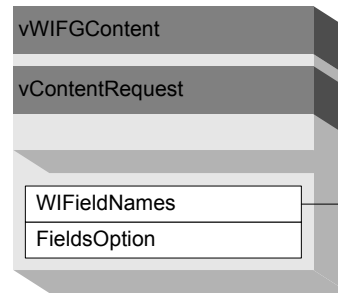
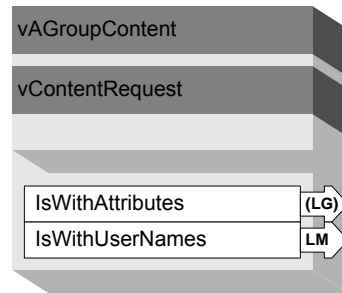
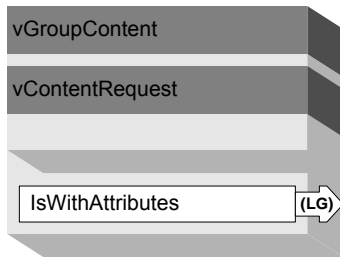
* PQ message is sent only when items are cases.

sPageableListR					
Extends sPageableList					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
Refresh	SWPLStatusType (p37)	??W	X		
DeltaCnt	int		X		
GetNextDeltaItem	vWorkItem (p13)		X		
LockFirstItem	vWILocked (p23)	LFI	X		

sPageableListJ					
Extends sPageableList					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
StartWorkQDeltaJMSPublish	void	QDS	X		
WorkQDeltald	String	QDW	X		
WorkQSyncld	String	QDW	X		

Key

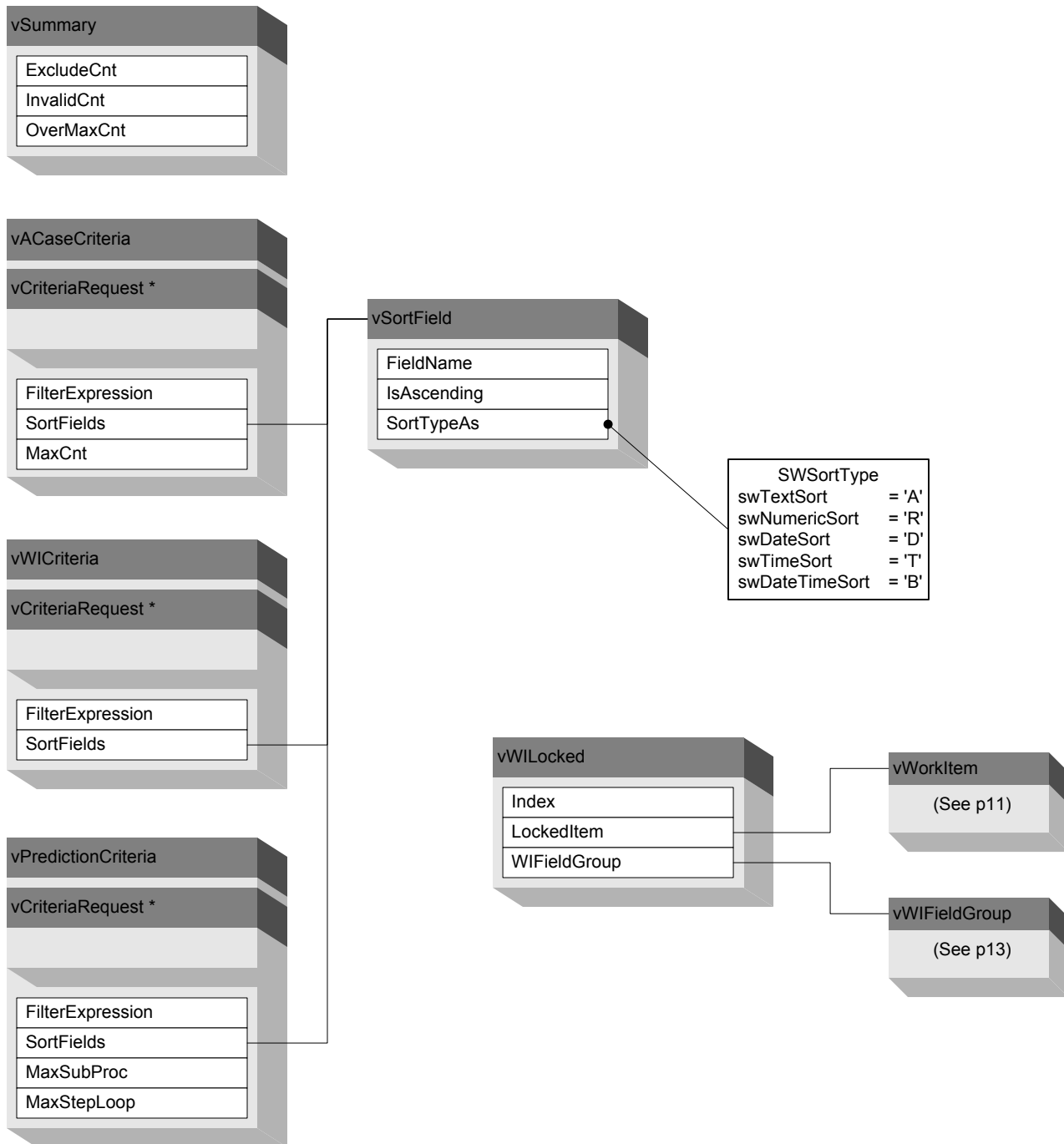
- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses) For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
-  Loop through message for each item.
-  Get entire list and loop through to find items.



If the message is in parenthesis, it means this flag will cause additional data to be parsed out of an existing message, no additional message is sent.



If the message is not in parenthesis, it means this flag will cause an additional message to be sent to the server to get this data.



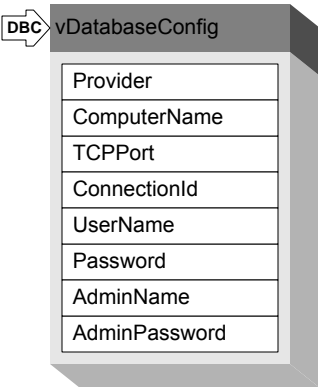
* Note that vCriteriaRequest is the base class for all "criteria" objects, but does not contain any public methods.

This Server Object is used to get database configuration information on a TIBCO iProcess Engine.

Use the sProcManager object if programming to the base interface; use the xProcManager object if programming to the XML interface.

sIPEConfig / xIPEConfig					
sIPEConfig extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
GetDatabaseConfig	vDatabaseConfig (p24)	DBC	X	X	X
GetActivityPub	string	GAP	X	X	X
SetActivityPub	void	SAP	X	X	X

* On xIPEConfig, all methods return void; you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.



Key


(pXX)

Page in this document showing returned object.


(msg)

(Msg in parentheses)

For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.



Loop through message for each item.



Get entire list and loop through to find items.

This Server Object is used to generate and dynamically configure log messages for the client application.

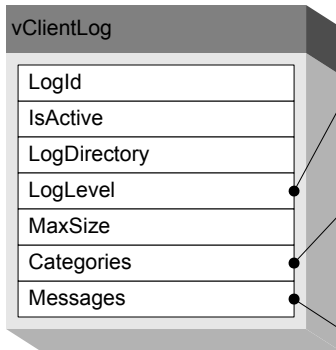
The sClientLog object is different than most Server Objects in that it does not require node, user name, and password information when it's constructed. It also does not create a session with a node when it is constructed.

Also note that there is no client log available in the XML namespace.

sClientLog					
Methods	Returns	Messages	Available In Interfaces:		
			Base	XML	
LogId	void		X		
Active	void		X		
LogDirectory	void		X		
LogLevel	void		X		
MaxSize	void		X		
Categories	void		X		
EnableCategory	void		X		
DisableCategory	void		X		
Messages	void		X		
EnableMessage	void		X		
DisableMessage	void		X		
GetClientLog	vClientLog (p26)		X		
Log	void		X		
ResetLog	void		X		

Key

- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
- Loop through message for each item.
- Get entire list and loop through to find items.



SWLogLevelType	
swLogError	= 1
swLogWarning	= 2
swLogInformation	= 3
swLogDebug	= 4

SWLogCategoryType	
swCatAll	= 0x7FFFFFFF
swCatAlwaysLog	= 0x00000001
swCatSEouser	= 0x00000002
swCatConstDestr	= 0x00000004
swCatReceiveThread	= 0x00000008
swCatMessages	= 0x00000010
swCatMsgSend	= 0x00000020
swCatMsgReceive	= 0x00000040
swCatUDP	= 0x00000080
swCatWinSock	= 0x00000100
swCatConversion	= 0x00000200
swCatTiming	= 0x00000400
swCatMethodCalls	= 0x00000800
swCatObjectWrapping	= 0x00001000
swCatMemory	= 0x00002000

SWLogMessageType	
swMsgAll	= 0x7FFFFFFF
swMsgTCP	= 0x00000001
swMsgUDP	= 0x00000002
swMsgLogin	= 0x00000004
swMsgPassword	= 0x00000008
swMsgUser	= 0x00000010
swMsgAttribute	= 0x00000020
swMsgRole	= 0x00000040
swMsgGroup	= 0x00000080
swMsgProcedure	= 0x00000100
swMsgProcedureQuery	= 0x00000200
swMsgProcedureDefinition	= 0x00000400
swMsgQueueAccess	= 0x00000800
swMsgQueueQuery	= 0x00001000
swMsgCase	= 0x00002000
swMsgNode	= 0x00004000
swMsgEvent	= 0x00008000
swMsgWorkItem	= 0x00010000
swMsgForwarding	= 0x00020000
swMsgInstrumentation	= 0x00040000
swMsgMemoAttachment	= 0x00080000
swMsgForm	= 0x00100000
swMsgTable	= 0x00200000
swMsgListValidation	= 0x00400000

This Server Object provides access to the definition of a procedure. Procedures are defined using the TIBCO iProcess Modeler.

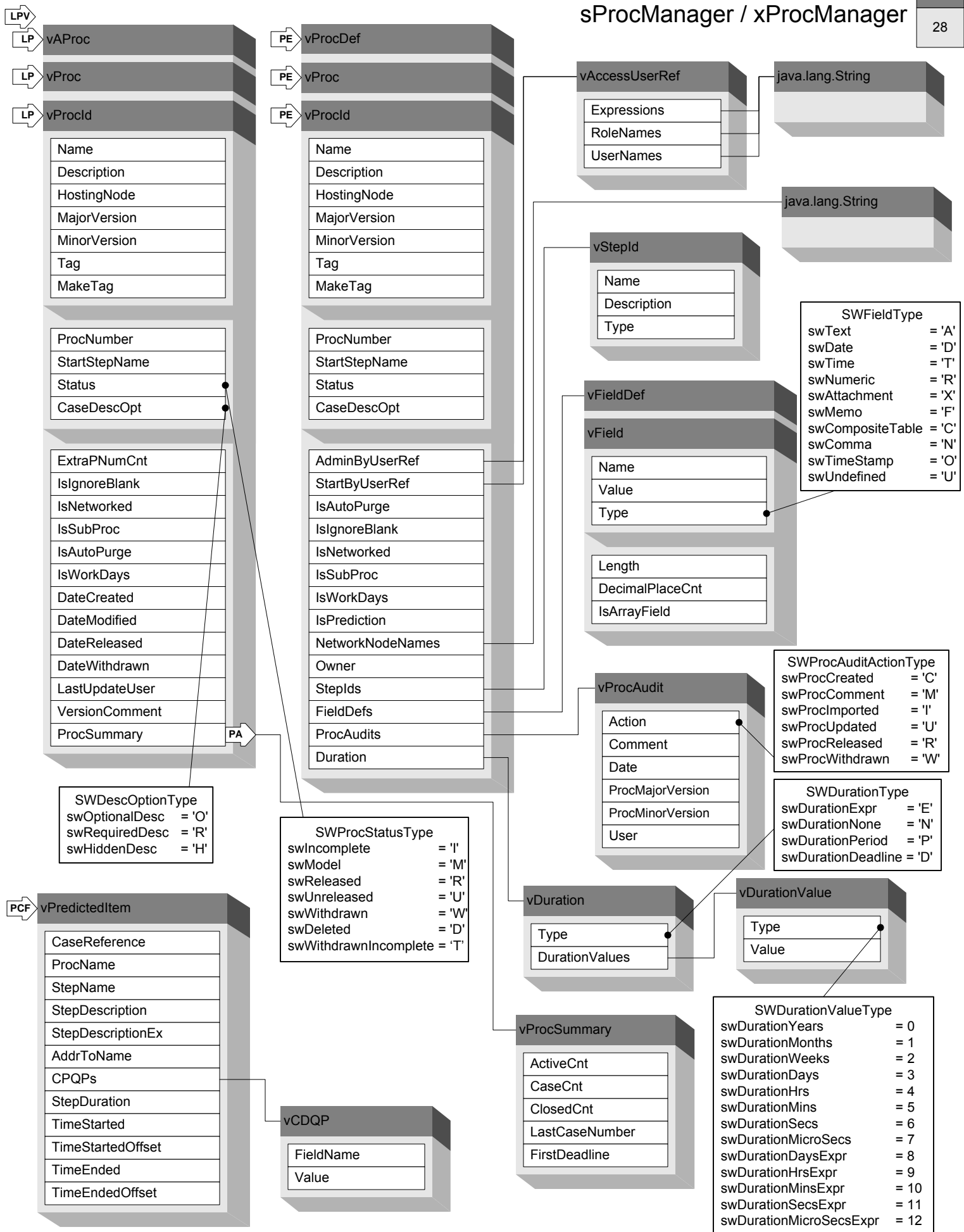
Use the sProcManager object if programming to the base .NET interface; use the xProcManager object if programming to the XML interface.

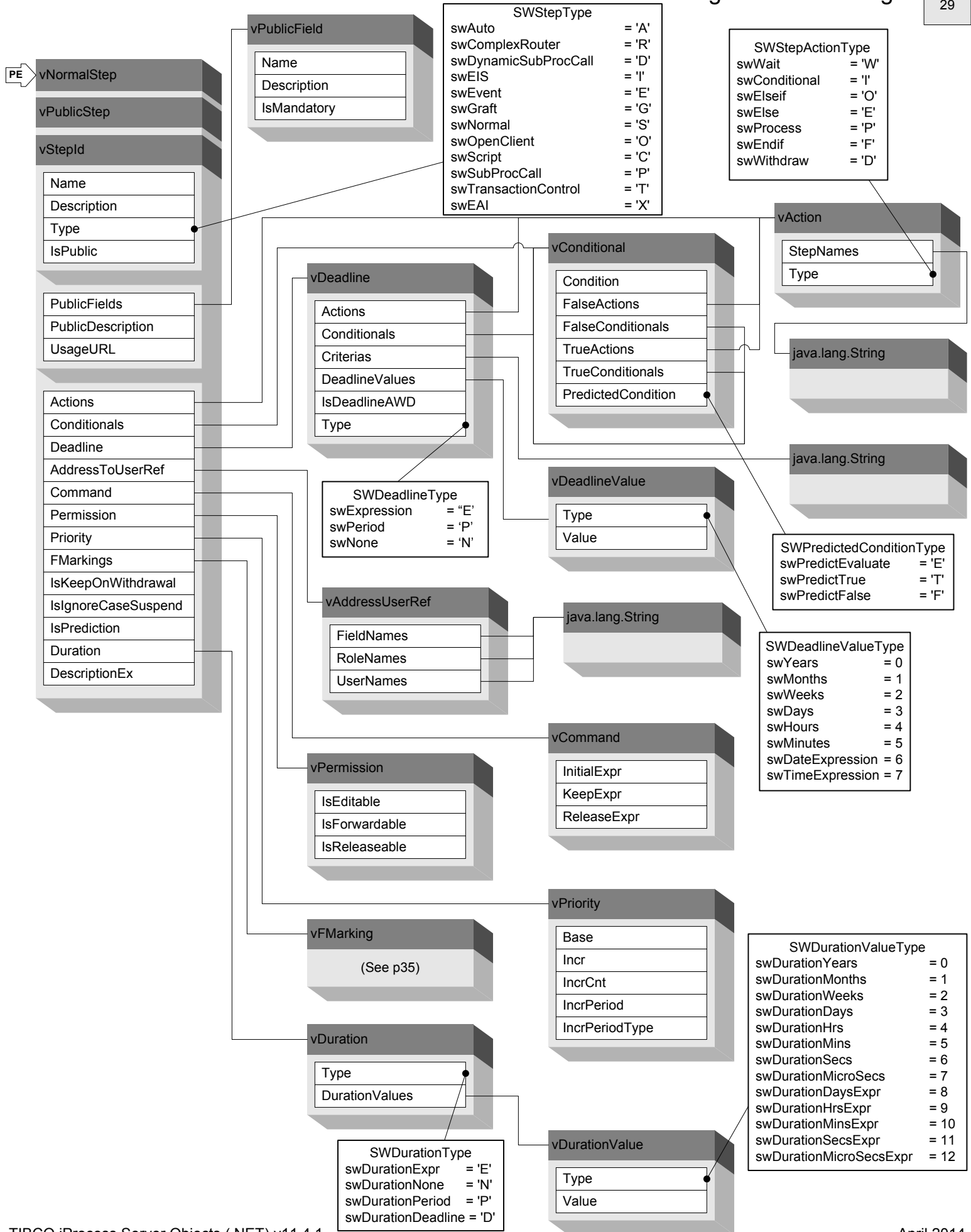
sProcManager / xProcManager					
sProcManager extends sBase					
Methods	Returns *	Messages	Available In Interfaces:		
			Base	XML	
GetProcIds	vProcId[] (p28)	LP	X	X	
GetProcs	vProc[] (p28)	LP	X	X	
GetAProcs	vAProc[] (p28)	LP,PA	X	X	
GetProcDefs	vProcDef[] (p28)	PE	X	X	
GetProcVersions	vAProc[] (p28)	LPV,PA	X	X	
GetStepIds	vStepId[] (p29)	PE	X	X	
GetSteps	vStepId[] (p29)	PE	X	X	
GetPublicSteps	vPublicStep[] (p29)	PE	X	X	
GetFieldDefs	vFieldDef[] (p30)	PE	X	X	
GetForm	vForm (p35)	PF,ME	X	X	
GetFormMarkings	vAFMarking[] (p35)	PF,ME	X	X	
GetExternalForm	string	EF	X	X	
GetPluginForm	vPluginForm (p30)	FPI	X	X	
SimulateCase	vPredictedItem[] (p28)	CS	X	X	

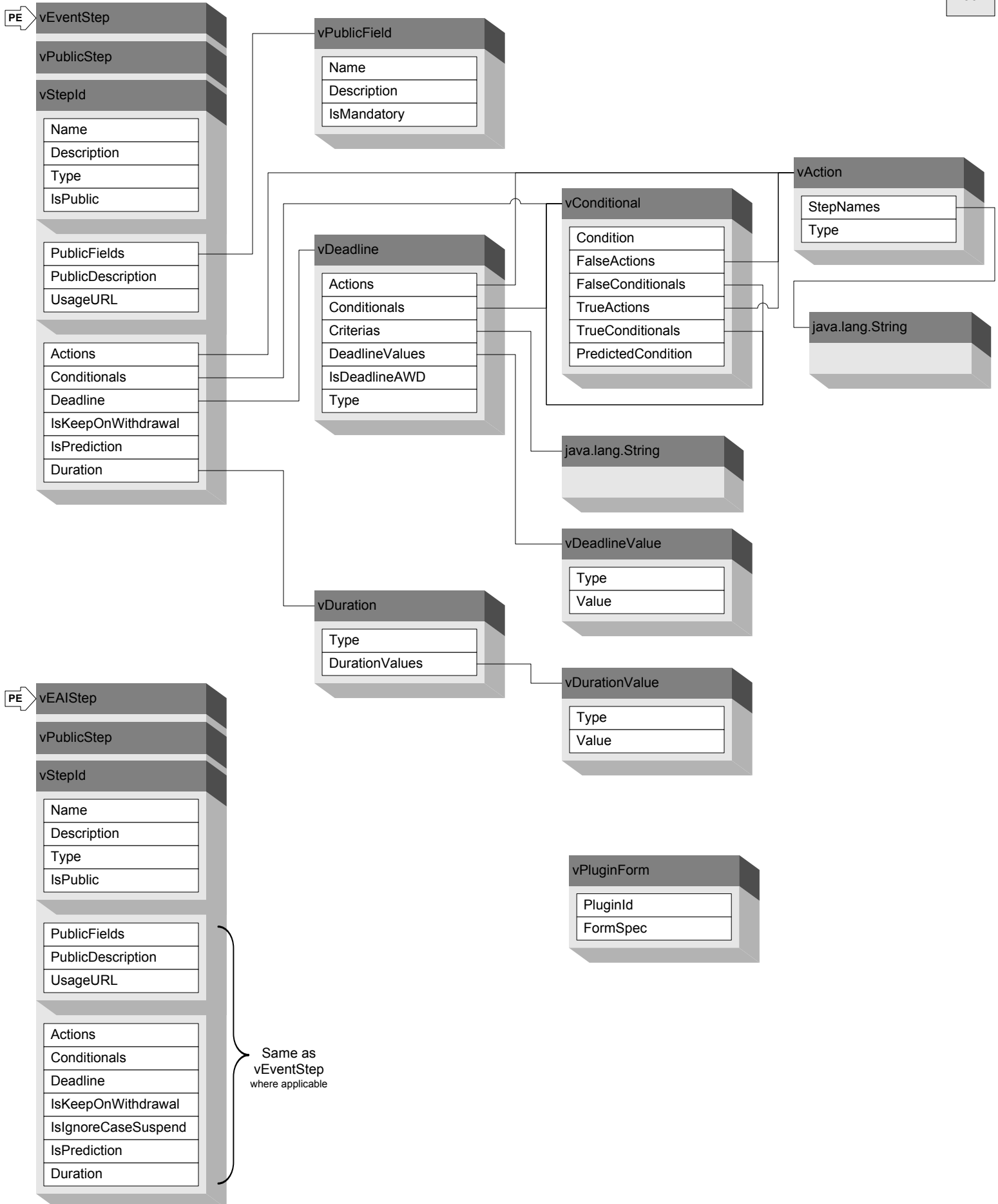
* On xProcManager, all methods return void; you must call xSession.GetXMLResults to get the serialized version of the Value Objects in an XML data stream.

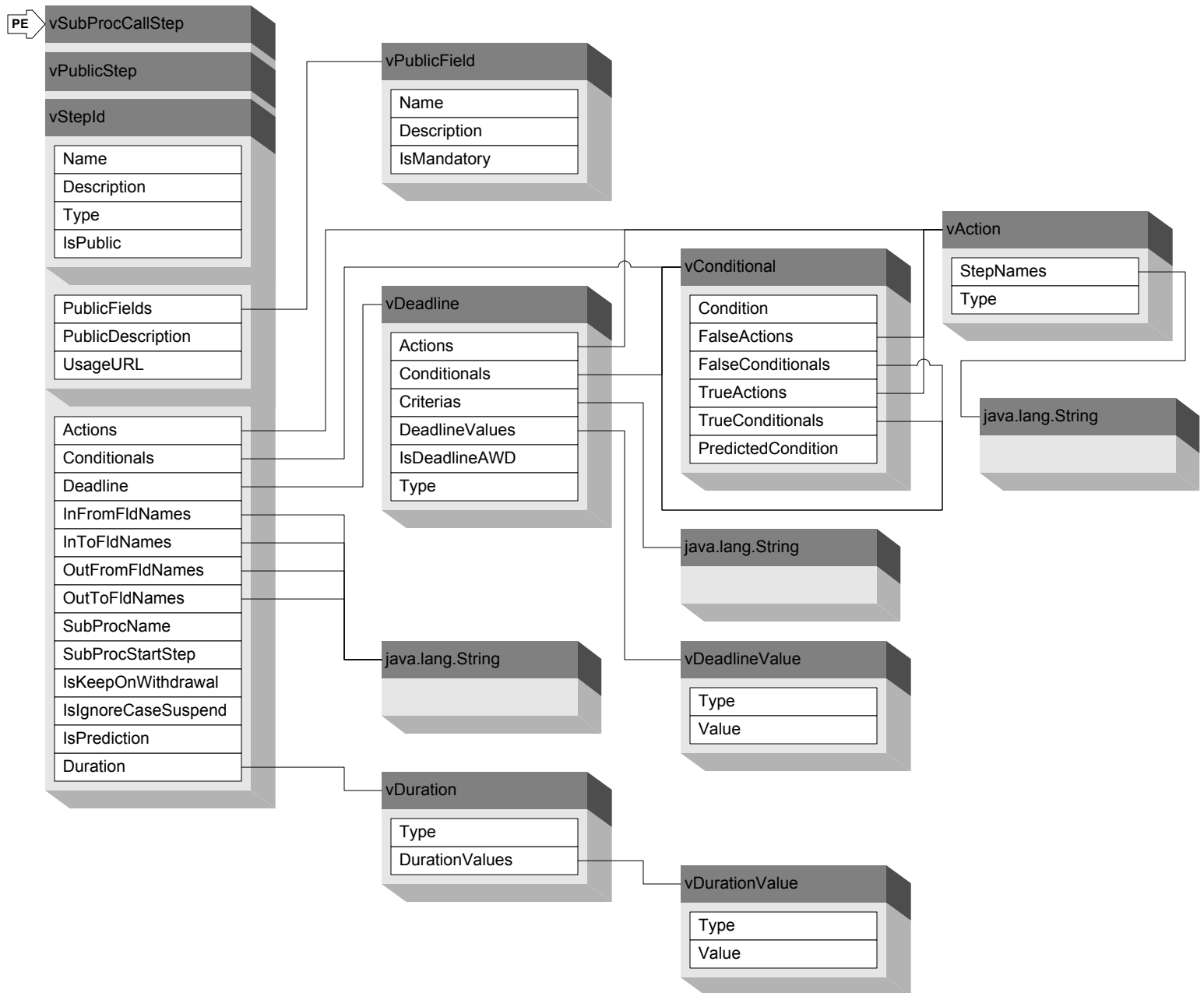
Key

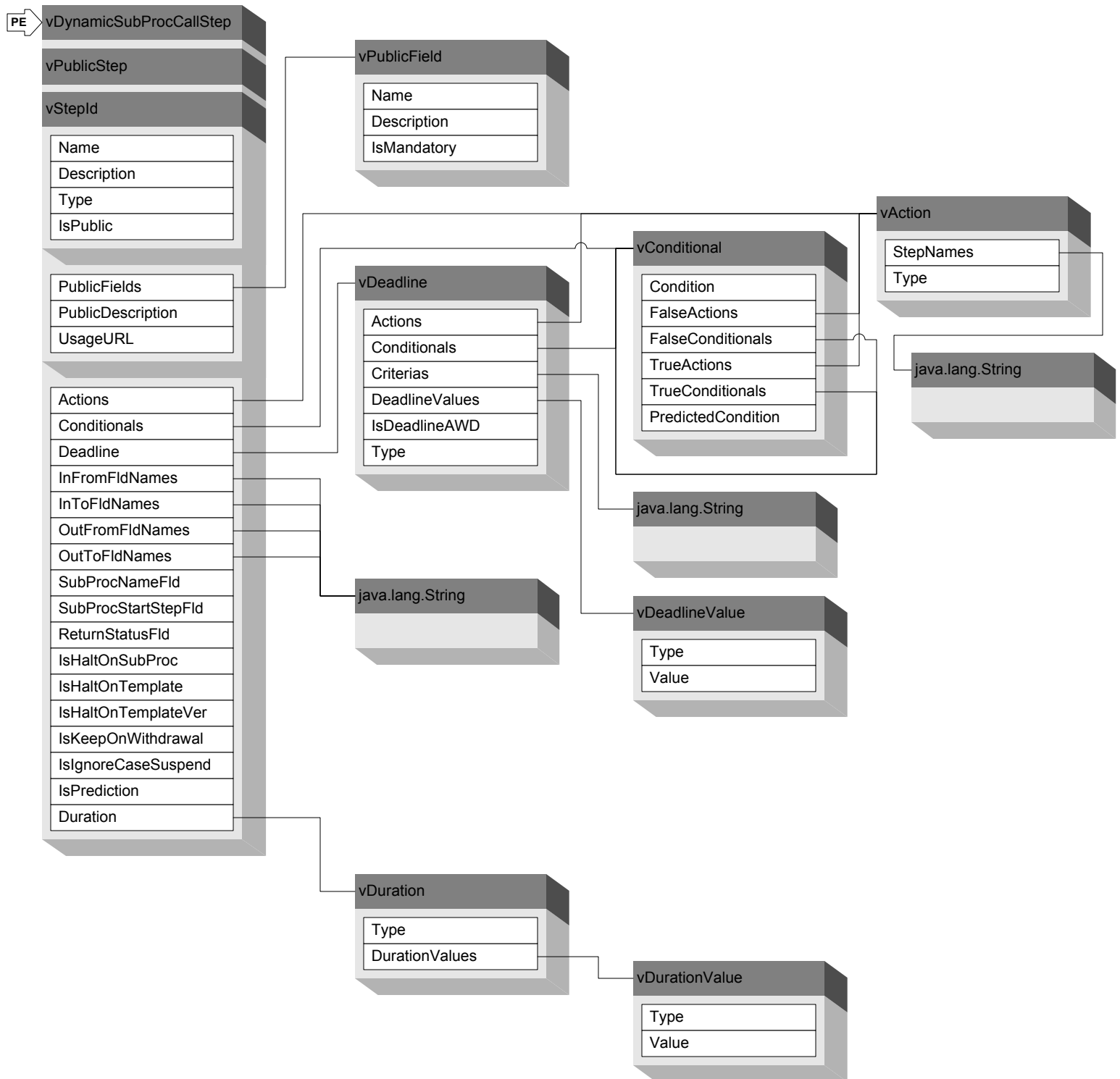
- (pXX) Page in this document showing returned object.
- (msg) (Msg in parentheses)
For non-work item pageable lists, the message that is actually sent to the server is in parenthesis.
- Loop through message for each item.
- Get entire list and loop through to find items.

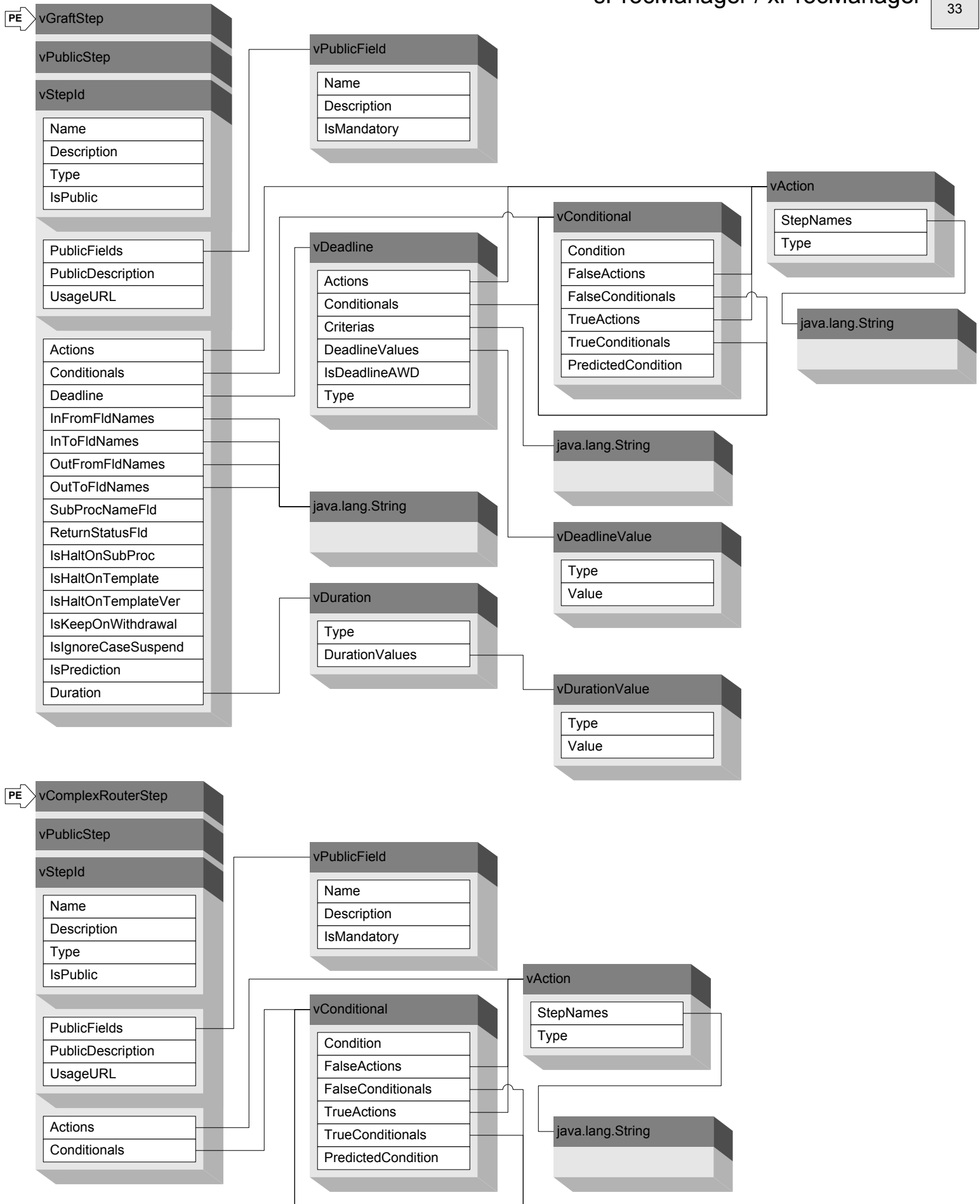


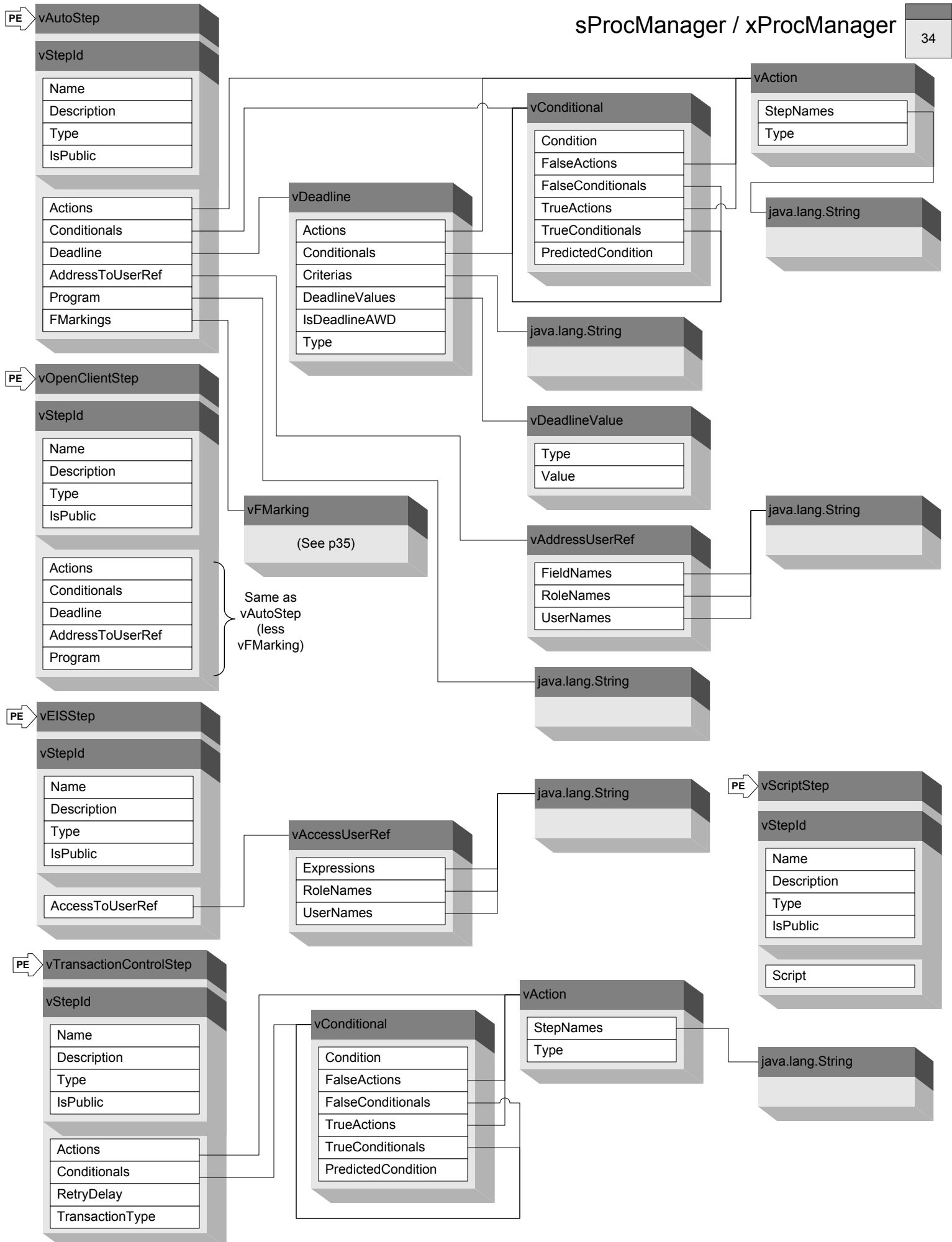












SWAttributeType

swTextAttr	= 'A'
swNumericAttr	= 'R'
swDateAttr	= 'D'
swTimeAttr	= 'T'

SWAuditActionType

swStartCase	= 0
swProcessedTo	= 1
swReleasedBy	= 2
swDeadlineExp	= 3
swForwarded	= 4
swProcessedFor	= 5
swError	= 6
swTermAbnormal	= 7
swTermPremature	= 8
swTermNORMAL	= 9
swRevisedBy	= 10
swReleasedMBox	= 11
swModifiedBy	= 12
swDeadlineWdl	= 13
swResent	= 14
swEventIssued	= 15
swSubCaseStart	= 16
swSubCaseComp	= 17
swSubCaseTerm	= 18
swSubCaseExpired	= 19
swSubCaseWithdrawn	= 20
swRedirectedTo	= 21
swSuspendedBy	= 22
swResumedBy	= 23
swCaseJumpBy	= 24
swDynaGraftCaseStart	= 25
swTaskCountSet	= 26
swTaskDeleted	= 27
swSubCaseGrafted	= 28
swExtProcessGrafted	= 29
swGraftInitiated	= 30
swExtProcessReleased	= 31
swGraftReleased	= 32
swDynamicReleased	= 33
swCaseMigrated	= 34
swGraftWithdrawn	= 35
swDynaGraftDeadlineExp	= 36
swDynamicWithdrawn	= 37
swKeepOnWithdraw	= 38
swReleasedNoAddressees	= 39
swReleasedNoSubProcs	= 40
swForwardedBy	= 41
swEAIcAllInitiated	= 50
swEAIcAllComplete	= 51
swEAIcAllExpired	= 52
swEAIcAllWithdrawn	= 53
swTransProcessed	= 54
swTransStarted	= 55
swTransRestart	= 56
swCasePurged	= 57
swCDModified	= 58
swWIOpenedBy	= 59
swWIKeptBy	= 60
swTriggered	= 61
swCaseDeadline	= 62
swEAIcAllFailed	= 80
swErrMaxActions	= 81
swErrGenericTransfail	= 82
swEAINoPlugin	= 83
swErrBadSubProc	= 84
swErrDiffTemplate	= 85
swErrDiffTemplateVer	= 86
swTransAborted	= 87
swDeliveredToExchange	= 128
swReleasedFromExchange	= 129
swWithdrawnFromExchange	= 130
swBWActivity	= 131
swCaseDataByUser	= 133

SWDeadlineType

swExpression	= 'E'
swPeriod	= 'P'
swNone	= 'N'

SWDescOptionType

swOptionalDesc	= 'O'
swRequiredDesc	= 'R'
swHiddenDesc	= 'H'

SWDeadlineCalculateType

swNoReCalc	= '0'
swCaseOnly	= '1'
swIncludeSubCases	= '2'

SWDeadlineValueType

swYears	= 0
swMonths	= 1
swWeeks	= 2
swDays	= 3
swHours	= 4
swMinutes	= 5
swDateExpression	= 6
swTimeExpression	= 7

SWDeltaStatusType

swNotDeltaItem	= '0'
swDeleted	= 'd'
swAdded	= 'a'
swModified	= 'm'

SWDurationType

swDurationExpr	= 'E'
swDurationNone	= 'N'
swDurationPeriod	= 'P'
swDurationDeadline	= 'D'

SWDurationValueType

swDurationYears	= 0
swDurationMonths	= 1
swDurationWeeks	= 2
swDurationDays	= 3
swDurationHrs	= 4
swDurationMins	= 5
swDurationSecs	= 6
swDurationMicroSecs	= 7
swDurationDaysExpr	= 8
swDurationHrsExpr	= 9
swDurationMinsExpr	= 10
swDurationSecsExpr	= 11
swDurationMicroSecsExpr	= 12

SWErrorGroupType

swParameterException	= 0
swMethodException	= 1
swItemException	= 2
swMessageException	= 3
swSEOServerException	= 4
swSSOException	= 5
swOS_APIException	= 6
swSocketException	= 7
swNoSuchElementException	= 8
swOutOfMemoryException	= 9
swIllegalAccessException	= 10
swUncategorizedException	= 11

SWFieldsOptionType

ssoFormMarkings	= '0'
ssoAllMarkings	= '1'
ssoFieldList	= '2'

SWFieldType

swText	= 'A'
swDate	= 'D'
swTime	= 'T'
swNumeric	= 'R'
swAttachment	= 'X'
swMemo	= 'F'
swCompositeTable	= 'C'
swComma	= 'N'
swTimeStamp	= 'O'
swArrayOfText	= 'a'
swArrayOfDate	= 'd'
swArrayOfTime	= 't'
swArrayOfNumeric	= 'r'
swArrayOfMemo	= 'f'
swArrayOfCompositeTable	= 'c'
swArrayOfComma	= 'n'
swUndefined	= '?'

SWGraftSubTaskType

swSubProcedure	= 'S'
swExternalProcess	= 'E'

SWLogLevelType

swLogError	= 1
swLogWarning	= 2
swLogInformation	= 3
swLogDebug	= 4

SWLogCategoryType

swMsgAttribute	= 0x00000020
swCatAll	= 0x7FFFFFFF
swCatAlwaysLog	= 0x00000001
swCatSEOUser	= 0x00000002
swCatConstDestr	= 0x00000004
swCatReceiveThread	= 0x00000008
swCatMessages	= 0x00000010
swCatMsgSend	= 0x00000020
swCatMsgReceive	= 0x00000040
swCatUDP	= 0x00000080
swCatWinSock	= 0x00000100
swCatConversion	= 0x00000200
swCatTiming	= 0x00000400
swCatMethodCalls	= 0x00000800
swCatObjectWrapping	= 0x00001000
swCatMemory	= 0x00002000

SWLogMessageType	
swMsgAll	= 0x7FFFFFFF
swMsgTCP	= 0x00000001
swMsgUDP	= 0x00000002
swMsgLogin	= 0x00000004
swMsgPassword	= 0x00000008
swMsgUser	= 0x00000010
swMsgAttribute	= 0x00000020
swMsgRole	= 0x00000040
swMsgGroup	= 0x00000080
swMsgProcedure	= 0x00000100
swMsgProcedureQuery	= 0x00000200
swMsgProcedureDefinition	= 0x00000400
swMsgQueueAccess	= 0x00000800
swMsgQueueQuery	= 0x00001000
swMsgCase	= 0x00002000
swMsgNode	= 0x00004000
swMsgEvent	= 0x00008000
swMsgWorkItem	= 0x00010000
swMsgForwarding	= 0x00020000
swMsgInstrumentation	= 0x00040000
swMsgMemoAttachment	= 0x00080000
swMsgForm	= 0x00100000
swMsgTable	= 0x00200000
swMsgListValidation	= 0x00400000

SWMarkingType	
swOptional	= 'O'
swRequired	= 'R'
swHidden	= 'H'
swDisplay	= 'D'
swCalculated	= 'C'
swEmbedded	= 'E'

SWPageableListType	
ssoPLUser	= 1
ssoPLAGroup	= 2
ssoPLWorkQ	= 3
ssoPLAWorkQ	= 4
ssoPLOSUser	= 5
ssoPLCase	= 6
ssoPLWorkItem	= 7
ssoPLAWorkItem	= 8
ssoPLWorkQId	= 9
ssoPLPredictedItem	= 10

SWPredictedConditionType	
swPredictEvaluate	= 'E'
swPredictTrue	= 'T'
swPredictFalse	= 'F'

SWPLRefreshType	
swUpdate	= '0'
swRecreate	= '1'
swUpdateWithDelta	= '2'

SWPLStatusType	
swPLNoChange	= '0'
swPLStatusOnly	= '1'
swPLChanged	= '2'
swPLOrphaned	= '3'

SWProcAuditActionType	
swProcCreated	= 'C'
swProcComment	= 'M'
swProcImported	= 'I'
swProcUpdated	= 'U'
swProcReleased	= 'R'
swProcWithdrawn	= 'W'

SWProcStatusType	
swIncomplete	= 'I'
swModel	= 'M'
swReleased	= 'R'
swUnreleased	= 'U'
swWithdrawn	= 'W'
swDeleted	= 'D'
swWithdrawnIncomplete	= 'T'

SWStepActionType	
swWait	= 'W'
swConditional	= 'I'
swElseif	= 'O'
swElse	= 'E'
swProcess	= 'P'
swEndif	= 'F'
swWithdraw	= 'D'

SWStepType	
swAuto	= 'A'
swComplexRouter	= 'R'
swDynamicSubProcCall	= 'D'
swEIS	= 'I'
swEvent	= 'E'
swGraft	= 'G'
swNormal	= 'S'
swOpenClient	= 'O'
swScript	= 'C'
swSubProcCall	= 'P'
swTransactionControl	= 'T'
swEAI	= 'X'

SWSortType	
swTextSort	= 'A'
swNumericSort	= 'R'
swDateSort	= 'D'
swTimeSort	= 'T'
swDateTimeSort	= 'B'

SWSubProcStatusType	
swNoAttempt	= 0
swStarted	= 1
swCompleted	= 2
swErrSubProc	= -1
swErrTemplate	= -2
swInTemplateVer	= -3
swOutTemplateVer	= -4

SWSubProcPrecedenceType	
swPrecedenceR	= '0'
swPrecedenceUR	= '1'
swPrecedenceMR	= '2'
swPrecedenceUMR	= '3'
swPrecedenceMUR	= '4'

SWTransactionType	
swAbort	= 'A'
swContinue	= 'C'
swConcede	= 'D'

TIBCO iProcess Server Objects (.NET)

Object Model

Index

<u>Object</u>	<u>Page #</u>	<u>Object</u>	<u>Page #</u>	<u>Object</u>	<u>Page #</u>
sBase.....	1	vDatabaseConfig.....	24	vProc.....	11,28
sCaseManager.....	18	vDate.....	16	vProcAudit.....	28
sClientLog.....	25	vDateTime.....	16	vProcDef.....	28
sIPEConfig.....	24	vDeadline.....	29-34	vProcDefContent.....	22
sNode.....	4	vDeadlineValue.....	29-34	vProcId.....	11,19,28
sNodeManager.....	8	vDuration.....	28-33	vProcSummary.....	28
sPageableList.....	21	vDurationValue.....	28-33	vPublicField.....	29-33
sPageableListJ.....	21	vDynamicSubProcCallStep.....	32	vPublicStep.....	29-33
sPageableListR.....	21	vDynamicSubProclItem.....	20	vRedirection.....	16
sProcManager.....	27	vEAllItem.....	20	vRole.....	7
sSession.....	2	vEAIStep.....	30	vSALInfo.....	5,9
sUser.....	10	vEISStep.....	34	vScriptStep.....	34
sWorkQ.....	12	vEventItem.....	20	vSortField.....	14,23
sWorkQManager.....	15	vEventStep.....	30	vStepContent.....	22
vACase.....	19	vException.....	3	vStepId.....	28-34
vACaseContent.....	19,22	vExceptionDetail.....	3	vSubProcCallStep.....	31
vACaseCriteria.....	19,23	vExternalGraftProcess.....	20	vSubProcCallItem.....	20
vACaseListState.....	19	vFConditional.....	14,35	vSubProcCase.....	20
vAccessUserRef.....	28,34	vField.....	13,19,20	vSummary.....	13,19,23
vAction.....	29-34	vFieldDef.....	28	vTable.....	7
vActiveUser.....	5,9	vFMarking.....	29,35	vTableField.....	7
vAddressUserRef.....	29,34	vFont.....	35	vTime.....	16
vAFMarking.....	35	vForm.....	14,35	vThreadInfo.....	5,9
vAGroup.....	6	vFRow.....	14,35	vTransactionControlStep.....	34
vAGroupContent.....	6,22	vGraftItem.....	20	vTransactionControlItem.....	20
vAGroupListState.....	6	vGraftStep.....	33	vUser.....	6
vANode.....	5,9	vGraftSubTask.....	20	vUserContent.....	6,22
vAProc.....	28	vGroup.....	6,11	vUserId.....	6,11
vAProcContent.....	22	vGroupContent.....	22	vUserListState.....	6
vAttribute.....	6,7,11	vGroupId.....	6,11	vWICContent.....	13,22
vAttributeDef.....	7	vLabel.....	35	vWICriteria.....	13,14,23
vAuditMsgDef.....	11,14	vListValidation.....	7	vWIFGContent.....	22
vAuditStep.....	19	vListState.....	6,13,17,19	vWIFieldGroup.....	13
vAutoStep.....	34	vNode.....	1,5,9	vWILocked.....	23
vAWIContent.....	13,22	vNodeCtx.....	2	vWISummary.....	13
vAWorkItem.....	13	vNodeId.....	1,5,9	vWorkItem.....	11,13
vAWorkItemListState.....	13	vNodeLocale.....	5,9	vWorkItemId.....	11,13
vAWorkQ.....	16,17	vNormalItem.....	20	vWorkItemListState.....	13
vAWorkQContent.....	22	vNormalStep.....	29	vWorkQ.....	6,11,14,16,17
vAWorkQListState.....	17	vOpenClientStep.....	34	vWorkQDeltaJMSId.....	14
vCase.....	19	vOSUser.....	6	vWorkQId.....	6,11,14,16,17
vCaseId.....	19	vOSUserListState.....	6	vWorkQListState.....	17
vCDQP.....	13,19,27	vOutstandingItem.....	20	vWorkQIdListState.....	6,17
vCDQPDef.....	16	vOutstandingItemContent.....	22	vWQSessionInfo.....	5,9
vClientInfo.....	5,9	vParticipation.....	16	xCaseManager.....	18
vClientLog.....	26	vPermission.....	29	xIPEConfig.....	24
vCommand.....	29	vPluginForm.....	30	xNode.....	4
vComplexRouterStep.....	33	vPredictionCriteria.....	19,23	xNodeManager.....	8
vConditional.....	29-34	vPredictedItem.....	19,28	xProcManager.....	27
vConfigInfo.....	5,9	vPredictedItemListState.....	19	xSession.....	2
vContentRequest.....	22	vPreference.....	7,11	xUser.....	10
vCriteriaRequest.....	23	vPriority.....	29	xWorkQ.....	12
				xWorkQManager.....	15